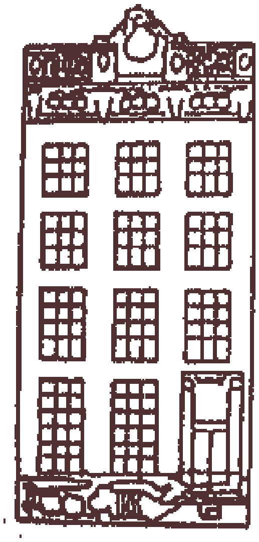
FRYSS

A first step towards Frisian TTS

Jelske Dijkstra



Institute of Phonetic Sciences
University of Amsterdam
Report nr. 142
2004

FRYSS
A first step towards Frisian TTS

Jelske Dijkstra

Supervisors: Dr. R.J.J.H. van Son (Institute of Phonetic Sciences, Amsterdam)
Dr. W. Visser (Fryske Akademy, Ljouwert/Leeuwarden)
Prof. Dr. Ir. L.C.W. Pols (Institute of Phonetic Sciences, Amsterdam)

MA Thesis Taalwetenschap, Speechcommunication and -technology
University of Amsterdam

June 2004

Table of contents

Preface

Already during the course of Speech Technology in 2002, the subject of my thesis became clear to me: try and create a Frisian TTS system. My failed and succeeded efforts to do so are put in this thesis report. To come to this ultimate goal I had help from lots of people to whom I want to express my gratitude.

First of all, I would like to thank my supervisors Rob van Son of the Chair of Phonetics in Amsterdam and Willem Visser of the Fryske Akademy in Ljouwert/Leeuwarden. They always sent me (back) into the right direction with their advices and remarks, both from different angles. Concerning the linguistic part of this thesis I was helped by Willem Visser. For the support on the part of speech synthesis I could ask Rob van Son for help. Further, I would also like to thank Louis Pols for his advices and support, although in the background. His contributions were very helpful.

Moreover, without Rob van Son and Louis Pols I would never have participated in the 5th ISCA Speech Synthesis Workshop, held from 14-16 June 2004 in Pittsburgh, USA. They helped me a lot and together we wrote a fine paper on this thesis. So, a special thanks to both of them is in place here.

Another special thanks goes Joop Kerkhoff and even more so to Erwin Marsi. Both of them always answered my (enormous amount of) questions patiently and in a clear way. Also I want to express my gratitude to the Fryske Akademy for giving me the opportunity to work with an pre-final version of the Frysk Hânwurdboek. A special thanks hereby goes to Hindrik Sijens, who sent me over 63,000 words with their pronunciations and handed me lots of corrections back as well. Although it was a large job, we had a lot of fun doing it.

Furthermore, I would like to thank a few other people who, one way or the other, helped me during this thesis period. Like David Weenink, who helped me conducting a Perl script for developing the pronunciation lexicon, Paul Boersma, who gave me advices about the phoneme set for Frisian, Ton Wempe, who always got me out of (computer) trouble, Hugo Quené, who provided me with a function word list for Dutch, and Renée van Bezooijen, Vincent van Heuven and Durk Gorter for giving useful advices.

Of course I would also like to thank all the respondents for their participation in the evaluation test and everybody I have forgotten in this long list. Last, but not least, I would like to thank my parents for giving me the opportunity to acquire a university degree.

1. Introduction

With the computerization of modern society and the resulting large growth of digital media, it is important for every language, great or small, to take part in this sphere of usage. Exclusion can be a major setback. For example, when a speaker cannot use his native language for accessing speech-related digital applications in education, commerce or media, he can look at his own language as not being fit for the modern age. In a sense one could say this has consequences for the attitude towards the mother tongue. And, when looking at a higher level, no digital applications being available for a certain language means exclusion from the digital domain; a domain that will grow even more in the future. The prospects of any language depend on its sphere of usage. Whenever a language is excluded from a domain of life, it becomes less attractive for its users. If these exclusions progress, the language will end only being used at home. And then, when its speakers choose not to pass this language on to their children, the language will eventually disappear, together with its wealthy and valuable culture.

Language is often linked to the future of the community. So it is no surprise that speakers of minority languages are in constant struggle to keep their language maintained in important domains, like, education and mass media. Often languages have to content with a lack of money to support these domains. As for language and speech research, only a few minority languages are prosperous enough to host a viable commercial speech technology market. Sometimes, minority languages have access to funds to develop their own systems, e.g., the Simputer project in India[1], where a fully integrated speech interface is developed. Sometimes, an individual researcher is in a position to complete such a project, e.g., the Welsh Text-to-Speech system (Williams, 1995). However, for almost all other minority languages, any speech and language technology application has to be developed as a community project. Often research depends on grants and gifts. This results in small scale projects for research, handled by volunteers and/or one single researcher. Moreover, the small resources available (if any) are often unpredictable and intermittent. Using a prototype of, for instance, a Text-to-Speech (TTS) system, can enable researchers to predict the costs and time needed for developing a full scale application more precisely. Also, developing such a system for a minority language could be a stimulus for the

[1] see also http://www.simputer.org

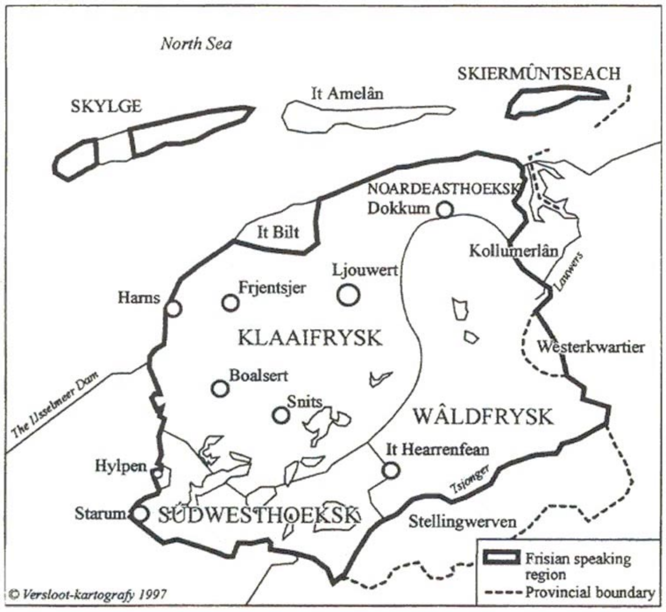
community to spend more time and money in speech and language research and to build other resources in that language.

In my MA-project I changed an existing Dutch TTS system based on the multi-lingual TTS system of Festival (Black et al., 1999), step by step into a Frisian TTS system. I am a native speaker of this language, which is a minority language in the Netherlands. Like in many other minority languages, there are few linguistic, digitalized resources for Frisian. So, the challenge of this project is making the system produce Frisian speech in the best way possible with a minimum of resources. The newly created TTS system is called FRYSS: Fryske Spraak Synteze (Frisian Speech Synthesis). Besides this Frisian TTS system, a prototyping framework for building TTS for other minority languages with few or no resources was developed.

2. Frisian[2]

Frisian is a minority language, spoken in the province of Fryslân, in the north of the Netherlands, and in a few border villages in the neighboring province of Groningen. With over 634,000 inhabitants this province counts less than 4% of the total population of the Netherlands. Of the population of Fryslân 74% is able to speak Frisian. In addition, 55% of the inhabitants of Fryslân has learned Frisian as their mother tongue, which comes roughly down to 350,000 native speakers (Gorter, 2003). Furthermore, 74% of the population is able to understand Frisian, 65% can read the language and 17% can write in Frisian (Gorter & Jonkman, 1995).

Over the last decades the language surveys of 1967, 1980 and 1994 show a small decline in the ability to speak Frisian. Also, the language is influenced by Dutch more and more (Breuker, 2001). It is suspected that these tendencies will continue in the future.



Map 1: Dialect map of Fryslân (Versloot cartography 1997, in: Visser, 1997)

Frisian counts three main dialects: Klaaifrysk, Wâldfrysk and Súdwesthoeksk. Some of the literature also mentions a fourth main dialect, viz. Noardeasthoeksk (Visser, 1997). There exist several smaller dialects as well, mostly mixtures of Dutch and Frisian. But despite of this huge

[2] There are several variants of Frisian. I want to point out that when I speak of Frisian in this thesis, I mean West Frisian, spoken in the province of Fryslân in the Netherlands. Other variants of Frisian are spoken in Germany, viz. North Frisian, close to the border of Denmark, and Eastern Frisian (or Sealtersk Frisian) spoken in a few towns in Saterland (near Oldenburg).

variety, in general all dialect variants are mutually comprehensible (exceptions are, e.g., Skiermûntseagersk, Skylgersk and Hylpersk). The standard variant of Frisian is mostly based on the Klaaifrysk forms of Frisian.

In 1970 the Frisian language was officially recognized by the Dutch government as the second language of the Netherlands. Since its recognition the position of Frisian has improved in the fields of education, media, science, church, public administration and law (Visser, 1997), though the amount of Dutch used in those formal domains is still considerably larger (Breuker, 2001). Its strongest domains are family, work and the village community (Gorter & Jonkman, 1995).

Looking at written sources in Frisian, one finds a small number of (literary) periodicals. Though, the literary production of books is considerable. Annually, about a 100 books of various kinds are published. The two daily newspapers in Fryslân produce less than 3% Frisian texts and one special Frisian page every week (Gorter, 2001). The Frisian sites on the internet are often literary as well. There are about a few hundred sites in Frisian, mostly bi- or trilingual.

3. The architecture of NeXTeNS/Festival

The Festival Speech Synthesis System[3], in short Festival, is an Open Source Text-to-Speech system, which can be used as a framework for building TTS systems in other languages. It is developed at the Centre for Speech Technology and Research of the University of Edinburgh. Festival is built up from modules, each handling a certain aspect of language, e.g., sentence accent or pause breaks. This means that it is relatively easy to change the system to another language. The modules are mainly built in the Scheme programming language. The system itself is written in the more concise C++ (Black et al., 1999).

Festival already implements several languages. Since 2003 there is also a Dutch version available, developed among others, by Erwin Marsi (University of Tilburg) and Joop Kerkhoff (University of Nijmegen) in the NeXTeNS-project[4]. NeXTeNS stands for "Nederlandse Extensie voor Tekst Naar Spraak" (Dutch Extension for Text-to-Speech). The purpose of this Dutch implementation of Festival was developing a clean, multi-platform, Open Source TTS-system to be used in research. The waveform synthesizer operates on the MBROLA[5] diphone synthesizer and it runs on the Dutch nl3-voice.

The architecture of NeXTeNS is derived from the standard architecture of Festival:

- Token Module: tokenisation, i.e., change tokens into words
- POS Module: Part-Of-Speech tagging
- Syntactic Module: syntax parsing
- Phrasing Module: phrase break prediction
- Intonation Module: placement of sentence accents
- Tune Module: tune choice needed for ToDI[6]
- Word Module: grapheme-to-phoneme conversion with lexicon, letter-to-sound-rules, and building of prosodic structures
- Pauses Module: insertion of pause segments

[3] see also the website of Festival Speech System: http://www.cstr.ed.ac.uk/projects/festival

[4] see also the website of NeXTeNS: http://nextens.uvt.nl

[5] see also the website of MBROLA: http://tcts.fpms.ac.be/synthesis/mbrola.html

[6] Transcription of Dutch Intonation (Gussenhoven et al., 2003), the intonation contour used in NeXTeNS, see also Appendix B: B.5.11).

- Postlexical Module: assigning postlexical rules and phone mapping to the phones of the nl3 MBROLA database
- Duration Module: determination of segment and pause-durations
- Fundamental frequency control: apply ToDI intonation contour to utterance
- Waveform synthesis: sending TTS-information to MBROLA-voice

The precise operations of these modules are (among other things) specified in Appendix B.

Nintens is the GUI (graphical user interface) of NeXTeNS. In this GUI it is possible to synthesize utterances, and look at the ToDI-accents and boundary tones, the F0-contour and the phoneme-string. It is also possible to manipulate these ToDI-values and some parameters, like e.g., speaking rate. Further, one is able to look at the underlying source code in the tab called "Log", and to type commands manually. Hopefully it becomes possible to manipulate at other levels than ToDI in the future as well.

During the synthesis process, the program constructs various relations around the utterance, which help to collect the information and segments needed. One of these relations is for example the Word relation, which contains every word in the utterance. Also information about breaks and sentence accent are stored here. The relations involved in the synthesis of an utterance can be checked with the next command (which synthesizes "hello world" in Dutch):

```
festival> (utt.relationnames (SayText "hallo wereld"))
```

The first beta-release of NeXTeNS was used for building a Frisian TTS system. The conversion was performed incrementally, in a step-by-step fashion. In this way a demonstration of a working system could be given at all times, which was important for debugging.

4. Changing NeXTeNS into the Frisian FRYSS

This chapter summarizes which resources and actions are needed to change NeXTeNS into a Frisian TTS system. It also shows which solutions were used when necessary resources lacked for Frisian. For more details and a step by step description of this transformation I refer to Appendix B which gives a complete overview of all changes made per module.

4.1. Preparations

An inventory of available digital recourses was already been made during the Speech Technology course of 2002. During that time, I also obtained some experience with Festival, because I changed a Spanish Festival version into a Frisian one that could generate one word in Frisian.

One of the first steps was copying the files of the Dutch voice of NeXTeNS and change the name and extensions to Frisian ones. Another step was widening my knowledge about the programming language Scheme with help from the manuals of Festival (Black et al., 1999; Black & Lenzo, 2003), and a reference manual (Texas Instruments, 1990). During this time-consuming phase I discovered a bug in the *net_nl_tune.scm* file. Further, some old and out of use source code was detected.

4.2. Phoneme set

A Frisian phone set (also called: phoneme set) had already been created during the Speech Technology course of 2002. This phone set was derived from the SAMPA set, used by the Fryske Akademy. Instead of SAMPA, I used the Worldbet-annotation (Hieronymus, 1994) to code the actual symbols, because it codes each IPA symbol uniquely over all languages. Moreover, Worldbet allows transparent coding of complex sounds (e.g., triphthongs, nasalized diphthongs) and transitions between narrow and broad transcriptions. For Frisian this is needed when dealing with nasalized vowels (e.g., nasalized diphthongs) and triphthongs, that go beyond SAMPA's two character codes.

Due to several uncertainties I reviewed this set quite a few times during the thesis period, sometimes with the expertise of Paul Boersma. One can find the Frisian phone set in Appendix A. There are several remarks about this set. Below is a list of remarks about phonemes that I use and that are different compared to those in other phoneme sets of Frisian. Unless stated otherwise, the Worldbet annotation is used in the next subsections.

4.2.1. Remarks: consonants

Cohen et al. (1961) only distinguish the bilabial [w] and labiodental [V|][7]. They do not speak about the fricative [v]. Instead of [v] they annotate the voiced counterpart of [f] as [V|], e.g., "haffel" [hAf&l] (mouth) and "hawwe" [hAV|&] (to have) (Cohen et al., 1961).

In the dictionary of Zantema (1992), the initial sound of the word "wetter" (water) is transcribed as the labiodental voiced fricative [v], equal to the [v]-sound in "skevel" (wag). Here, no labiodental approximant [V|] is distinguished. As for bilabial [w] in first or second element of a diphthong, Zantema annotates [u̯] (Fryske Akademy dictionary-annotation).

Hoekstra & Siebenga (2001) distinguish only [v] and [V|] in the SAMPA-set of the Fryske Akademy. They claim this [V|], as in "wyt" [V|it] (white), is "pronounced somewhat like an approximant, or a [v] without friction" (Hoekstra & Siebenga, 2001:2). In their comments one can read they follow the SAMPA for Dutch when they transcribe a [V|], contrary to the [v]-annotation of the dictionary of Zantema (1992). One of the reasons is that transcribing [v] is, phonetically speaking, problematic because of the lack of friction. And looking from a phonological angle a [v] would break the rule of no word-initial fricatives in Frisian. Another motivation is found in the example of "kwart" (quarter), which would be transcribed as [kvat], if one used the phonetic signs used by the dictionary. Due to assimilation processes this transcription would likely be changed to [kfat] or [Gvat], which is not the case. Moreover, "Frisian does not exhibit onsets consisting of a voiceless plosive followed by a voiced fricative" (Hoekstra & Siebenga, 2001:3).

As for the diphthongs, as a first element Hoekstra & Siebenga (2001) distinguish [w] (bilabial pronunciation) and [u] as a second element.

Next to the bilabial [w] I think one should distinguish both [v] and [V|]. For instance in a word like "weve" [V|e:v&] (to weave), in my opinion, more friction is heard at the [v]-sound, than at the [V|]. This could be due to the following vowels, or to the influence of Dutch in my pronunciation. A minimal pair with [v] and [V|] in medial position is found in: "hawwe" [hAV|&]

[7] In Worldbet the labiodental approximant [ʋ] (IPA-symbol) is annotated as [V[ ], but since this is confusing when one also uses square brackets, I chose for the annotation of [V|].

(to have) and "aventoer" [Av&ntu&r] (adventure), or "avesearje" [Av&sI&rj&] (to shoot up). The distinctions between these sounds should be examined more deeply.

The palatal nasal [n~] is only mentioned by Cohen et al. (1961). Due to the nasalisation of the vowels before a cluster consisting of /nj/, the /n/ disappears. In my opinion, from hearing and feeling, the [j]-sound could change in a palatal nasal in those cases. Though Feitsma (1958) does not agree with this. Further, both Hoekstra & Siebenga (2001) and Cohen et al. (1961) distinguish the [S], although Hoekstra & Siebenga (2001) see it as a foreign sound. In my opinion, the [Z] should also be acknowledged, in agreement to Cohen et al. (1961). I doubt the claim that [S] and [Z] are foreign sounds. Frisian counts lots of /sj/-clusters in onset, and one would suspect that they would palatalize, due to assimilation. Feitsma (1958) does not agree to this. She claims one does not hear [S] of the French word "chien" in the Frisian "sjonge" (to sing), but rather the combination of [sj] as in the French word "sien". Though, "chien" is actually pronounced as [SjE~], so this is not a good example and we, Paul Boersma and I, are not convinced by the argumentation of Feitsma. These palatal features should be examined more properly as well.

4.2.2. Remarks: vowels

In this phone set, the lax-vowels, viz. [ɪ], [ʏ], [ʊ] (IPA-symbols), were used as lowered [e], [ø], [o] (IPA-symbols), or [e̞], [ø̞], [o̞] (IPA-symbols), since they are located between [e], [ø], [o] and [ɛ], [œ], [ɔ] (both rows: IPA-symbols). These lax vowels (in Worldbet: [I], [Y], [U]) are also used in diphthongs, but not for long vowels. Here I used [e:], [7:] and [o:]. I realize this grouping is debatable, and I hope these sounds can be examined more properly in the future.

4.2.3. Remarks: Diphthongs/triphthongs

Hoekstra et al. (2001) transcribe the diphthong in the word "moai" (beautiful) as [o:i]. After recording and analyzing this sound with help from Paul Boersma we discovered this sound is pronounced as [U>i] or [U&i] by me. This looks more like a triphthong. It is not exactly clear what the second part of the triphthong is. I chose to use the latter notation because I also use the diphthong [U&], e.g., "boat" [bU&t] (boat). In this way it fits better in the phone set, as [U&i] can be seen an extension of [U&] with a glide. The sound is noted in Worldbet as [U&_i] (using the Worldbet extension rules).

These remarks about the phoneme set are also included in Appendix A. As one could see, Frisian contains many more long vowels and diphthongs which lack in Dutch. It even has triphthongs and nasalized vowels, since every vowel has, theoretically speaking, a nasalized counterpart.

4.3. Using Dutch voice by phone mapping

And what about the voice? Of course the TTS system needs an output. NeXTeNS uses the Dutch MBROLA nl3-voice. This database contains a set of diphones for Dutch. A diphone consists of two halve phones. It starts in the middle of the first phone and ends in the middle of the second one, i.e., it contains the transition between two phonemes (Rietveld & Van Heuven, 2001). In this way assimilation data is preserved best and the output quality is much better than in synthesis that uses a set of phones. The disadvantage of creating such a diphone set is, that one has to collect all possible diphone-combinations of a language (both within and between words). Theoretically, this should be the square number of the number of phones of that language. Therefore, constructing a diphone set is quite a job.

Since there does not exist a Frisian diphone set, and there was too little time to create one during this project, I decided to continue using the Dutch nl3-voice instead and map the Frisian vowels and consonants to their closest Dutch relative. A similar approach was used by Campbell (1998). He produced synthesized speech in another language (viz. English) than the database speaker (Japanese) to create a multi-lingual TTS system. Unfortunately the quality of the resulting speech by mapping was not good enough. He improved this by using the cepstral information of similar speech of a native speaker of the target language with the segments of the pre-stored voice. This procedure was out of the scope of the Frisian TTS and thus not used.

The phone mapping takes place in two stages. In the definition of the Nucleus in file *net_fy_lex.scm* complex phones, like diphthongs and nasalized vowels are mapped to the components they exist of. This means, for example, that [i&] of the word "iepen" [i&p&n] (open) is mapped to [i] and [&]. Likewise with [u:_~] in the word "jûns" [ju:_~s] (in the evening), this nasalized vowel is mapped to [u:] and [n] to preserve some sort of nasality in the output. All diphthongs and nasalized vowels were mapped in this way. The diphthong [I&], like in the word "hea" [hI&] (hay) was even mapped to three sounds. The output after mapping to [I] and [&] was not considered good enough since it sounded more like [Ib&]. This problem was intercepted by putting the glide [j] between the two parts, resulting in [hIj&] for the word "hea". Mapping in this section of *net_fy_lex.scm* was done to Frisian phones.

In the second stage of the mapping process the Frisian sounds are mapped to their Dutch closest relative, which are annotated in SAMPA. This mapping section is located nearly at the bottom of the file *net_fy_postlex.scm*. Here, also the triphthong [U&_i], e.g., "moai" [mU&_i] (beautiful), was mapped to the diphthong [oi] (SAMPA-annotation). Mapping to its components provided a bad output with a glottal sound. That is why is chosen for mapping to a diphthong.

Long vowels, like [u:] in "lûd" [lu:d] (sound, noise), [y:] in "drúf" [dry:f] (grape), and [i:] in "tiid" [ti:t] (time), were created by lengthening of the vowel duration of [u], [y] and [i].

4.4. Grapheme-to-phoneme conversion

Another important element of a TTS system is the grapheme-to-phoneme conversion, where a word is changed from a orthographic notation into a phoneme-string. During grapheme-to-phoneme conversion, first the word is looked up in a pronunciation lexicon. In case the word is not available in this lexicon, it is built up by so-called letter-to-sound rules. Both letter-to-sound rules and pronunciation lexicon needed to be created for Frisian, although the first was already been done (for the most part) during the preceded Speech Technology course. The Frisian letter-to-sound rules were revised and attached to the system. With help of a Perl script the words and pronunciations of a pre-final digital version of the Frysk Hânwurdboek (De Haan & Sijens, forthcoming), lit. "Frisian concise dictionary", of the Fryske Akademy was changed into a Scheme readable format. The resulting lexicon, called *fhwlex-1.0.out* was also attached to the system.

4.5. Sentence accent

POS, or Part-of-Speech tagging, is used for assigning sentence accent and breaks in the utterance. A POS tag indicates if a word is for example a verb or a noun, etc. Since such a POS tagging file does not exist for Frisian, I chose to assign sentence accent based on a function/content word division. When a word is not available in a self-made function word list for Frisian, it is treated as a content word and accented. In a group of more than two accents in a row, each second accent is removed. In this way the output sounds particularly more natural and rhythmic than without this removal. Breaks are assigned based on punctuation.

4.6. Fundamental Frequency Control

There has not been done much research on Frisian intonation. Most of the literature on Frisian intonation claim there is no difference between Dutch and Frisian intonation patterns, though this is never really investigated. Only Hoekstra (1991) has taken a step in this direction with his study on prepositions, where he found that lexical and specific functional prepositions are more often stressed in Frisian than in Dutch, and less often than in English (Hoekstra, 1991).

Therefore I chose to use the Dutch intonation of ToDI (Gussenhoven et al., 2003), which was already implemented in and used by NeXTeNS. ToDI stands for "Transcription of Dutch Intonation. For more information about these intonation structures I refer to (Gussenhoven, 2004) and the interactive course of ToDI on the internet: http://todi.let.kun.nl/ToDI/home.htm.

Again, for more specific details on the changes that have been made, or an explanation per module, I refer to Appendix B.

5. Evaluation

5.1. Informal testing of TTS system

First some informal testing was done with newspaper texts taken from internet. Subsequently, several mistakes were corrected. Others still remained, though. What follows is an overview of things that still go wrong.

One of the most disturbing inconveniences is the lack of nasalized vowels in the output, since this is an important feature of Frisian. Because there are no nasalized vowels in the nl3-database, this problem can only be solved once a Frisian diphone database is created. Secondly, the output can suddenly be interrupted when a certain diphone combination is not available in the nl3-database. This problem occurs for instance when dealing with the combinations: Oi-@, Oi-I, Oi-i, Oi-A, Oi-j (SAMPA-annotation) for instance in the word "maaie" (May), which is pronounced as [mOi@] (SAMPA).

Another disturbance is the wrong placements of stress in some plurals or derivations, since these words are not available in the lexicon and built up by letter-to-sound rules which place stress on the first syllable unless the Nucleus of this syllable is schwa. In that case lexical stress is placed at the second syllable. For example, "abrikoazen" [AbrikU&z&n] where lexical stress is placed on the first syllable "a-" due to letter-to-sound rules, while it should be on the last-but-one syllable "koa-". Likewise with the diminutive "abrikoaske" [AbrikU&sk&]. Lexical stress is again placed at the first syllable "a-", while it should be on "koa-". This problem could be solved by inserting all morphological variants in the pronunciation lexicon, or by developing a morphological analyzer (see also Möbius, 1998)) which abstracts the root from the synthesized word. Perhaps it is possible to convert this root to a phoneme-string by a lexicon lookup and just add the affixes to it.

Further, the word "dy". "Dy" can be a personal pronoun (with the meaning: you, 2nd person, object), or a demonstrative pronoun (meaning: that (one), those (ones)). "Dy" in the meaning of "you" is pronounced as [dEi] in Klaaifrysk. In Wâldfrysk, though, it is pronounced as [di]. But since standard Frisian mostly coincides with Klaaifrysk, only the first option was chosen in the pronunciation lexicon. Unfortunately, this means that "dy", in the meaning of "that/those", which should be pronounced as [di], is now pronounced as [dEi] as well.

The synthesis of numbers with points or commas, e.g., 130,000 or 2.5, fails because the number is not recognized in the Token Module and treated as a word. Since the number is not a word in the lexicon, it is built up by letter-to-sound rules. But, only words consisting of letters can be built by letter-to-sound rules, so "nil" is returned (and pronounced). By deleting the points or commas in large numbers, one could intercept this problem. It is not possible for FRYSS to pronounce numbers with decimals at the moment.

Though words built up by letter-to-sound rules often contained mistakes, at first I was under the impression the number of mistakes in the conversion was not disturbing. I knew lots of pronunciation mistakes concerned schwa mistakes in which [e:] or [E] is returned instead of [&]. Also other mistakes that could not always be gathered from spelling, were known to me, like for instance, the letters "oe", that can stand for [u] or [u&], "ie", that can stand for [i] or [i&], and "ei", that has not always the pronunciation of [>i], but can also be pronounced as [Ei], e.g., in "elektrisiteit" [e:lEktrisitEit] (electricity). At the end of the thesis period, just after the end evaluation test, I became more and more curious about the exact percentile rank of correctly converted words, so I decided to test the letter-to-sound rules. For the results of this test I refer to 5.3.

Finally, the synthesis of texts that contain apostrophes could go wrong. Before synthesizing the text, the space between the shortened "'e" (reduced variant of the definite article "de" (the) when it occurs following certain prepositions (see also Tiersma, 1999)), or "'t" (reduced variant of the particle "it" (the), before a vowel), and its preceding word should be deleted, e.g., "yn 'e tún" (in the garden) becomes "yn'e tún", similarly "yn 't âld hûs" (in the old house) becomes "yn't âld hûs". This procedure should also be taken at other reduced variants with apostrophes, like for example in "op 'en doer" (eventually). Otherwise the system would replace the apostrophe for a pause (after all, it is also a punctuation sign, see B.5.4. in Appendix B), while both words should be pronounced without a pause, e.g., "yn'e" should be pronounced as [in&], not as [in_&] ([_] stands for a pause-element), and "yn't" as [ynt], not [yn_t]. So, it is important to examine the text before synthesizing it and delete the space between those particles and their preceding words. Or else these sounds could cause problems in the intelligibility of the utterance. To be honest, I do not have a solution for this problem.

5.2. Evaluation with subjects

Two evaluations with subjects have been performed: one pilot study to produce preliminary results needed for a paper for the 5th ISCA Speech Synthesis Workshop (see Appendix D), and a bigger evaluation at the end of the thesis period.

5.2.1. Pilot study

While writing a paper about this thesis project for the 5th ISCA Speech Synthesis Workshop (see Appendix D) a pilot study was performed on eleven native speakers of Frisian over the internet. These subjects were informally selected from my personal contacts and via a Frisian student association (but also here, most respondents were contacts). They were asked to judge 20 sentences, harvested from internet sources such as newspapers, party manifestos, internet editions of literary magazines and publications of several youth associations. The subjects had to indicate the intelligibility, general quality and acceptability of the stimuli, each on a 7 point scale where higher is better. As for acceptability, the subject was asked if (s)he judged the stimulus to be acceptable as a first attempt to produce synthesized speech. At the time of this evaluation the pronunciation lexicon was not ready, so the stimuli were built up by letter-to-sound rules only. The utterance length varied between 9 and 19 words, and included features of Frisian where synthesis would go wrong, e.g., nasality of vowels, wrong placement of (default) lexical stress, and the feature of breaking. In breaking, vowel change takes place in derived forms of the stem, cf. "doar" [dU&r] (door) versus "doarren" [dwAr&n] (doors) and "doarke" [dwArk&] (small door); "hier" [hi&r] (hair) versus "hierren" [jIr&n] (hairs) and "hierke" [jIrk&] (small hair); "foet" [fu&t] (foot) versus "fuotten" [fwUt&n] (feet) and "fuotsje" [fwUtsj&] (small foot); "beam" [bI&m] (tree) versus "beammen" [bjEm&n] and "beamke" [bjEmk&] (small tree) (Tiersma, 1999). Breaking is a feature which cannot always be gathered from spelling.

Three of the eleven subjects were excluded from the final results, since they aborted the test. One of the remaining eight subjects only judged 18 of the 20 stimuli in a second attempt. His first trial was excluded because he aborted the test after 8 stimuli. This means that the total number of responses comes down to 158.

A division was made between long stimuli (>13 words) and short stimuli (≤13 words). Both sets contained 10 stimuli. The averages of the judgements are shown in Table 1.

Table 1: Mean judgements and standard error (between parentheses) on a 7 point scale, higher is better.

	short (N=78)	long (N=80)	total (N=158)
intelligibility	3.94 (0.21)	4.00 (0.18)	3.50 (0.14)
quality	3.67 (0.17)	3.78 (0.16)	3.38 (0.12)
acceptability	3.12 (0.16)	3.31 (0.15)	3.13 (0.11)

The synthesis quality and acceptability of the Frisian TTS is not stellar. The average judgements are actually below 4, the centre of the scale, whereas the scores for acceptability are just above 3. The low scores are probably the result of missing/wrong phonemes, the diphthongs (where components just were knocked together) and wrong lexical stress placement. It appears that the differences in scores between the long and short set are minimal.

Looking at the scores per stimulus, we see that some utterances have better results than others. This leaves room for improvement of the TTS. Since this was a pilot study, the result of this test should be seen as indicative only.

5.2.2. End evaluation test for this thesis period

At the end of the synthesis period a second evaluation was performed. Like the previous one, this evaluation was also done over the internet. Again, 20 stimuli were selected from internet sources, such as newspapers, party manifestos, literature, and magazines of youth associations. The stimuli were different from those used in the pilot study, though they also contained features where synthesis would go wrong, e.g., derived forms of stems (e.g., plurals), lots of diphthongs, breaking (though this problem was now mostly intercepted via the pronunciation lexicon), nasalized vowels, sentences that contain a shortened article (i.e., 'e or 't), and a question inside a someone's quote (here the system does not give a question contour in the F0).

Two Frisian student associations (other than the one from the pilot study), the association of the Frisian movement and a Frisian youth association were asked to send out an email to their members with an appeal to join the test. This email was also sent out by a Frisian linguistic mailing list. Since I am a (former) member of both student associations and of the youth association, most of the respondents are personal contacts.

In total, 54 subjects responded to my call. Unfortunately, 19 of these respondents had to be excluded from the test, either because they aborted the test[8], because they also participated in the pilot study[9], because Dutch was their mother tongue[10], or because the scores were very divergent compared to the others. This last point concerned one of the subjects, who never scored above 2, not even for the item speaking rate (which was never judged that often as 2 by any of the other respondents). Furthermore, this person needed the least time for the test, compared to the other subjects.

This meant that 35 of the subjects had valid scores. These subjects include three respondents who aborted the test after resp. eight, nine and eleven stimuli. This was probably due to the amount of time required for downloading the sound files.

The subjects were asked to judge the stimuli on 6 aspects: intelligibility, general quality, naturalness, lexical stress, sentence melody and speaking rate, each on a 5 point scale, where 1 stands for "min" (bad) and 5 for "goed" (good). As for speaking rate the lowest score stands for "stadich" (slow) and the highest for "fluch" (rapid). The stimuli varied between 7 and 21 words. Again, a set of long stimuli (≥13 words), and of short stimuli (<13 words) are compared. The results are shown in Figure 1.

[8] Seven subjects aborted the test after filling in their personal data, probably they did not achieve an audible signal on their PC. Another five subjects aborted after 1 to 3 stimuli. For at least two of these subjects the downloading of the sound files probably took too much time (concluded from the time table which came along with the results).

[9] This involved two respondents.

[10] This involved seven subjects, though three of them aborted the test. This leaves four native speakers of Dutch who participated in the whole evaluation test.

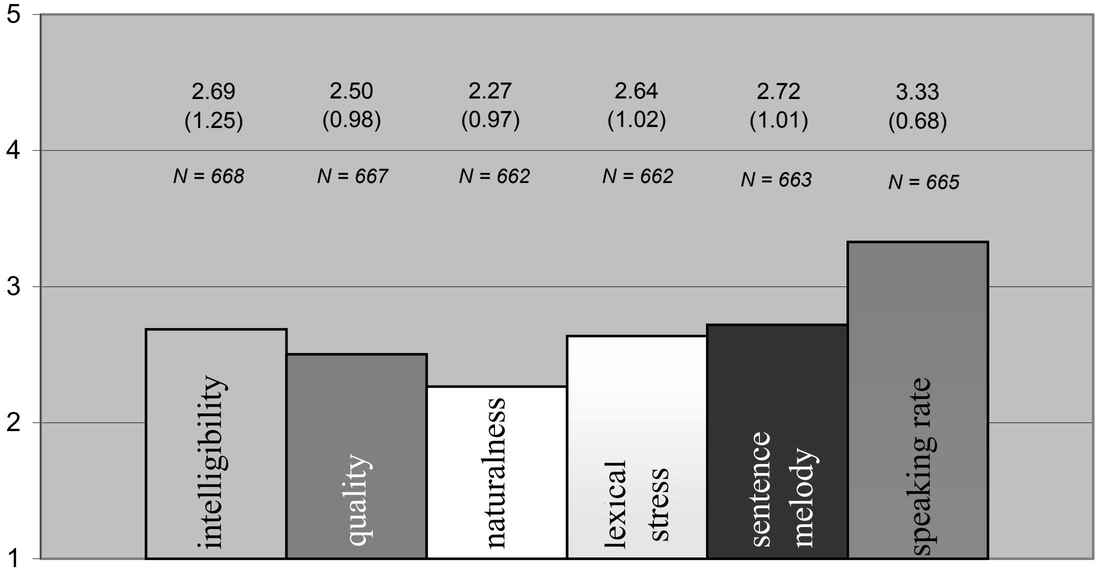


Figure 1: Mean judgements, standard deviation (between parentheses), and numbers of responses. Judgements on a 5 point scale, higher is better. For speaking rate higher is more rapid.

The mean scores of intelligibility, quality, lexical stress and sentence are judged below the centre of 3. Naturalness is evaluated lowest of all features. Low scores are probably due to features as diphthongs (where components were just knocked together), mistakes in pronunciation, and wrong accent placements, sometimes of lexical stress, sometimes of sentence accents. A distinction between short and long utterances, listed in Table 2, does not make much of a difference in the scores. Although, intelligibility is judged higher in short utterances. All other qualities, speaking rate excepted, are judged lower for the longer utterances, sentence melody the most.

Table 2: Mean judgements, standard deviation (between parentheses), and numbers of responses. Judgements on a 5 point scale, higher is better. For speaking rate higher is more rapid.

	short		***N***	**long**		***N***	**total**		***N***
intelligibility	2.57	(1.25)	*334*	2.80	(1.24)	*334*	2.69	(1.25)	*668*
quality	2.51	(0.99)	*333*	2.50	(0.97)	*334*	2.50	(0.98)	*667*
naturalness	2.31	(0.97)	*331*	2.22	(0.97)	*331*	2.27	(0.97)	*662*
lexical stress	2.67	(1.05)	*331*	2.58	(0.99)	*331*	2.64	(1.02)	*662*
sentence melody	2.79	(0.99)	*331*	2.64	(1.02)	*332*	2.72	(1.01)	*663*
speaking rate	3.30	(0.65)	*332*	3.35	(0.71)	*333*	3.33	(0.68)	*665*

As for speaking rate, the total scores are even just above the centre. This means that the speaking rate over all stimuli is judged quite normal, and, looking at the scores of speaking rate per utterance, where these judgements mostly varied between 3 and 4, sometimes (slightly) too fast. The number of responses (see Figure 1, or Table 2) vary because for unknown reasons respondents sometimes missed one or more judgements.

The four subjects with Dutch as mother tongue judged intelligibility over all stimuli just over the centre of 3. Another notable score over all stimuli was the one for lexical stress. The Dutch native speakers judged this quality as just below the centre of three, which is notably higher than the score of the Frisian native speakers. It has to be mentioned though that these four respondents all were personal contacts.

Since the scales are not identical to those in the pilot study, one cannot compare these scores with the ones from the pilot study properly. However, those scores were below the centre as well. All mean scores of the end evaluation test are above minimal (1) and, looking at the scores per utterance of all aspects, except speaking rate, some judgements have even the maximum score of 5. As mentioned in the pilot study, this leaves potential for improvement. FRYSS is intelligible, though it does not sound particularly good.

5.3. Testing letter-to-sound rules

Letter-to-sound rules have been tested by converting 1000 words from the pronunciation lexicon (i.e., every 630th word) into phoneme-strings through these rules. The results were very disappointing, since only 28.10% of the words were built correctly. I have not looked at the syllable division, since this is not always correct in the lexicon either. Moreover, in the letter-to-sound rules this division is based on spelling, in the pronunciation lexicon on sonority of phonemes. The results of the test are shown in Table 3.

Table 3: Results of building 1000 words from the pronunciation lexicon by letter-to-sound rules.

Correct words	281		28.10 %
Words with mistakes in phoneme-string only	525		52.50 %
Words with mistakes in the assignment of lexical stress only	22		2.20 %
Words with mistakes both in the assignment of lexical stress as in the phoneme-string	172		17.20 %

Most mistakes concerned pronunciation faults like the conversion of the grapheme "e", which often resulted in [e:] or [E] instead of [&]. Other mistakes were due to the feature of breaking or concerned letters like "oe" or "ie". Here, the correct conversion cannot always be gathered from spelling. Also, often vowels were converted as being long, but should be converted as short, or vice versa. This is mostly due to a bad syllable division. Maybe it is a suggestion to base the syllable division in the letter-to-sound rules on sonority as well. Some mistakes could be corrected in the rules, though automatically building new rules with TreeTalk could perhaps also intercept all of these problems. A disadvantage of building these rules automatically is that there are probably not enough words in the Frisian pronunciation lexicon to base these rules on. So it is not sure if automatically built rules would obtain better results than the handwritten ones.

6. Conclusions and suggestions for the future

The results of both judgement tests show that it is possible to develop a base-line demonstration TTS system for a language with a minimum of linguistic digital resources. Although not ideal, the system is intelligible. This case study can be considered as an example for other minority languages. With the help of this thesis one can estimate the costs and time for developing such a system. And hopefully, this project will be a stimulus to spend more money on speech synthesis in minority languages.

Since FRYSS was just a first step towards Frisian TTS and in this case a thesis period is actually too short to create a TTS system for a language with minimal digital resources, I would like to make some suggestions for the future. Though first I would like to make some remarks about NeXTeNS and Nintens, the GUI of NeXTeNS.

After some reading in the Festival manual, at first I was under the impression one could simply change a module and the system would still work afterwards. But since most of the variables are used in more than one module (e.g., phones, breaks, etc.), in most cases the system has to be debugged (endlessly) until it works again. This process of checking other files as well was sometimes a little bit frustrating when a little intervention gradually became a large one. Though without the modular structure, it would even be more difficult to make such changes, I suspect.

As for Nintens, I really missed the possibility to manipulate in the phoneme string and F0-contour, like in Fluent Dutch Text-to-Speech of Arthur Dirksen and Ludmila Menert of Fluency Speech Technology, Utrecht. I hope this will be possible somewhere in the future because it would certainly contribute to the user-friendliness of the system. I also regret the fact that under the tab of "Log" in Nintens the amount of signs or lines seems to be set on a maximum. When one has reached this maximum, one cannot look at the code generated by new synthesized utterances anymore. Now, it seems I only have complaints about Nintens. This is not true. Taking everything in account Nintens is a good program to work with, only one is not able to manipulate in the phoneme string and F0-contour, which sometimes can be a handicap.

Looking at all the remarks on the Frisian phoneme set, I wonder if a phonetic research on these difficulties would yield more certainties, since most phonemes are only investigated from a phonological point of view. Perhaps phonetic research can give some answers on the existence of [V|] (compared to [v]), [S] and [Z] (compared to [sj] and [zj]), and [n~] (see also Appendix A).

It is the intention that a Frisian diphone set will be created by Prof. Dr. Vincent van Heuven in the near future. The use of a Frisian diphone set would certainly improve the quality and intelligibility of FRYSS, since the pronunciation of some sounds is still somewhat like a Dutchman speaking Frisian.
A morphological analyzer (see also Möbius (1998)) that looks for the root of the word in the pronunciation lexicon, and adds the affixes to this phoneme-string, would improve the quality and intelligibility as well. Mistakes in pronunciation and wrong placement of lexical stress by building the word with letter-to-sound rules are avoided in this way. Another option is to insert all morphological variants in the pronunciation lexicon, though this is quite a job and one can never include all possible variants.

Furthermore, I am not satisfied with the letter-to-sound rules. Words, built up by these rules, still contain too many mistakes. Perhaps these results would be better when building a new set of rules automatically with TreeTalk. In that case one has to be sure though, that the syllabification is correct in the pronunciation lexicon. This is not always true at the moment. Also, a POS tagging file would improve the quality of the TTS. With a POS-tagger placement, of sentence accents and breaks could be improved. The option of translating a Dutch POS list to Frisian is an interesting approach to achieve such tagging.

Although this is not clearly proved by the outcomes of the second evaluation test (also because not all possible contours of ToDI can be reached by the system), I still wonder if ToDI gives the correct F0-contour for Frisian. Some researchers with Dutch as native language suspect that Frisian intonation is different from Dutch. I find it very difficult to hear this difference myself. I can hear that something is different, but I cannot say what it exactly is, or whether it is personal bounded or not. So I am very curious what investigation on Frisian intonation would yield.

References

Berg, R. van den, Gussenhoven, C. & Rietveld, T. (1992). Downstep in Dutch: implications for a model. In: G.J. Docherty and D.R. Ladd (eds.), *Papers in Laboratory Phonology II*, Cambridge: Cambridge University Press, p. 335-359.

Black, A.W. & Lenzo, K.A. (2003). *Building Synthetic Voices*. FestVox 2.0 Edition.

Black, A.W., Taylor, P. & Caley, R. (1999). *The Festival Speech Synthesis System*. Edition 1.4, for Festival Version 1.4.0.

Breuker, P. (2001). West Frisian in language contact. In: Munske, H.H. et al. (eds.), *Handbuch des Friesischen*, Niemeyer Verlag, Tübingen.

Campbell, N. (1998). Foreign-language speech synthesis. In: *Proceedings SSW3*, p. 117-180.

Cohen, A., Ebeling, C.L., Fokkema, K. & Holk, A.G.F. van (1961). *Fonologie van het Nederlands en het Fries. Inleiding tot de moderne klankleer*. Martinus Nijhoff, 's Gravenhage, 2nd edition (1st edition 1959).

Dijkstra, J., Pols, L.C.W., Son, R.J.J.H. van. (2004) Frisian TTS, an example of bootstrapping TTS for minority languages, In: *Proceedings SSW5*, Pittsburgh PA, USA.

Duijff, P. (2004). *Klam yn wurden*. Wurdboekstêf, Fryske Akademy, Ljouwert. Not published.

Feitsma, A. (1958). Om in nije stavering. It lûdsysteem fan de konsonanten. In: *Us Wurk*, Volume 7, p. 86-90.

Gorter, D. & Jonkman, R.J. (1995). *Taal yn Fryslân op 'e nij besjoen*, Fryske Akademy, Ljouwert.

Gorter, D. (2001). Extend and position of West Frisian. In: Munske, H.H. et al. (eds.), *Handbuch des Friesischen*, Niemeyer Verlag, Tübingen.

Gorter, D. (2003). Nederlands en Fries op gespannen voet? In: Stroop, J. (ed.) *Waar gaat het Nederlands naar toe?* Uitgeverij Ben Bakker, Amsterdam.

Graaf, T. de (1985). Phonetic aspects of the Frisian vowel system. In: *Nowele* 5, p. 23-40.

Gussenhoven, C., Rietveld, T., Kerkhoff, J. & Terken, J. (2003). *Transcription of Dutch Intonation, courseware*. http://todi.let.kun.nl/ToDI/home.htm (1st edition 1999).

Gussenhoven, C. (2004). Transcription of Dutch Intonation. In: Sun-Ah Jun (ed.), *Prosodic Typology: The Phonology of Intonation and Phrasing*. Oxford: Oxford University Press (in press).

Haan, R. de, Sijens, H. (forthcomming). *Frysk Hânwurdboek*, Fryske Akademy, Ljouwert.

Hieronymus, J.L. (1994). *ASCII Phonetic Symbols for the World's Languages: Worldbet*, AT&T Bell Labs, Murray Hill, USA.

Hoekstra, E. & Siebenga, S. (2001). *SAMPA for Frisian and Comments*. Fryske Akademy, Ljouwert.

Hoekstra, J. (1998). *Fryske wurdfoarming*, Fryske Akademy, Ljouwert.

Hoekstra, J. (1991). Oer it beklamjen fan ferhâldingswurden yn it Frysk, it Hollânsk en it Ingelsk. *Us Wurk*, Volume 40, p. 67-103.

Hoekstra J. (2002). Genitive compounds in Frisian as lexical phrases. *Journal of Comparative Germanic Linguistics* 6, p. 227-259.

Kager, R.W.J. (1989). *A Metrical Theory of Stress and Destressing in English and Dutch*, Doctoral dissertation, University of Utrecht.

Marsi, E.C. & Kerkhoff, J. (2003). *NeXTeNS*, http://nextens.uvt.nl

Möbius, B. (1998). Word and syllable models for German TTS synthesis. In: *Proceedings SSW3*, Jenolan Caves, Australia, p. 59-64.

Quené, H. & Kager, R.W.J. (1990). *PROS*, Research Institute for Language and Speech, Utrecht.

Rietveld, A.C.M. & Van Heuven, V.J. (2001). *Algemene fonetiek*, Uitgeverij Coutinho, Bussum, 2nd edition (1st edition 1997).

Schwartz, R.L. & Phoenix, T. (2001). *Learning Perl, Third Edition*. O'Reilly & Associates, Inc., Sebastopol, 3th edition (1st edition 1993).

Texas Instruments (1990). *PC Scheme, User's Guide & Language Reference Manual*. The MIT Press, Cambridge.

Tiersma, P.M. (1999). *Frisian Reference Grammar*, Fryske Akademy, Ljouwert, 2nd edition (1st edition 1985, published by Dordrecht Foris Publications).

Trommelen, M. & Zonneveld, W. (1989). *Klemtoon en metrische fonologie*. Dick Coutinho, Muiderberg.

Visser, W. (1997). *The syllable in Frisian*. HIL Dissertations 30, The Hague.

Vroomen, J., Bosch, A. van den & Gelder, B. de (1998). A connectionist model for bootstrap learning of syllabic structure. *Language and Cognitive Processes* 13(2/3), p. 193-220.

Williams, B. (1995) Text-to-speech synthesis for Welsh and Welsh English. In: *Proceedings Eurospeech*, Madrid, Spain, Volume 2, p. 1113-1116.

Zantema, J.W. (1992) *Hânwurdboek fan de Fryske taal*. Fryske Akademy, A.J.Osinga Uitgeverij, Drachten/Ljouwert, 12th edition (1st edition 1984).

Appendix A

Phoneme set for Frisian (Worldbet)

<table>
<tr><th>IPA-symbol</th><th>Worldbet</th><th>example</th><th>transcription</th></tr>
<tr><td>p</td><td>p</td><td>part (part)</td><td>p a t</td></tr>
<tr><td>b</td><td>b</td><td>bal (bal)</td><td>b > l</td></tr>
<tr><td>t</td><td>t</td><td>ta (to)</td><td>t a</td></tr>
<tr><td>d</td><td>d</td><td>daam (dam)</td><td>d a: m</td></tr>
<tr><td>k</td><td>k</td><td>klear (ready)</td><td>k l I& r</td></tr>
<tr><td>ɡ</td><td>g</td><td>goed (good)</td><td>g u& t</td></tr>
<tr><td>f</td><td>f</td><td>fol (full)</td><td>f o l</td></tr>
<tr><td>v</td><td>v</td><td>skevel (wag)</td><td>s k e: v & l</td></tr>
<tr><td>s</td><td>s</td><td>stil (quiet)</td><td>s t I l</td></tr>
<tr><td>z</td><td>z</td><td>wêze (to be)</td><td>V| E: z &</td></tr>
<tr><td>ʃ</td><td>S</td><td>lunch (lunch)</td><td>l Y n S</td></tr>
<tr><td>ʒ</td><td>Z</td><td>rûzje (to argue)</td><td>r u: Z &</td></tr>
<tr><td>x</td><td>x</td><td>rûch (rough)</td><td>r u x</td></tr>
<tr><td>ɣ</td><td>G</td><td>drage (to wear)</td><td>d r a: G &</td></tr>
<tr><td>h</td><td>h</td><td>heech (high)</td><td>h e: x</td></tr>
<tr><td>m</td><td>m</td><td>laam (lamb)</td><td>l a: m</td></tr>
<tr><td>n</td><td>n</td><td>noch (yet)</td><td>n > x</td></tr>
<tr><td>ɲ</td><td>n~</td><td>wenje (to live)</td><td>V| E~ n~ &</td></tr>
<tr><td>ŋ</td><td>N</td><td>ring (ring)</td><td>r I N</td></tr>
<tr><td>l</td><td>l</td><td>slikje (to lick)</td><td>s l I k j &</td></tr>
<tr><td>r</td><td>r</td><td>raar (strange)</td><td>r a: r</td></tr>
<tr><td>ʋ</td><td>V|</td><td>wetter (water)</td><td>V| E t & r</td></tr>
<tr><td>w</td><td>w</td><td>woartel (carrot)</td><td>w A t & l</td></tr>
<tr><td>j</td><td>j</td><td>jas (coat)</td><td>j > s</td></tr>
<tr><td colspan="4"><b>vowels</b></td></tr>
<tr><td>ə</td><td>&</td><td>de (the)</td><td>d &</td></tr>
<tr><td>i</td><td>i</td><td>dyk (dike)</td><td>d i k</td></tr>
<tr><td>ɪ</td><td>I</td><td>ik (I)</td><td>I k</td></tr>
<tr><td>ɛ</td><td>E</td><td>let (late)</td><td>l E t</td></tr>
<tr><td>y</td><td>y</td><td>nút (nut)</td><td>n y t</td></tr>
<tr><td>ʏ</td><td>Y</td><td>nut (use)</td><td>n Y t</td></tr>
<tr><td>ɑ</td><td>A</td><td>ta (to)</td><td>t a</td></tr>
<tr><td>u</td><td>u</td><td>rûch (rough)</td><td>r u x</td></tr>
</table>

ʊ	U	rom (large, wide)	r U m
ɔ	>	kat (cat)	k > t
iː	i:	tiid (time)	t i: t
eː	e:	reed (drive)	r e: t
ɛː	E:	bêd (bed)	b E: t
yː	y:	drúf (grape)	d r y: f
øː	7:	deun (tune)	d 7: n
aː	a:	baas (boss)	b a: s
uː	u:	sûch (dullard)	s u: x
oː	o:	rook (smoke)	r o: k
ɔː	>:	sâlt (salt)	s >: t
œː	8:	freule (unmarried noble lady)	f r 8: l &
diphthongs			
iə	i&	biede (to offer)	b i& d &
iu	iu	ieu (century)	iu
ɪə	I&	hea (hay)	h I&
ɛi	Ei	rij (row)	r Ei
ɔi	>i	laitsje (to laugh) [Klaaifrysk dialect]	l >i t s j &
ai	ai	laitsje (to laugh)	l ai t s j &
au	Au	gau (quick)	g Au
yə	y&	flues (fleece)	f l y& s
ʏə	Y&	gleon (glowing, red-hot)	g l Y& n
œy	8y	bui (shower of rain)	b 8y
uə	u&	goed (good)	g u& t
ui	ui	bloei (blossom)	b l ui
ʊi	Ui	floite (to whistle)	f l Ui t &
ʊə	U&	boat (boat)	b U& t
ʊəi	U&_i	moai (beautiful)	m U&_i

Some remarks regarding this phoneme set for Frisian[11]:

Consonants

The consonants [v], [z] and [G] do not occur at the beginning of a word.

[11] All the examples are presented in Worldbet-annotation, unless stated otherwise.

The consonants [g], [G], [x] occur in different positions, though sometimes they are positioned in the same context. The voiced stop [g] is usually located at the beginning of a word, the voiced fricative [G] in medial position between vowels and the unvoiced fricative [x] in word-final position. However, in some cases [x] and [G] occur in the same environment, e.g., "eagje" [E:Gj&] (to peer) and "eachje" [E:xj&] (small eye), or "bargje" [bArGj&] (to make a mess) and "barchje" [bArxj&] (little pig) (Cohen et al., 1961).

Cohen et al. (1961) only distinguish the bilabial [w] and labiodental [V|][12]. They do not speak about the fricative [v]. Instead of [v] they annotate the voiced counterpart of [f] as [V|], e.g., "haffel" [hAf&l] (mouth) and "hawwe" [hAV|&] (to have) (Cohen et al., 1961).

In the dictionary of Zantema (1992), the initial sound of the word "wetter" (water) is transcribed as the labiodental voiced fricative [v], equal to the [v]-sound in "skevel" (wag). Here, no labiodental approximant [V|] is distinguished. As for bilabial [w] in first or second element of a diphthong, Zantema annotates [u̯] (Fryske Akademy dictionary-annotation).

Hoekstra & Siebenga (2001) distinguish only [v] and [w] in the SAMPA-set of the Fryske Akademy. These sounds have the same annotation in Worldbet (Hieronymus, 1994). Hoekstra & Siebenga claim this [w], as in "wyt" [wit] (white), is "pronounced somewhat like an approximant, or a [v] without friction" (Hoekstra & Siebenga, 2001:2). In their comments one can read they follow the SAMPA for Dutch when they transcribe a [w] for the voiced approximant without friction, contrary to the [v]-annotation of the dictionary. One of the reasons is that transcribing [v] is, phonetically speaking, problematic because of the lack of friction. And looking from a phonological angle a [v] would break the rule of no word-initial fricatives in Frisian. Another motivation is found in the example of "kwart" (quarter), which would be transcribed as [kvat], if one used the phonetic signs used by the dictionary. Due to assimilation processes this transcription would likely be changed to [kfat] or [Gvat], which is not the case. Moreover, "Frisian does not exhibit onsets consisting of a voiceless plosive followed by a voiced fricative" (Hoekstra & Siebenga, 2001:3).

As for the diphthongs, as a first element Hoekstra & Siebenga (2001) distinguish [w] (bilabial pronunciation) and [u] as a second element.

[12] In Worldbet the labiodental approximant [ʋ] (IPA-symbol) is annotated as [V[ ], but since this is confusing when one also uses square brackets, I chose for the annotation of [V|].

Next to the bilabial [w] I think one should also distinguish [v] and [V|]. For instance in a word like "weve" [V|e:v&] (to weave), in my opinion, more friction is heard at the [v]-sound, than at the [V|]. This could be due to the following vowels, or to the influence of Dutch in my pronunciation. A minimal pair with [v] and [V|] in medial position is found in: "hawwe" [hAV|&] (to have) and "aventoer" [Av&ntu&r] (adventure), or "avesearje" [Av&sI&rj&] (to shoot up). The distinctions between these sounds should be examined more deeply.

Since the lingual r is for the most used in Frisian (Cohen et al., 1961), this variant is included in the phoneme set.

The palatal nasal [n~] is only mentioned by Cohen et al. (1961). Due to the nasalisation of the vowels before a cluster consisting of /nj/, the /n/ disappears. From hearing and feeling, the [j]-sound could change in a palatal nasal in those cases. Though Feitsma (1958) does not agree with this. I think this should be investigated more properly.

Both Hoekstra & Siebenga (2001) and Cohen et al. (1961) distinguish the [S], although Hoekstra & Siebenga (2001) see it as a foreign sound. In my opinion, the [Z] should also be acknowledged, in agreement to Cohen et al. (1961). I doubt the claim that [S] and [Z] are foreign sounds. Frisian counts lots of /sj/-clusters in onset, and one would suspect that they would palatalize, due to assimilation. Feitsma (1958) does not agree to this. She claims one does not hear [S] of the French word "chien" in the Frisian "sjonge" (to sing), but rather the combination of [sj] as in the French word "sien". Though, "chien" is actually pronounced as [SjE~], so this is not a good example and we, Paul Boersma and I, are not convinced by the argumentation of Feitsma. This feature should be examined more properly as well.

Vowels

After consulting Paul Boersma I chose to use the lax-vowels, viz. [ɪ], [ʏ], [ʊ] (IPA-symbols) as lowered [e], [ø], [o] (IPA-symbols), or [e̞], [ø̞], [o̞] (IPA-symbols), since they are located between [e], [ø], [o] and [ɛ], [œ], [ɔ] (both rows: IPA-symbols). These lax vowels (in Worldbet: [I], [Y], [U]) are also used in diphthongs, but not for long vowels. Here I use [e:], [7:] and [o:]. I realize this grouping is debatable, and I hope these sounds can be examined more properly in the future.

Diphthongs

The diphthongs of this phoneme set end in [i] or [u] instead of [j] or [w]. Another possible ending is in a schwa-sound. The diphthong [Y&] only occurs when it is followed by the consonants [n] or [r], e.g., "gleon" [glY&n] (glowing), "kleur" [klY&r] (color) (Hoekstra & Siebenga, 2001).

The diphthong in "laitsje" (to laugh) is transcribed by Hoekstra et al. (2001) as [ai]. Since this sound is usually pronounced as [>i] in the Klaaifrysk dialect and the standard Frisian is mostly based on this dialect, the [>i]-sound gets first choice.

Hoekstra et al (2001) use, just like Zantema (1992) eight rising diphthongs (of the feature of breaking), all starting in [j] or [w] as well. To my opinion, since these diphthongs start from a consonant, these sounds can also be characterized as two single sounds instead of a diphthong.

Triphthongs

Hoekstra et al. (2001) transcribe the diphthong in the word "moai" (beautiful) as [o:i]. After recording and analyzing this sound with help from Paul Boersma we discovered this sound is pronounced as [U>i] or [U&i] by me. This looks more like a triphthong. It is not exactly clear what the second part of the triphthong is. I chose to use the latter notation because I also use the diphthong [U&]. In this way it fits better in the phone set, as [U&i] is an extension of [U&] with a glide. The sound is noted in Worldbet as [U&_i] (using the Worldbet extension rules).

The dictionary of Zantema (1992) mentions seven triphthongs. Like Hoekstra et al. (2001) I would like to mention them in my remarks as well. Since these sounds too begin in a consonant ([j] or [w]), I do not talk about these triphthongs in the overview (see remark about rising diphthongs).

Below is a copy of Hoekstra et al. (2001:4), only adjusted to the Worldbet-annotation:

worldbetsymbol	example	transcription
wUi	muoike (aunt)	mwUik&
wai	moaist (most beautiful)	mwaist
wa:i	koai (artificial egg)	kwa:i
jyw	bliuw (stay, 1sg.)	bljyw
ju:	priuwe (to taste)	prju:&
jo:w	bleau (stayed, 1sg./3sg.)	bljo:w
jAu	fjouwer (four)	fjAu@r

Nazalized vowels

A vowel becomes a nasalized vowel when it precedes a consonant cluster existing of [n] followed by [s] [f], [v], [j], [w], [l], or [r] (Hoekstra et al., 2001; Cohen et al., 1961). It is never investigated to which vowels this rule applies. So, in theory, this phenomenon can happen with every vowel, diphthong or triphthong. Looking at the pronunciation dictionary, I also found the long nazalized vowel, viz. [I:_~], e.g., "tsjinst" [tsjI:_~st] (service). Visser (1997) mentions this vowel, that only occurs "in the context of nasalisation and only when preceded by the glide [j] that long /I/ shows up as [I:]" (1997: 22) as well.

In our Worldbet-annotation, the feature of nasality is indicated with a tilde-sign ([~]). When a sound already exist of two characters, nasality can by added with an underscore ([_~]).

Appendix B

Step by step changing to FRYSS

Being more familiar with working under a Windows Platform (Windows 98), I used the Windows version of NeXTeNS for working with synthesis. Unless stated otherwise, all files mentioned are located in the directory /net_fy_ib_mbrola/festvox/ of the Frisian voice.

B.1. Switch to another language

The first step of the actual conversion was copying the Dutch files, putting them in a new directory called Frisian and changing the extensions of all non-language specific files and definitions from "net_nl" to "net_fy", as FY is the language code for Frisian. The language initiation-file *siteinit.scm* (located in the lib-directory) was set to the Frisian voice: voice_net_fy_ib_mbrola. This voice is defined in the file *net_fy_ib_mbrola.scm*, where all files needed for TTS are activated. One could see this file as a kind of master file. To avoid errors in the synthesis process the command `(set! pos_lex_name nil)` was used in the file *net_fy_ib_mbrola.scm* to replace all source code for activating the POS-tagging file (located under "POS Module" in file *net_fy_ib_mbrola.scm*). This adjustment was necessary, because this POS-tagging file contained language specific information for Dutch, and could not be used for Frisian TTS. Also, other language specific files for Dutch were made inactive in *net_fy_ib_mbrola.scm*, for instance, the requirement of *net_nl_break_prosit.scm* and its parameter settings (located under "Phrasify Module" in file *net_fy_ib_mbrola.scm*), and the requirement of *net_nl_accent_prosit.scm* and its parameter settings (located under "Intonation Module" in file *net_fy_ib_mbrola.scm*).

The result of this entire switch was a TTS system which in name, when looking at the extensions, should convert Frisian text files to speech, but in fact still converted the texts with Dutch sounds and Dutch pronunciation rules.

B.2. Understanding Scheme

The next step was trying to understand the Scheme code and verify what happened in the different files. This was a very time-consuming phase. Though Scheme, a Lisp dialect, is a quite comprehensive language, many people do not like the language because it is filled with lots of parentheses. So, the GNU Emacs editor (which understands Lisp) was used, as recommended in the manual of Festival (Black et al., 1999). Furthermore, this Festival manual and the manual "Building Synthetic Voices" (Black & Lenzo, 2003) give some Scheme fundamentals for coding. There are also several internet sites[13] which provide some of the coding. As for myself I used the book "PC Scheme User's Guide & Software" (Texas Instruments, 1990) for extra support.

When one is stuck with questions regarding Festival or its source code, one can send them to the mailing lists of Festival and/or Festvox. The addresses can be found on their websites[14]. Here one can also find the archives of those lists. I was in the fortunate position that I could ask the people of the NeXTeNS-project all my questions. A helpful command in Festival is `(doc ...)`, where one can place an unfamiliar piece of code at the place of the dots.

B.3. Frisian phoneme set

A phoneme set was already created (see also Appendix A) and its vowels and consonants were implemented in the *net_fy_phones.scm* file (see also B.4.1.).

B.4. Preparation of a basic synthesizer structure for Frisian

B.4.1. Using the Dutch voice: insert Frisian phones and map these to Dutch ones

Because I used the Dutch nl3 diphone database, the Frisian phones had to be implemented between the Dutch ones in the definition of the phone set in the *net_fy_phones.scm* file. It was of great importance that these phones remained available in this file. If the Dutch phones were deleted, the system could not recognize the mapped Dutch phones at the end of the synthesizing process and would give an error message.

The file *net_fy_phones.scm* also contains a nasalized counterpart of every vowel. This is especially stated here, because these vowels are not noted in the phone set (see also Appendix A). As extra vowels [I:_~] and [&:] (long schwa) are noted. Both of these vowels were stated in some words of the pronunciation lexicon that was created later in this project.

Again, to get an output, the unknown Frisian phones had to be mapped somewhere in the synthesis process to their Dutch closest relative, using the SAMPA-notation of this Dutch voice.

[13] e.g., http://www.gnu.org/software/mit-scheme, or http://www.swiss.ai.mit.edu/projects/scheme, etc.

[14] Festival: http://www.cstr.ed.ac.uk/projects/festival; Festvox: http://festvox.org

This mapping process was inserted at the bottom of the *net_fy_postlex.scm* file (see also B.5.9.), where NeXTeNS already mapped some other Dutch phones to the ones of the nl3-database. This *net_fy_postlex.scm* file is located nearly at the end of the synthesis process.

A serious problem is that Frisian contains many more diphthongs than Dutch and even several triphthongs, that Dutch lacks. The Dutch nl3-database also lacks nasalized vowels; a very important feature of Frisian. The original idea was to map these complex sounds with the components they exist of. So, e.g., the Frisian diphthong [i&][15] would be built up by [i] and [&] (schwa). Because NeXTeNS uses diphone-synthesis I expected the transition between the two sounds would sound fairly natural, provided that the necessary diphone combinations were available in the nl3-database, of course. Unfortunately the mapping method in *net_fy_postlex.scm* only worked for one-to-one mapping (see also B.5.9.). Therefore, a solution for these complex sounds had to be found, later on in this TTS-project. For the moment, as an interim solution (one simply needs an output), these sounds were mapped to their first component, i.e., the diphthong [i&] was mapped to [i] only.

B.4.2. Other necessary adaptations

After dealing with the phones, the Dutch pronunciation lexicon directory (kunlex) was copied and called *fhwlex* (a self-made abbreviation of the Frysk Hânwurdboek, the name of the dictionary the Frisian lexicon should be derived from, see also Word Module). The copied *kunlex-1.0.out* file was renamed as *fhwlex-1.0.out* and the content was deleted. This resulted in an empty pronunciation lexicon. New word entries and pronunciations for Frisian were inserted later in the thesis period (see also B.5.7.1.). Secondly, a file with letter-to-sound rules for Frisian (see B.5.7.2.), called *net_fy_lts.scm*, was developed and attached to the system by the command of `(require 'net_fy_lts)` in the Word Module section of the file *net_fy_ib_mbrola.scm*. After making new references at the bottom of the *net_fy_lex.scm* file to the Frisian (empty) lexicon, a basic synthesizer structure for Frisian was created in which all words were built up by letter-to-sound rules.

[15] One could say this diphthong was available in the nl3 Dutch voice, looking at Dutch examples like "bier" [bir], which is rather pronounced as [bi&r]. Here, a schwa, [&], arises, influenced by [r]. Though there is no diphthong like that available in the nl3-database. Since this database consists of diphones, and a diphone concerns the transition between phones, this diphthong-effect to the [i] is already available in diphone /i-r/. This is also true for other vowels influenced by [r].

B.5. Changing the modules

Next, I will discuss the changes made per module. So, from this point on the chronological order of processing over all modules is lost. The chronological order inside every module is mostly preserved. The concerning files are noted between parentheses behind the name of the modules in the next subsections.

B.5.1. Token Module *(net_fy_token.scm)*

Tokenisation is necessary for changing unknown tokens like abbreviations, numbers, symbols, acronyms and dates into words. In the beta-release of NeXTeNS, this file was not completed with language specific details for Dutch yet. To avoid problems in synthesis, abbreviations (from the Frisian-Dutch dictionary (Zantema, 1992)) were inserted into the code. One has to make sure not to insert the last dot of an abbreviation in this file. This last dot is seen by the program as a full stop instead of being a part of the abbreviation. A problem in this section is the ambiguity of some abbreviations. For instance, the abbreviation "dg." can stand for "dagen" (days) or "desigram" (decigram). Also abbreviations that exist of only one letter can cause problems. For instance, the abbreviation "a." can be a short form for the word "are" (119.6 square yards), but also for the letter "a", since the dot is seen as a full stop. Of these ambiguous abbreviations only one is working (the one that is mostly used in my opinion) at present, the other possibilities are made inactive.

In an earlier pilot project (i.e., the Speech Technology course of 2002) a copy of the number-to-word conversion for the Spanish el-voice was adapted to Frisian. The source code of this file was implemented in the *net_fy_token.scm* file. The order of pronunciation was changed to the order of pronunciation in Frisian (and Dutch). For instance, instead of converting the number "31" to "treinta y uno" (lit. "thirty and one") as in Spanish, this number was converted to the Frisian "ienentritich" (lit. "one-and-thirty"). The system could now pronounce integer, positive numbers.

Thirdly, this file contains the function word list. The problems of ambiguity of this list are discussed in B.5.2. Due to lack of time less attention has been paid to symbols, acronyms and dates. Examples of these implementations are given in the English version of Festival, though. This version contains a huge variety of token-to-word conversions.

B.5.2. POS Module *(net_fy_token.scm)*

POS, or Part-of-Speech, tagging is mainly used for accent and break assignment. Since so far no Part-of-Speech tagger for Frisian exists, I decided to make use of a simple function and content word division. This automatic POS tagging function, the guess-pos function, was not operational in NeXTeNS at the moment and after several unsuccessful efforts I chose to make a separate list, like the guess_pos-list, and fill it with function words. As mentioned, this list can be found in the *net_fy_token.scm* file. All kinds of function words from the Frisian Reference Grammar (Tiersma, 1999) were copied into this list and missing words that were available on a Dutch function word list (Quené & Kager, 1990) were translated into Frisian and inserted between them. Since the POS tagging file of NeXTeNS also contained several auxiliary verbs, I decided to copy these verbs from this file and translate them into Frisian. These were implemented in my self-made function word list with all its derivations as well.

As expected, there were some problems with ambiguity in this function word list. For instance, the word "jûn" (given) is also known as a content word with the meaning "evening".
A second setback was the fact that all variants of a word had to be present in the list. In case of numerals, this meant, for example, that all spelling variants of the number sixteen had to be inserted: "sechtjin", "sechstjin", "sechstsjin", and "sechtsjin". The same held true for the ordinals. Further, in Frisian the second person singular "do" (you) can also appear in the enclitic form "-sto", "-ste" or simply as "-st", when it directly follows the finite verb or a subordination conjunction (Tiersma, 1999),

cf. "do hast" (you have) versus "hasto", "haste" or "hast" (have-you).

Or:

It hûs datst kocht hast.
the house that-you bought have
'The house you have bought.'

During synthesizing texts for testing, I discovered that function words starting with a capital letter had to be present in the list as well. Otherwise, a function word would not be recognized as a function word when it occurred at the beginning of a sentence. This problem was solved in the *net_fy_accent.scm* file (see also B.5.5.). It is not until this file is reached, that the system checks whether a word is a member of the function word list. Now, just before this check, the word in question is set into lowercase letters.

Another possibility for getting POS-information would, for instance, be to translate a Dutch POS-file into Frisian. Of course, such a file would contain errors for Frisian, since POS-information is not always the same for Frisian and Dutch.

B.5.3. Syntactic Module *(net_fy_syntax.scm)*

Since there is no syntax parser for Frisian the default option of no syntax method was chosen.

B.5.4. Phrasing Module *(net_fy_break.scm)*

In the Phrasing Module breaks are predicted by means of punctuation. The default option is a punctuation cart tree (Black et al., 1999; Black & Lenzo, 2003). Alternatives are assigning breaks by means of POS. Since there is no POS-tagging file for Frisian available, the default option was chosen. Whenever an utterance ends in a full stop (period), exclamation mark, question mark, or semi-colon the value "heavy" is given to the feature pbreak (prosodic break) at that point. At the place of apostrophe, quotation mark (double quote), parentheses, comma, or colon, "medium" is given. The utterance always ends in a value for pbreak (i.e., "heavy" or "medium"). If the utterance does not end in a punctuation mark, the value of pbreak is "heavy".
The default file of Festival, also used by NeXTeNS, was followed in determining which value of the break feature belongs to the punctuation mark. This meant that the source code stayed the same, except for the question mark.

In the standard cart tree of Festival, at the place of a question mark "heavy" is given as value of the feature break. At the end of my thesis-period I changed the value at this mark into a new value, "question", to get a question-intonation in the prosody of the utterance (see also B.5.6.).

As mentioned in chapter 3, various relation streams are constructed around the utterance during the synthesis process. These relation streams help and collect the information and segments needed for synthesis. One of these relations is the Word Relation which contains all separate words of the synthesized utterance. Here, also the values of the feature pbreak (i.e., "heavy", "medium", "question") are stored. If a word is not followed by a pbreak feature, "0" is stored as value of pbreak at that word (see also chapter 3, and B.5.5.).

B.5.5. Intonation Module *(net_fy_accent.scm)*

As with breaks, sentence accents can also be assigned by means of POS. This method is used in NeXTeNS. Here nouns, adjectives, and verbs (except auxiliary verbs) get sentence accent. Since in FRYSS a simple function/content word division is used, this rule is replaced by one that gives accent to every word that is not a member of the function word list (see also B.5.2.). Before this function word check, the word is set in lowercase letters (if necessary) by the definition of net_fy_downcase. In this way, a capitalized function word can be looked up in the list as well.

If the first letter of the word is an /A/, /O/ or /U/, this letter is corrected in some cases. Frisian does not apply accents in uppercased letters, so these letters not only stand for /a/, /o/ or /u/, but also for /â/, /ô/, /û/, or /ú/. This rule, defined as net_fy_correct_downcase in *net_fy_accent.scm*, was applied at the end of the thesis period and is based on the content of the pronunciation lexicon (i.e., the pronunciation of the words of the Frysk Hânwurdboek). Each of the corrected letters contains a few exceptions, but in general:

- when the word starts with /ald/-, /a/ is changed into /â/;
- when the word starts with /of/- this /o/ is changed into /ô/;
- if the word starts with /ul/- or /un/-, the first letter is changed into /û/;
- if the word starts with /us/- or /ut/-, the first letter is changed into /ú/.

One would get the wrong pronunciation when, for instance, one wants to synthesize "Undertusken" (meanwhile). Without this rule, the result would be: [**Y**nd&rtYsk&n], while it should be [**u**nd&rtYsk&n]. Moreover, the lowercased word "undertusken" would not be recognized as a function word (because the lowercased function word is "ûndertusken") and it would be treated as a content word. In short, one would not just get the wrong pronunciation in this example, but also a function word with sentence accent.

Back to the actual function of this file: placing sentence accents. Placing sentence accent on the base of content words gave a restless and unnatural output with quite a lot of rises and falls in the intonation contour, when several accents occurred successively in an utterance. The rhythm was lost. Therefore, it was decided to remove each second sentence accent in a group of at least three accents. During the synthesis process, the program searches for three accents in a row, and constantly eliminates the middle accent of this group of three. So now there was a maximum of two successive accents that could occur in the output. The result sounded more natural, to my opinion.

The information about sentence accent is stored in the Word Relation. If a word should have sentence accent, the value "+" is given to the feature acc. One can check the information stored in the Word Relation with the next command (synthesized text (Frisian) means "we walk"):

```
festival> (utt.relation.print (SayText "wy rinne") 'Word)
```

This gives the next output on the screen:

```
()
id _3 ; name wy ; pbreak "0" ;
id _4 ; name rinne ; pbreak "heavy" ; acc + ;
nil
```

Here one can clearly see the contents of the Word Relation: separate words and information about prosodic breaks (see B.5.4.) and sentence accents. For more information about relations, see also chapter 3.

B.5.6. Tune Module *(net_fy_tune.scm)*

The words of the utterance are located in the Word Relation. Here is also stated whether a word has got a sentence accent (acc-feature, see B.5.5.) and whether the word is followed by a prosodic break (pbreak-feature, see B.5.4.). Now, ToDI-accents and boundary-tones are assigned to the words by means of these sentence accents and prosodic breaks. The different ToDI-values are stored in the Intonation Relation, which is created in this module as well. These values are necessary in the Fundamental Frequency Control Module when a ToDI-intonation is assigned to the utterance.

ToDI stands for Transcription of Dutch Intonation (Gussenhoven et al., 2003). In this section I will only talk about which ToDI-accents and boundary-tones are assigned to the words of the utterance. For information about the different ToDI-values, see B.5.11. or the interactive course of ToDI on the internet[16].

[16] see http://todi.let.kun.nl/ToDI/home.htm

In NeXTeNS, the values "%L" and "L%" are assigned to the beginning and the end of each utterance respectively. When a word is preceded by a medium or heavy break somewhere in the middle of the utterance (i.e., in case of a comma or parentheses, see B.5.4.), the boundary tone "%" is assigned to the previous word, and an initial boundary tone ("%L") is placed at the current word. As mentioned earlier, the specific boundary tones are stored in the Intonation Relation. At the same time, a connection between the word and its boundary tone is made in the Word-Int Relation. This latter relation is created in this file as well.

Whenever a sentence accent is assigned to a word (i.e., when "+" is given to the acc-feature in the Word Relation, see B.5.5.), the program checks whether the word is the final accent of a non-final phrase. If this is not the case, the word is associated with a default pitch accent, i.e., "H*L", see the words "jonge" (boy) and "iepen" (open) in example (1). In the same example, the word "rút" (window) is a word with final accent of a non-final phrase:

(1) *De* ***jonge*** *rint nei it* ***rút****, en docht it* ***iepen****.*
%L H*L H* % %L H*L L%
the boy walks towards the window and does it open
'The boy walks towards the window, and opens it.'

When the word is the final accent of a non-final phrase, e.g., the word "rút" (window) in example (1), another ToDI-value is selected. Here, in example (1), the word "rút" (window) is the last word of the non-final phrase, so the value "H*" is selected. If the word is not the last word of the non-final phrase, e.g., the word "mem" (mother) in example (2), the value "L*H" is assigned to the word.

(2) *De* ***jonge*** ***draaft*** *op syn* ***mem*** *ta, want hy kriget in* ***suertsje****.*
%L H*L H*L L*H % %L H*L L%
the boy runs at his mother to because he gets a candy
'The boy runs to his mother, because he gets a candy.'

There is also a third option for a word with final accent in a non-final phrase: the value "H*LH". In NeXTeNS, this value is assigned to a verb before a relative clause. This ToDI-accent will never be reached in Frisian TTS, though, because there is no POS-tagging file for Frisian.

The original *net_nl_tune.scm* file of NeXTeNS (Marsi & Kerkhoff, 2003) does not contain a question intonation. At the place of a question mark in the utterance, NeXTeNS returned the value "L%", instead of "H%". So I included one change in the original source code of NeXTeNS by inserting the possibility to assign "H%" wherever the "question" feature of the prosodic break (see B.5.4.) occurred in the utterance. There are two situations in which this could appear, viz. at the end of the utterance (example (3)), or somewhere in the middle of the utterance (example (4)).

(3) *Giest nei* ***hûs?***
%L H*L H%
go-you to home
'Are you going home?'

(4) *Giest nei* ***hûs? Ik*** *gean ek fuort.*
%L H*L H% %L L%
go-you to home I go as well away
'Are you going home? I will leave as well.'

Without assigning "question" to the feature of the prosodic break, the word "hûs" (home) in example (3) would be associated with the value of "L%", since it is the end of the utterance. With this "question" marker, the value "H%" is assigned. Example (4) takes extra attention. Here, the word "hûs" (home) would be associated with the tone "%", since the prosodic break would have the value of a "heavy" break. Now, "H%" is assigned to the word, and, as expected, the initial boundary tone of "%L" is assigned to the word "Ik" (I). After assigning these boundary tones, it is important to change the value of the prosodic break feature from "question" back to "heavy" again. After all, this was the value of the prosodic break before the insertion of the question-feature in the Phrase Module (B.5.4.). The change to "heavy" is necessary since these features are needed again in the Pause module (B.5.8.) and during the calculation of duration (see B.5.10.).

Now, every accented word is associated with a ToDI accent, e.g., "H*L", "L*H", etc. and every boundary with a ToDI-boundary tone, e.g., "%L", "L%", "H%", etc. Once more, these associations between the words and their tone-elements are located in the Relation Word-Int. The actual words are still located in the Word Relation and the ToDI accents and tones in the Intonation Relation. Later on in the synthesis process, after building the prosodic word in the Word Module, the tone elements are associated more specifically with the intended syllables.

B.5.7. Word Module

In this module, the actual grapheme-to-phoneme conversion takes place. Every word in the utterance is looked up first in the pronunciation lexicon, and when it does not exist in this lexicon, it is built up by letter-to-sound rules. NeXTeNS uses the kunlex-lexicon and the FONPARS letter-to-sound rules, obtained by training data. In the next sections the realization of both the lexicon and letter-to-sound rules for Frisian are discussed. Also some attention is paid to the building of the prosodic word and phone-mapping of complex sounds in this module.

B.5.7.1. Pronunciation lexicon *(net_fy_lex_addenda.scm and /lib/dicts/fhwlex/fhwlex-1.0.out)*

As for the pronunciation lexicon one can distinguish an addenda (short list of hand added words) and a compiled lexicon (large lexicon, located in another directory).

The lexicon lookup process starts with the addenda, located in *net_fy_lex_addenda.scm*. This file could contain the pronunciation of letters (e.g., the letter "a" in example (5)), symbols (e.g., the asterisk symbol in example (6)), and punctuation (e.g., the punctuation mark comma in example (6)). Also, (common) words that are not available in the compiled lexicon and cannot be built properly by the letter-to-sound rules can be added to this file, e.g., the last name of the Queen's Commissioner of the province of Fryslân: "Nijpels".

```
(5)     (lex.add.entry '("a" nil ((((a:) 1)))))
(6)     (lex.add.entry '("*" n (((A s) 0) ((t &) 0) ((r i s k) 1))))
(7)     (lex.add.entry '("," nn (((k U) 1) ((m A) 0))))
(8)     (lex.add.entry '("Nijpels" nil ((((N Ei) 1) ((p & l s) 0)))))
```

This file *net_fy_lex_addenda.scm* is not used for Frisian at the moment, but could contain pronunciations for Frisian in the future. New lexical entries are added by hand. One has to make sure to use the same format as in examples (5)-(8). If no match is found in the addenda, the search continues with the compiled pronunciation lexicon, located in /lib/dicts/fhwlex/fhwlex-1.0.out.

The pronunciation lexicon is especially useful for words with irregular pronunciation, or with irregular stress patterns. I was fortunate to have access to a digital version of a pre-final dictionary, i.e., the Frysk Hânwurdboek (lit. "Frisian concise dictionary") (De Haan & Sijens, forthcoming), which contained over 63,000 word entries. The Fryske Akademy sent all word entries and pronunciation parts over email. Every entry and its pronunciaton had to be converted to a Scheme readable file. This conversion was done with a self written Perl-script (Schwartz & Phoenix, 2001). Before the conversion took place, the entries and pronunciations were checked for mistakes.

Because the Frysk Hânwurdboek was a pre-final version, lots of typing errors were present in the pronunciation part. Furthermore, I had to check every [v]-sign in the pronunciation data of the Fryske Akademy manually, because I distinguished the [V|] and [v], for which the Fryske Akademy both used the [v]-sign (see also Appendix A). Also diphthongs were manually checked. These had to have a dot in the middle, indicating that one was dealing with a diphthong-sign. If this dot was missing, the script would convert the two parts of the diphthong as monophthongs. Other mistakes were intercepted by special scripts that would give back all entries with unknown phonemes (due to typing errors, e.g., using [e] instead of [@] (schwa)), mistakes in lay-out (e.g., ending of the pronunciation part in ">" instead of "]"), and entries with two stress markers (i.e., two apostrophes in the word entry). Finally, a final script would give back the words with no accent at all. All mistakes were sent back to the Fryske Akademy again and their comments were processed in the material.

Now, the actual conversion could take place. This was done with a Perl script. The entry, e.g., the one for the word "bjusterbaarlik" (miraculous) in (9), had to be changed to a Scheme format, see (10).

(9) **bjuster'baarlik** [bjöst@rba:rl@k]

(10) ("bjusterbaarlik" nil ((((b j Y s) 0) ((t & r) 0) ((b a: r) 1) ((l & k) 0))))

First, the phonetic signs used by the Fryske Akademy were set to the Worldbet annotation. The next step in the conversion was the syllable division of the pronunciation part. This is needed for assigning word stress to the correct part (i.e., syllable) of the word in the pronunciation, based on the sonority principle. "The SP [Sonority Principle] states that within a syllable, sonority starts

low at the onset, increases towards a peak value at the nucleus position (...), and gradually decreases along the coda until the end of the syllable." (Vroomen et al., (1998): 196). Therefore sonorants (viz. nasals, liquids, and glides) only occur next to the nucleus. For this syllable division a script of Rob van Son was used. In this script, phones are associated with a certain weight of sonority, which corresponds to the degree of constriction of the vocal tract. Although this ranking of weight is not entirely consistent and does not provide for a perfect syllable division, the script provided a reasonably accurate result for Frisian. Problems arose with the [s] in compounds, which could belong to either the previous or the latter syllable, cf. "houtskroef" (wood screw), which has the next syllable division: [hAut] and [skru:f], versus "houtskoal" (charcoal) which should be divided as [hAuts] and [kU&l] (and not as: [hAut] and [skU&l]!).
Also, the [j] caused several problems. For example: "útjefte" (expense), where syllable division goes wrong: not [y], [tjEf] and [t&], but [yt], [jEf] and [t&]. But: "jubelteannen" (upturned toes), which is divided correctly as [jy], [b&l], [tjE], and [n&n].
Mistakes were mostly corrected manually afterwards. The syllable boundary was indicated in this Perl script as "-" and replaced later by parentheses (see below).

A third step was assigning lexical stress to the right syllable in the pronunciation part. In the original lemma of the Fryske Akademy, lexical stress was marked with an apostrophe before the stressed syllable in the entry (see example (9)). The stress marking had to shift to the correct syllable in the pronunciation part. This was done by counting the nuclei before the apostrophe in the entry (this is equal to the number of syllables before the stressed one) and then start counting until the same number of nuclei/syllables was reached in the pronunciation part. After that, the stress-marker "+" was assigned to the next one. This marker replaced the syllable boundary marker "-". Finally, the apostrophe was removed from the orthographic word, and this entry was placed between double quotes.
Next, POS-information was given. Since I wanted to follow NeXTeNS as much as possible, every word was accompanied by the feature "nil", as in the kunlex-lexicon. One could also assign a specific part of speech tag to it, to get a better chance for the correct pronunciation in its context. "Nil" matches any part of speech tag. The necessary parentheses were assigned to the syllables in the phonetic realization, and the stress-markers "+" and "-" were replaced by "1" or "0" respectively, where "1" stands for stressed syllable and "0" for no stress. Because the Frysk Hânwurdboek dictionary only contained primary accent placement, the Frisian synthesis was limited to primary accents as well.

The pronunciation lexicon should not only contain the base forms of a word, but all their morphological variants, like all conjugations of a verb, plurals and diminutives of nouns, etc., as well. These variants are usually not available in a dictionary. As for the Frysk Hânwurdboek only a few diminutives are present. Including all morphological variants is a large but realistic job to do. Unfortunately, looking at the time available for this MA-project, I had to leave these variants out. When one is dealing with languages with extensive word compounding or agglutinative languages like Finnish or Turkish, Black & Lenzo (2003) advise to develop a proper morphological analyzer to intercept this problem (see also Möbius, 1998). Also, this was outside the scope of the project. So, these morphological variants are built up with letter-to-sound rules now.

The results were stored in a file called *fhwlex.scm*, located in the /lib/dicts/fhwlex-directory. This directory also contained the empty pronunciation lexicon *fhwlex-1.0.out* (see also B.4.2.). This .out-file is a compiled file for lexicon lookup, which works more efficiently because of the use of a binary search and less loading time. The obtained Scheme-file was compiled to the .out-file with the next command:

```
festival> (lex.compile (path-append lexdir "fhwlex" "fhwlex.scm")
(path-append lexdir "fhwlex" "fhwlex-1.0.out"))
```

B.5.7.2. Letter-to-sound rules *(net_fy_lts.scm)*

When a word does not occur in the lexicon, it is built up by letter-to-sound rules. These can be made automatically and manually. Since Frisian has the advantage of having a relatively strong relationship between the letters in a word and their pronunciation, it was easier to write the rules by hand. The more so because this had already been done in the preceding Speech Technology course. In an ideal situation there would be a mapping from a string of graphemes to a string of phonemes. Though, this is difficult, because for example, a grapheme can correspond to different phonetic signs, like the grapheme "e" in "letterteken" [lEt&rte:k&n] (character), which can correspond to either [E], [&] (schwa), or [e:]. The letter-to-sound rules can be built from existing examples from the Festival distribution. A copy of the converted Spanish example, which was converted during the preceding Speech Technology course, was inserted in the system as a new file, called *net_fy_lts.scm*. It consisted of a conversion to lowercase letters, a grapheme-to-phoneme conversion, a conversion into syllables, and a definition for assigning lexical stress to the word in question. The definition to change certain vowels into weaker ones, needed for Spanish letter-to-sound rules, was removed. For practical reasons, syllabification has been put before the actual letter-to-sound rules. A separate definition for assigning the nasal feature to vowels was given later on. This means that a word is built up in the following way:

First the utterance is set to lowercase letters, then a division into syllables takes place. The hyphen sign is used as symbol for the syllable break. When two identical consonants occur a syllable break is given between those consonants. When a consonant is placed between two vowels a syllable break is given before the consonant. Furthermore, all possible consonant clusters are listed (from Cohen et al., 1961) together with their syllable breaks. Then, the actual letter-to-sound rules in which graphemes are changed into phonemes, can take place. Also reaks that occur at the wrong place are for the most part corrected here. The letter-to-sound rules have the following form (Black & Lenzo, 2003), where LC stands for left context, RC for right context:

(LC [alpha] RC = beta)

In practice this comes down to the next examples:

(11) ([y] = i)

(12) (VOWEL [- g] VOICEDC = - G)

Example (11) is a simple letter-to-sound conversion. The phone [i] is assigned to the letter "y". In case of example (12) a voiced [G] is given whenever the letter "g" is placed between vowels (left side) and voiced consonants (on the right). As mentioned earlier, the hyphen sign is the annotation for a syllable break.

In the next step of the letter-to-sound conversion the nasalization rule for vowels is assigned (see also Tiersma, 1999). When a vowel is followed by /n/ and /s/, /z/, /f/, /v/, /j/, /r/, /l/, or /w/ (orthographic signs in Tiersma), it becomes a nasalized vowel. In the definition of nasality the sounds [s], [z], [S], [Z], [f], [v], [j], [r], [l], and [V|] are used instead of the orthographic signs. The vowel in question is changed into its nasalized counterpart (to be recognized by the tilde-sign), and the [n]-sound disappears.

In the final stage of the conversion to phonemes, parentheses are added to the pronunciation and a default stress is given to the first syllable of the word unless this syllable contains a schwa-nucleus. In that case the second syllable gets lexical stress. One could also assign lexical stress to one of the last three syllables (e.g., the penult syllable), counting from the right edge of the word, which is quite common in Germanic languages like English and Dutch (Kager, 1989; Trommelen & Zonneveld, 1989). Nevertheless, I chose for the first syllable having lexical stress (seen from the left edge of the word), guided by the lexical stress rule in the "Frisian Reference Grammar" (Tiersma, 1999).

Duijff (2004) has recently made an overview of the remarks about lexical stress in compounds made in Hoekstra (1998). These remarks are used by the staff of the Fryske Akademy as guideline to assign lexical stress in the "Frysk Hânwurdboek". This overview is very interesting and useful, but since it looks at word classes and pre- and suffixes, it is difficult to program in FRYSS at the moment. This is true as well for the article of Hoekstra (2002) which (as a side-issue) deals with the stress marking in genitive compounds.

The result of the letter-to-sound rule file has the same format as the pronunciation part of the word entry of the compiled lexicon. In example (13) one can see the output of the letter-to-sound file for the word "hynder" (horse).

(13) ((((h i n) 1) ((d & r) 0)))

Here one can clearly see the outer pair of parentheses which seem redundant, because the pronunciation part is already enclosed in a pair of parentheses. However, these parentheses are of great importance. Without them, the program would not built the word properly, since it builds a prosodic tree in several layers, where most layers match the content of a deeper layer pair of parentheses (see B.5.7.3.). If the outer pair of parentheses would miss, an error message would be the output of the system.

It is also possible to construct letter-to-sound rules automatically. Black & Lenzo (2003) give instructions how to do this. In the NeXTeNS-version for Dutch, the TreeTalk method was used to create such rules. TreeTalk is a self training method which can be trained on a set of samples. Since TreeTalk needs more than a hundred thousand words with pronunciation and since our dictionary "only" contained about 63,000 words it was decided to use hand-written rules. Moreover, the hand-written rules were already available.

When there are no letter-to-sound rules, and a word does not occur in the lexicon, Festival can give as a feedback that it does not know the word or it can spell the word out. The recipe for this is given in Black & Lenzo (2003:64). This recipe is not implemented, because FRYSS does use letter-to-sound rules.

B.5.7.3. Building of the prosodic word *(fy::build_prosword_structure* in *net_fy_lex.scm)*

In this section the word is built up from the top down in the Prosodic Tree relation, in short ProsTree. This master relation consists of several relations, or levels, viz. ProsWord1, ProsWord2, Foot, Syllable, SylPart and Segment. The whole word is set as item of the relation of ProsWord1. Next, when the word concerns a compound, the different parts are set in the ProsWord2 relation. Compounds are distinguished in the second pair of parentheses in the pronunciation part. For Frisian such information is not available, so in FRYSS the word in the relation of ProsWord2 is equal to the one in the relation of ProsWord1. Thirdly, feet are defined with a metrical feature "strong" or "weak", depending on the number of syllables before the stressed one. All the data collected at this level are stored in the relation Foot. Then, the syllables are divided in the relation Syllable. After that, the syllables are divided in "Onset" (which contains all consonants before the vowel), "Nucleus" (which contains the vowel) and "Coda" (the latter consonants of the syllable). When defining the Nucleus, phone-mapping to more than one vowel also takes place (see B.5.7.5.). Therefore, in this Frisian TTS system it is possible that the Nucleus contains more than one vowel. The data of Onset, Nucleus and Coda are stored in the SylPart relation. In the final stage, in the relation of Segment, the segments (sounds) are defined. One can see the different items and its features of, e.g., the relation Segment, after synthesizing an utterance (e.g., "hynder" (horse)) as:

```
festival> (utt.relation.print (SayText "hynder")  'Segment)
```

To view the items of another segment replace "Segment" by the name of another relation. See also chapter 3 and B.5.5.

B.5.7.4. Associating tone elements to syllables *(fy::associate_int_to_syls* in *net_fy_lex.scm)*

After building the prosodic word, the ToDI-values and boundary tones (assigned to the words, see B.5.6.) are associated to certain syllables. An initial boundary tone ("%L", "%H"), is associated with the first syllable of the first word of the phrase. Likewise, the final boundary tone ("L%", "H%", "%") is linked with the last syllable of the last word of the phrase. In case of a ToDI-value, lexical stress has to be assigned to the stressed syllable (the one with "1" in the pronunciation part of the lemma, see also B.5.7.1. and B.5.7.2.).

These values and boundary tones are set in the new relation Syl-Int. The new relation Word-Pros links the correct syllable (stored in the relation Syllable) to its ToDI-value or boundary tone.

The original source code of Marsi & Kerkhoff (2003) has not been changed (except for the definition of the Nucleus, see B.5.7.5.), because literature did not mention a difference in the prosodic word for Frisian, compared to Dutch.

B.5.7.5. Phone mapping of complex sounds *(net_fy_lex.scm)*

Frisian counts several long vowels, many more diphthongs, a few triphthongs, and of all vowels there exists a nasalized counterpart, which, unfortunately enough, all lack in Dutch. This means that these vowels should be mapped to its components, i.e., split into two or more vowels. These vowels are selected from the Frisian phoneme set, and annotated in Worldbet. Later, almost at the end of the synthesis process in the Postlexical Module, the Frisian Worldbet phonemes are mapped to their closest Dutch relatives (see also B.5.9. and appendix C), so that the correct diphones from the Dutch MBROLA nl3-voice will be activated.

For example, the diphthong [i&], in the word "liet" [li&t] (song), should be mapped to [i] and [&], see example (14).

(14) [l i& t] ➔ [l i & t] in definition of Nucleus (*net_fy_lex.scm*)
[l i & t] ➔ [l i @ t] in phone mapping section of *net_fy_postlex.scm*

This is done by setting [i&] to [i] (resulting in [l i t]), and inserting the schwa [&] after the [i] (result: [l i & t]) in the phone mapping section of *net_fy_postlex.scm*. Unfortunately this did not work, because the inserted sound (schwa) had not been present in the process of building the prosodic word (Word Module) and in this way it was not inserted in the ProsTree-relation properly. After several wasted efforts of appending this insertion to the ProsTree as yet, it became obvious that the substitution of the complex vowels had to be moved closer to the Word Module (i.e., before the building of the prosodic word). In the first instance, this form of mapping only succeeded immediately after building the word with letter-to-sound rules. Unfortunately, no mapping of this kind took place when the word was built up by the pronunciation lexicon. It proved to be impractical to convert the vowels just after the grapheme-to-phoneme conversion (both letter-to-sound and lexicon), because the vowels were located inside complex bracketed structures. Therefore, the conversion was incorporated into the Nucleus definition of the prosodic word. Here, the process of mapping complex vowels to several vowels by inserting an extra vowel (and in some cases consonants, like with nasalized vowels who are mapped to their unnasalized part and [n]) was successful. All diphthongs were constructed this way, except for [I&]. An example of a word with this diphthong is "each" [I&x] (eye). The combination of [I] and [&] resulted in an output which sounded a lot like [Ib&]. To prevent this output a glide [j] was placed between the two parts. Though, only vowels were defined as Nucleus. As soon as an consonant followed the vowel (in this case when dealing with diphthongs and nasalized vowels: vowels) of the Nucleus, this was defined as Coda. In case of nasalized vowels (that are mapped to their unnasalized counterpart and [n]), this was not a problem, because the [n] came just before the consonants of the Coda. In case of this newly created diphthong [I j &], the [I] was defined as Nucleus and [j] and everything that followed in that syllable was defined as Coda. To make sure that the program inserted all three sounds in the Nucleus, this [j] was called [Q], an imaginary vowel. This imaginary vowel first had to be defined in the Frisian phone set (viz. *net_fy_phones.scm*). In case of the word "each" (eye), the pronunciation was now annotated as [I Q & x]. In the phone mapping section of the Postlexical Module (see also B.5.9.) it was mapped to [j] again. So, the resulting phoneme string concerning this diphthong was [I j & x] again.

The triphthong [U&_i], like in the word "moai" [mU&_i] (beautiful), was also not mapped to its three components, because a glottal stop was heard. Also the combinations of [U] + [>i] (for the word "moai" resulting in [m U >i]), or [U] + [&] + [i] (for the word "moai" resulting in [m U & i]), were no success, as the glotal stop was still present in the output. So, this triphthong was matched to the Dutch diphthong [oi] (SAMPA-notation) in the mapping section of *net_fy_postlex.scm*. The other triphthongs (see Appendix A) are built up from their components. In FRYSS they are not specified as triphthong-sounds. As mentioned earlier, the nasalized vowels are substituted by their original non-nasal equivalent. To prevent losing all nasality in the output an [n]-sound was inserted after the equivalent.

Due to the sticking process of those vowels, all diphthongs now have a duration which is basically a bit too long, viz. the duration of the two components. Because the diphthong [I&] (mapped to [I Q &]) sounded extremely long, the [Q] in the middle was shortened by half. Even then the duration is actually still too long. Unfortunately, there was not enough time to correct the duration of these diphthongs.

To preserve clarity, I want to point out once again that during the synthesis process the system works with the Frisian phonemes (which are Worldbet annotated). This means that mapping of complex sounds in the definition of the Nucleus in the *net_fy_lex.scm* file, is done to its Frisian (Worldbet) components. It is not until the one-to-one mapping at the bottom of the *net_fy_postlex.scm* file that the Dutch phonemes (SAMPA-notation) come in action. Here, the Frisian phonemes whose Dutch counterparts have a different annotation in SAMPA (after all, it is unnecessary to map to the same annotation), or that do not exist in the Dutch database are mapped to their closest Dutch relative. So, in example (14), only the schwa [&] of the word "liet" (song) [l i & t] is actually mapped in *net_fy_postlex.scm* to its Dutch counterpart: [@], resulting in [l i @ t]. From this point (i.e., the phone mapping section in *net_fy_postlex.scm*) on, the system works with these Dutch (SAMPA) phonemes, see also B.5.9. and appendix C.

B.5.8. Pauses Module *(net_fy_pauses.scm)*

In the Pauses Module, the actual pauses are inserted in the phoneme-string. A silent segment is inserted at the beginning and end of the utterance, and wherever a heavy or medium pbreak (see B.5.4.) is given. These silent segments are stored as feature of break in the Segment relation. Be aware of the difference between the pbreak and the break here. Pbreak is a feature which is stored with the words in the Word relation. Break (without p!) is a feature that is clustered to the segments from the Segment relation.

In the Duration Module, the duration of both breaks have a different but fixed value, so one can now finally hear the real difference between a heavy and a medium break (until this point heavy and medium break were treated the same). The original code of this file, written by Marsi & Kerkhoff (2003) has not been changed.

B.5.9. Postlexical Module (*net_fy_postlex.scm*)

Postlexical rules are required when assimilation occurs inside words or between word boundaries.

The original Dutch assimilation rules concerned:

- fricative rule: devoicing of fricative, when the previous sound is voiceless;
- coda devoicing: final devoicing;
- regressive assimilation: devoicing of fricative or plosive, when the next sound is voiceless. This only concerns the conversion to unvoiced segments. In this section, a small bug was found, which was already corrected in the new and not yet released version of NeXTeNS. The bug concerned the fact that plosives were not inserted in the if-clause;
- n-deletion: when [n] is part of the word-final coda and the word-final nucleus is schwa, [n] is deleted.
- cc-deletion: whenever two of the same consonants occur, one is deleted;
- remove empty SylParts: if an empty SylPart was detected, it was deleted (This rule was double coded twice in the beta-release, one can be left out);
- insert a glottal stop in some cases where two nuclei in a row occur.

All but one of the Dutch assimilation rules remained active. The rule of n-deletion was changed, because in Frisian one does not delete the [n] in this situation, but the schwa. This process is called syllabification of [n] (Tiersma 1999), since the [n]-sound becomes a syllable on its own. This schwa-deletion caused several output errors, because the Dutch nl-3 voice was not prepared for some of the now resulting diphones, i.e., the diphones /j-n/, e.g., in the gerund "beljen" [bEljn] (calling) (schwa deleted between [j] and [n]), and /z-n/, for example the plural "hazzen" [hAzn] (hares) (schwa deleted between [z] and [n]).

Since these diphones were not available in the database, the output would simply stop. Another problem was the sometimes unnatural sounding of the [n], like it was chopped off. To avoid these problems, the schwa was inserted again, and then the duration of schwa was shortened. The result of this adaptation sounded better, though the schwa could not be shortened too much, because then it would sound too short and unnatural again. So, this solution is far from optimal. Therefore, a Frisian diphone set would be the right answer to the problems.

Syllabification not only occurs with words ending in [&n], but also with words ending in schwa and [l], [r], [m], or [N]. Examples of such words are:

(15) winkel [V|INk&l], or [V|INkl] (store)
(16) hammer [hAm&r], or [hAm&r] (hammer)
(17) beammen [bjEm&n], or [bjEmm] (trees)
(18) ringen [rIN&n], or [rINN] (rings)

Also in these cases, schwa is shortened and the following sound is lengthened. In some cases (see example (17) and (18)), the following sound has even been changed, due to assimilation processes.

Another assimilation rule which was inserted but later ruled out again, concerned palatization of [s] and [z], whenever they are followed by [j]. I am not sure whether this form of assimilation happens all the time, or that it is idiosyncratic. This feature has not been investigated properly yet.

Finally, in this module the phones were mapped to the Dutch SAMPA phones of the nl3-database. Until now the system worked with the Frisian phones (in Worldbet-annotation). If this mapping would be skipped, some of the phones would not be recognized because of a difference in annotation. Mapping took place to a sound that is the closest Dutch relative. This mapping only succeeded when it concerned one-to-one-mapping. The mapping of one-to-more vowels was done earlier on in the *net_fy_lex.scm* file (to Frisian phones), at the place where the Nucleus was defined. One extra mapping was done, by inserting the mapping of the imaginary [Q]-sound to [j] (see also B.5.7.5.).

The original *net_nl_postlex.scm* file of NeXTeNS ended in the definition of *net_nl_create_diphthongs*, which could also be deleted, because this section was rewritten earlier in this file. This coding was redundant.

B.5.10. Duration Module (*net_fy_dur_kun.scm)*

In the *net_fy_dur_kun.scm* file, the duration of every phoneme is defined. Special rules can shorten or lengthen this default duration, e.g., when it concerns a consonant cluster, the duration of these consonants are shortened. The source code of this file remained the same, though a few small commands were inserted. One of these commands concerned the shortening of the schwa, when the syllabification rule (see B.5.9.) was applied, and the shortening of the [j]-sound in the diphthong [I&] (i.e., the imaginary [Q]-sound). The [n]-sound of the syllabification-rule was

lengthened, to get a better impression of syllabification. I doubt if this was successful, because I do not hear much of a difference myself. Further, the long vowels [i:], [u:] and [y:], were lengthened here. At first, to obtain these sounds, their short relatives were doubled, but then a glottal stop would occur. The lengthening solution provided the best output.
De Graaf (1985) has estimated the duration of long and short vowels of Frisian, but since this was done in CVC-context (isolated words), these durations were not used in this TTS system.

One can also choose the default Festival option concerning the duration of phones, where every phone has the same length. This can be done in the *net_fy_ib_mbrola.scm* file under Duration Module. But we did not use this default, since the *net_fy_dur_kun.scm*, also used by NeXTeNS, provided a better, more natural output.

B.5.11. Fundamental Frequency Control *(net_fy_int_todi.scm)*

In this module the fundamental frequency, F0, is assigned to the utterance. There has not been done much research on Frisian intonation. Most of the literature claim that the intonation patterns of Dutch and Frisian are the same (Cohen et al., 1961; Tiersma, 1999). The study of Hoekstra is more or less done in the field of sentence accent. This study claims that lexical and specific functional prepositions are more often stressed in Frisian than in Dutch, and less often than in English (Hoekstra, 1991).

Since the intonation structure for Frisian is said to be the same as for Dutch, I chose to use the ToDI-intonation, which is used in NeXTeNS, for Frisian as well. ToDI stands for "Transcription of Dutch Intonation" (Gussenhoven et al., 2003). In the Tune Module (see B.5.6.), boundary tones and pitch accents were attached to the utterance. In *net_fy_int_todi.scm* these tones and accents are processed further into an actual intonation contour. In the *net_fy_int_todi.scm* file, all possible boundary tones and pitch accents are available, but not all tones and accents will be reached, due to previous coding.
What follows is a brief description of the boundary tones and some of the pitch accents used in NeXTeNS. More information about ToDI can be found at http://todi.let.kun.nl/ToDI/home.htm in the form of an interactive course.

Speech can be split up in several parts, which in some way reflect the sentence structure. These parts, called intonational phrases (IPs), are audibly separated by means of a (brief) pause, a relatively long syllable before the end of a phrase, or a melodic feature, or a combination of these (Gussenhoven et al., 2003). By means of ToDI, so-called boundary tones are assigned to the left and right edge of each IP, depending on the way they are realized.

The next realizations are possible boundary tones at the beginning of an IP (Gussenhoven, 2004:7):

Initial boundary tones: %L[17]
%H
%HL

Here, "%L" is the neutral option and stands for a mid or low pitch. "%H" is high pitched and not uncommon before low pitched accents, and "%HL" is a rare, highly marked falling pattern. In *net_fy_tune.scm* (see also B.5.6.) "%L" is assigned to the beginning of each IP as a standard value. So the two latter patterns, "%H" and "%HL", though available in the *net_fy_int_todi.scm* file, cannot be reached because the previous coding (in *net_fy_tune.scm*) does not account for "%H" and "%HL" (yet).

Further, there are two final boundary tones for the right edges of the IP:

Final boundary tones: L%
H%

If "L%" occurs, the utterance will end in a low pitch. The same accounts for "H%", where the utterance ends in a high pitch. A third option is when one hears a pause and the fall/rise is half-completed, i.e., when it sounds as if the speaker is not finished yet. In that case, there is no boundary tone, which is marked as "%". In the NeXTeNS and FRYSS this latter value is assigned wherever a medium or heavy break occurs inside the utterance. Only when this break is caused by a question mark, FRYSS selects "H%" as final boundary tone, otherwise "L%" is chosen. NeXTeNS selects only "L%" as standard value.

[17] L stands for Low, and H for High.

The so-called pitch accents regulate the accentuation. Pitch accents are melodic elements, which can occur on one or more words in the IP (Gussenhoven et al., 2003). These accents are placed on the words with sentence accent, in FRYSS this involves content words, or the words that got a "+" as accent feature in the Intonation Module. There are several options for pitch accents available in the file of *net_fy_int_todi.scm*. I will only treat those pitch accents that can be reached via the previous coding. For more information about certain pitch accents, I refer to the website of ToDI (Gussenhoven et al., 2003) or Gussenhoven (2004).

Pitch accents[18]: H*L
L*H
H*

The option of "H*L" consists of a sudden high pitch on the stressed syllable, immediately followed by a fall to low on the next syllable. Likewise is the pitch accent "L*H" an accent with a low pitch, followed by a rise to high. The "H*" accent just consists of a high pitch, it lacks an immediate fall to low. The pitch accents determine, together with the boundary tones the intonation contour and are discussed further in Gussenhoven (2004) and at the ToDI-website http://todi.let.kun.nl/ToDI/home.htm (Gussenhoven et al., Terken, 2003).

The source code of the *net_fy_int_todi.scm* file (Marsi & Kerkhoff, 2003) has not been changed. The implementation of the scaling of the F0 targets is done according to the model described in (Van der Berg et al., 1992).

B.5.12. Waveform synthesis *(/lib/mbrola.scm)*

Finally, the phoneme string and all other information has to be sent to the MBROLA nl-3 voice. In this module the *mbrola.scm* file in the /lib-directory is loaded. This file offers standard support while using MBROLA within Festival. It leads the way to the MBROLA-program, and the location of the nl3-database.

As indicated before, there is no diphone database for Frisian yet. This problem is intercepted by using the Dutch nl3-voice and by mapping the Frisian sounds to their closest Dutch relative. This

[18] Only those that can be reached in NeXTeNS and FRYSS.

happens in two stages. In the definition of the Nucleus in the file of *net_fy_lex.scm*, complex sounds are mapped to the monophthongs they exist of (see B.5.7.). These monophthongs are still Frisian phones. It is not until the bottom of the *net_fy_postlex.scm* file, that the Frisian sounds are mapped to their Dutch closest relative. Therefore it is important that the Dutch phones are still available in *net_fy_phone.scm* as well, otherwise one would get an error message during synthesis.

Of course, when synthesizing text FRYSS will run into diphone-combinations which do not exist in the nl3-database. When it reached the missing diphone, the system gives an error message and the synthesis is stopped. Unfortunately, there is no solution for this problem. Meanwhile work has started at the Fryske Akademy to create a Frisian diphone set, so it seems unnecessary to find a way out of this dilemma.

Appendix C

Mapping Table

[-] means no change in annotation, no mapping. Same phone(s) is (are) used.

Frisian phones **(Worldbet-annotation)**		**Dutch phones nl3-MBROLA database** **(SAMPA-annotation)**	
Frisian phones as defined in *net_fy_phones.scm*	**phone mapping in definition of Nucleus (*net_fy_lex.scm*)**	**phone mapping in *net_fy_postlex.scm***	**Dutch phones as sent to nl3-voice**
consonants			
p	-	-	p
b	-	-	b
t	-	-	t
d	-	-	d
k	-	-	k
g	-	-	g
f	-	-	f
v	-	-	v
s	-	-	s
z	-	-	z
S	-	-	S
Z	-	-	Z
x	-	-	x
G	-	-	G
h	-	-	h
m	-	-	m
n	-	-	n
n~	-	-	J
N	-	-	N
l	-	-	l
r	-	-	r
V|	-	w	w

w	-	-	w
j	-	-	j
vowels			
&	-	@	@
i	-	-	i
I	-	-	I
E	-	-	E
y	-	-	y
Y	-	-	Y
A	-	-	A
u	-	-	u
U	-	O	O
>	-	O	O
long vowels			
i:	-	i (+long)	i
e:	-	e	e
E:	-	-	E:
y:	-	y (+long)	y
7:	-	2	2
a:	-	a	a
u:	-	u (+long)	u
o:	-	o	o
>:	-	O:	O:
8:	-	9:	9:
dipthongs/tripthongs			
i&	i &	i @	i @
iu	i u	-	i u
I&	I Q &	I j @	I j &
Ei	-	-	Ei
>i	-	Oi	Oi
Au	-	-	Au

y&	y &	y @	y @
Y&	Y &	Y @	Y @
8y	-	9y	9y
u&	u &	u @	u @
ui	-	-	ui
Ui	-	Oi	Oi
U&	U &	O @	O @
U&_i	-	oi	oi
nasalized vowels			
&~	& n	@ n	@ n
i~	i n	-	i n
I~	I n	-	I n
E~	E n	-	E n
y~	y n	-	y n
Y~	Y n	-	Y n
A~	A n	-	A n
u~	u n	-	u n
U~	U n	O n	O n
>~	> n	O n	O n
nasalized long vowels			
i:_~	i n	i (+long) n	i n
e:_~	e: n	e n	e n
E:_~	E: n	-	E: n
y:_~	y: n	y (+long) n	y n
7:_~	7: n	2 n	2 n
a:_~	a: n	a n	a n
u:_~	u: n	u (+long) n	u n
o:_~	o: n	o n	o n
>:_~	>: n	O: n	O: n
8:_~	8: n	9: n	9: n
nasalized dipthongs/tripthongs			

i&_~	i & n	i @ n	i @ n
iu_~	i u n	-	i u n
I&_~	I Q & n	I j @ n	I j @ n
Ei_~	Ei n	-	Ei n
>i_~	>i n	Oi n	Oi n
Au_~	Au n	-	Au n
y&_~	y & n	y @ n	y @ n
Y&_~	Y & n	Y @ n	Y @ n
8y_~	8y n	9y n	9y n
u&_~	u & n	u @ n	u @ n
ui_~	ui n	-	ui n
Ui_~	Ui n	Oi n	Oi n
U&_~	U& n	O @ n	O @ n
U&_i~	U&_i n	oi n	oi n
extra phones			
&:	&	@ (+long)	@
I:_~	I: n	I (+long)	I n
Q	-	j	j

Appeared in: Proceedings SSW5, Pittsburgh PA, USA (2004)

FRISIAN TTS, AN EXAMPLE OF BOOTSTRAPPING TTS FOR MINORITY LANGUAGES

Jelske Dijkstra, Louis C.W. Pols and R.J.J.H. van Son

Chair of Phonetic Sciences, University of Amsterdam

ABSTRACT

A Frisian adaptation of a Dutch TTS system based on Festival, NeXTeNS, is presented as a case study in prototyping TTS for resource-poor minority languages. For these languages, demonstrator systems are essential to seed projects in speech and language technology. The conversion of a Dutch TTS system to a new language with minimal speech and language resources, Frisian, demonstrates that a TTS prototype can be built rapidly using existing modules and voices. An informal evaluation with native speakers of Frisian shows that such a hybrid prototype can already produce intelligible speech for demonstration purposes.

1. INTRODUCTION

A shared language is a strong binding force for communities. In the modern world, people often feel that the future of their community is linked to the future of their language (even when this is absurd, see http://www.usenglish.org/ and many others). On the other hand, the prospects of any language depend largely on its sphere of usage. Whenever a language is excluded from a domain of life, it becomes less attractive to its users. Once these exclusions progress, a language will eventually disappear, often together with the community and its defined and valuable cultural heritage.

By definition, minority languages are excluded from large domains of society. So it is no surprise that communities fight to claim as much territory as possible for their shared tongue. Focal points in their political actions are teaching and access to mass media, e.g., TV, radio and newspapers, in the native language.

With the computerization of modern societies, digital media have rapidly become mass media themselves. Exclusion from these digital media and services would be a major setback for any language community. A lot of work has been done on the creation of authoring tools (e.g., spelling and grammar checkers) and localization of digital interfaces (e.g., non-western writing systems). Currently, the localization of a full toolset for digital media is rather straightforward (e.g., http://www.kyfieithu.co.uk/ for Welsh, see also http://l10n.openoffice.org/localization_responsibilities.html). The Simputer project in India (http://www.simputer.org) and the African Speech Technology project (http://www.ast.sun.ac.za/the_project.htm) have demonstrated the importance of a fully integrated speech interface for minority languages. If community members cannot use their own language for ever more ubiquitous speech-related services, both for commerce, mass media and in teaching, this will be a disincentive for the language itself. Moreover, it will strengthen often existing feelings that their language is inadequate for the modern age.

Many communities speaking a minority language do have access to some, limited, resources for technology projects. What these resources have in common is their unpredictability and intermittence. To have any chance of success, implementing a large language application for a minority language has to be divided into small, incremental sub-projects that can be handled by small groups of volunteers or single researchers over a short time-scale. To access these resources, it is important to have an example prototype that can demonstrate the feasibility of the project. Even with a limited prototype, members of the target community can estimate the costs and benefits of a full scale system and decide whether they want to participate. This holds equally well for community volunteers as for grant agencies that try to stimulate the use of the language.

In this paper we present the results of a case study into a rapid prototyping framework for building a TTS system for a minority language, Frisian, with only minimal digital resources. First results of an evaluation of the synthesis quality are given. This study was performed as a MA-thesis of the first author who is a native speaker of Frisian. It is our intention to release the Frisian adaptations as Open Source.

2. THE FRISIAN LANGUAGE

When we speak of Frisian in this paper, we mean West-Frisian, mainly spoken in the province of Fryslân, one of the twelve provinces of the Netherlands. The Frisian language is a member of the West Germanic branch of the Indo-European language family. Several parallels have been found between Old-Frisian and Old-English, though nowadays Frisian tends to become more and more similar to Dutch [2].

2.1. Frisian and Fryslân

The total population of the province of Fryslân counts over 634,000 inhabitants, which is less than 4% of the total population of the Netherlands. Of those inhabitants 74% is able to speak Frisian. For 55% of the total population Frisian is their mother tongue, which comes down to roughly 350,000 native speakers [7]. Furthermore 94% of the population of Fryslân can understand Frisian, 65% can read and 17% can write in Frisian [5]. Language surveys from 1967, 1980 and 1994 show a small decline in the ability to speak Frisian. Also, the Frisian language becomes gradually more and more similar to Dutch due to language assimilation [2]. Our prospects are that both the decline in number of speakers and the assimilation will continue in the future.
Fryslân was traditionally an agricultural area with little industry which induced work-related emigration of younger people. This explains why the education level and income of the Frisian population is below average compared to the rest of the Netherlands. Recently there has been an increase in service-related (financial) industry which might reverse this trend [6].

2.2. Dialects

There are three main dialects of Frisian: Klaaifrysk, Wâldfrysk, and Súd-Westhoeksk [6] and several smaller dialects, mostly mixtures of Dutch and Frisian. In general, all dialect variants are mutually comprehensible. The accepted standard Frisian language is mostly based on the Klaaifrysk forms of Frisian.

2.3. Domains

In 1995 there has been a socio-linguistic survey [5] which concluded that family, work and the village community are the strongest domains for Frisian.
Since its recognition in 1970 by the Dutch government, the position of Frisian has improved, although slowly. Now, for example, Frisian has equal goals in education as for Dutch, and it is allowed to use Frisian in court and in the correspondence of public administrations. Though the amount of Dutch used in those formal domains is still considerably larger [2].
There are two daily newspapers in Fryslân, which produce < 3% Frisian texts and one special Frisian page every week. Furthermore there is a small number of Frisian (literary) journals and magazines [6]. Together, these give only a limited amount of digital text to work upon for language technologies.

3. CHANGING AN EXISTING FESTIVAL TTS SYSTEM TO PROCESS A NEW LANGUAGE

The approach we chose for rapid prototyping was to take an existing implementation of the Festival TTS system and adapt it piecewise to generate Frisian speech. Given historic influences, we chose to use a Dutch implementation of Festival, NeXTeNS [10], which was adapted to process Frisian instead of Dutch text.

3.1. Festival and the NeXTeNS-project

The Dutch NeXTeNS project aimed to produce a Dutch TTS for research purposes [10]. NeXTeNS is built upon the common Festival system. The waveform synthesizer operates on the MBROLA diphone synthesizer and it uses the Dutch nl3-voice. It is freely available for research purposes.

The architecture of NeXTeNS is derived from the standard Festival system architecture:

- Token Module: tokenisation
- POS Module: Part-Of-Speech tagging
- Syntactic Module: syntax parsing
- Phrasing Module: phrase break prediction
- Intonation Module: accent placement
- Tune Module: tune choice needed for ToDI
- Word Module: lexicon, letter-to-sound rules, building prosodic structures
- Pauses Module: pause insertion
- Postlexical Module: assigning postlexical rules and phone mapping
- Duration Module: determination of segment and pause durations
- Fundamental frequency control: apply ToDI to utterance
- Waveform synthesis: sending TTS-information to MBROLA-voice

For our Frisian prototype TTS system, many of the advanced features, e.g., POS tagging, NP chunking and ToDI labeling, are not available as they could not be re-trained for Frisian without adequate training corpora.

3.2. Language resources and tools

For Frisian as an official language of the Netherlands, there exists a language research infrastructure. Most research for Frisian has been coordinated and hosted by the Fryske Akademy ("Frisian Academy"). The linguistic information needed for creating letter-to-sound rules and intonation and duration modules was largely provided by the Academy.

There are several associations that provide language information and resources. LDC (http://ldc.upenn.edu) and ELRA (http://www.elra.info) distribute large annotated corpora, and are parent associations for lots of initiatives (e.g., the LREC conferences). Cocosda (http://www.cocosda.org) tries to coordinate language resources and tools. The IMDI-project of EAGLES/ISLE collects data on existing corpora (http://www.mpi.nl/IMDI/ and/or http://www.mpi.nl/ISLE/). Organizations working on minority and endangered languages are SALTMIL (http://isl.ntf.uni-lj.si/SALTMIL/) and DOBES (http://www.mpi.nl/DOBES). Other initiatives are the Foundation for Endangered Languages (http://www.ogmios.org), the Endangered Language Fund (http://sapir.ling.yale.edu/~elf/), and

the International House for Endangered Languages (http://www.tooyoo.l.u-tokyo.ac.jp/ichel/ichel.html).
Many voices for speech synthesis are available on the MBROLA website (http://tcts.fpms.ac.be/synthesis/mbrola.html).

4. STEP BY STEP PROCESSING

4.1. Phoneme set

First of all, a computer-readable phoneme set was created. For Frisian we created a phone set based on the SAMPA set used by the Fryske Akademy. However, instead of SAMPA we used the Worldbet-annotation [8] because it codes each IPA symbol uniquely and over all languages. Moreover, Worldbet allows transparent coding of complex sounds (e.g., triphthongs, nasalized diphthongs) and transitions between narrow and broad transcriptions. For Frisian this was needed when dealing with nasalized vowels (e.g., nasalized diphthongs) and triphthongs, who go beyond SAMPA's two characters codes.

These Frisian phonemes were inserted between the Dutch ones in the phoneme file in NeXTeNS. Because we continued using the Dutch voice (see also 4.13.), it was important that the Dutch phonemes remained in the phoneme file. At the bottom of the file with postlexical rules, the Frisian phonemes were mapped to their Dutch counterparts. So, if these Dutch phonemes were absent in the phoneme file, Festival would give an error.

After referring NeXTeNS to use an empty lexicon and letter-to-sound-rules file, a basic synthesizer was created.

4.2. Token Module

Tokenisation is necessary to change unknown tokens like abbreviations, numbers, symbols, acronyms and dates into words. A standard file in NeXTeNS was completed with the language-specific details. To avoid most problems we only implemented abbreviations (from an older Frisian-Dutch dictionary [14]) and a number-to-word conversion. The latter was done by copying the number-to-word conversion for the Spanish el-voice and by changing the order of pronunciation to the order in Frisian (and Dutch). For instance, instead of converting the number "31" to "treinta y uno" (lit. "thirty and one") as in Spanish, it was converted to the Frisian "ienentritich" (lit. "one-and-thirty").

Due to lack of time less attention has been paid to symbols, acronyms and dates. Examples of these implementations are given in the English version of Festival, though. This version contains a huge variety of token-to-word conversions.

4.3. POS Module

Part-of-Speech tagging is mainly used for accent and break assignment. Since there is no Part-of-Speech tagging for Frisian, we decided to make use of the simple function and content word division by using the guess_pos-function. Hence, the automatic POS tagging function was not operational in NeXTeNS (at the time of writing), so a separate list of function words was made by copying the function words from a Frisian grammar [13] and by inserting translated missing words from a Dutch function word list [12]. Both guess_pos-list and this separate list of function words were located in the tokenisation file.

An alternative for creating a Frisian POS file would be to translate a Dutch one into Frisian.

4.4. Syntactic Module

Since there is no syntax parser for Frisian the default option of no syntax method was chosen.

4.5. Phrasing Module

In this module breaks are predicted by means of punctuation. Breaks can be heavy or medium. The default option is a punctuation cart tree, which we chose. Alternatives are assigning breaks by means of POS (if POS-tagging is available).

4.6. Intonation Module

In NeXTeNS nouns, adjectives and verbs (except auxiliary verbs) get sentence accent. Since we used a simple function/content word division, this rule was replaced by one that gives accent to every word that is not a member of the function word list (see also 4.2. and 4.3.). Furthermore in a group of accents every second accent was removed.

4.7. Tune Module

In this module sentence accents and breaks are replaced by ToDI-values, which are necessary for the fundamental frequency control (see also 4.12.). The values %L and L% are assigned to the beginning and the end of each utterance, respectively. In case of a medium or heavy break (see also 4.5.) the module refers to the %-value. Sentence accents are usually replaced by H*L-values. This source code was written by Marsi & Kerkhoff [10]. For more information about these ToDI-values see http://todi.let.kun.nl/ToDI/home.htm. At the time of writing not all options could be reached by the code, because in some cases the POS was needed to assign a ToDI-value, e.g., in the case of H*LH, which was assigned in special cases after a verb.

4.8. Word Module

In the Word module, the graphemic word is transformed into a phonemic one. This happens by means of a pronunciation lexicon. When a word does not occur in this lexicon it is built up by letter-to-sound rules (LtS). After the lexicon lookup or LtS, the prosodic structure of the word is built up.

4.8.1. Letter-to-sound rules

LtS rules can be written by hand, or automatically. In the Frisian language, there is a relatively strong relationship between the letters in a word and its pronunciation. For languages like this it is often easier to write the rules by hand. The LtS can be built from existing examples from the Festival distribution. The Spanish example that we used contained a conversion to lower-case letters, a grapheme-to-phoneme conversion, a conversion into syllables, and a definition for assigning lexical stress to the word in question. The definition to change certain vowels into weaker ones, needed for Spanish LtS, was removed. For practical reasons, syllabification was put before the actual LtS. A separate definition for assigning the nasal feature to vowels was given later on. So first the word was set to lower-case letters, then a division into syllables took place. The hyphen sign was used as symbol for the syllable break. When two identical consonants occurred a syllable break was given between those consonants. When a consonant was surrounded by vowels a syllable break was given before the consonant. Furthermore, all possible consonant clusters were listed [4] together with their breaks. Breaks that occurred at the wrong place were for the most part corrected in the next definition, the actual LtS, in which graphemes were changed into phonemes. Next, a default stress was given to the first syllable of the word unless this syllable contained a schwa vowel. If necessary the feature nasal was assigned to the vowels in question.

The LtS rules have the following form [1]:

(LC [alpha] RC = beta)

Some examples are:

(1) ([y] = i)
(2) (VOWEL [- g] VOICEDC = - G)

Example (1) is a simple LtS conversion The sound [y] is assigned to the letter <i>. In case of example (2) a voiced [G] is given whenever <g> is placed between vowels (left side) and voiced consonants (on the right). As mentioned earlier, the hyphen sign is the annotation for a syllable break.

LtS rules could also be constructed automatically. Black and Lenzo [1] give instructions how to do this. In the NeXTeNS-version with Dutch, the TreeTalk method was used to create such rules. TreeTalk is a self training method which can be trained on a set of samples. Since TreeTalk needs more than a hundred thousand words with pronunciation and since our dictionary "only" contained about 70,000 words it was decided to use hand-written rules.

At the end of the LtS file the word was built up like the pronunciation part of the word entry of a lexicon (see also 4.8.2.). For example, the output of the LtS file for the word "hynder" (horse) looks like this:

```
( ( ( (h i n) 1) ( (d & r) 0) ) )
```

4.8.2. Pronunciation lexicon

However, there are still words with irregular pronunciation, or with an irregular stress pattern. Therefore it is advantageous to use a pronunciation lexicon. In general, if an extensive digital pronunciation dictionary is available, this should be converted to the standard Scheme form. A recurrent problem here is the incompatibility of the phoneme sets used in the dictionary and that necessary for TTS. If necessary, the dictionary transcription has to be “augmented” by special LtS rules to disambiguate the incompatible words. When no digital dictionary is available, it can be built starting with an automatic transcription of a (large) word list with the LtS and syllabification rules. Volunteers can then correct this transcription for known problems and check and correct the rest. Such work can easily be distributed over the internet, and includes proofreading and other management tasks (see the project Gutenberg, http://www.gutenberg.net).

We were fortunate to have access to a digital version of the "Frysk hânwurdboek"-dictionary from the Fryske Akademy, for which we are grateful. The lemma, which contained lexical stress in the form of an apostrophe before the stressed syllable, and its pronunciation had to be converted into a Scheme file. Each word was converted separately with help of a Perl script. First, the phonetic signs used by the Fryske Akademy were replaced by the Worldbet annotations. Then a syllable division took place on the pronunciation. This was based on sonority, which provided a reasonably accurate syllable division. The number of nuclei before the apostrophe in the lemma part were counted and in this way a lexical stress was assigned to the correct syllable in the pronunciation part. As a last step, the apostrophe was taken out of the lemma. Because the dictionary contained only the primary accent placement, our synthesis was limited to primary accents as well. As we followed the NeXTeNS-project and as such the architecture of the KUNLEX-lexicon, we assigned 'nil' to the POS information. One could also assign a Part-of-Speech tag to it, to get a better chance for the correct pronunciation in its context.

A word entry in the final lexicon should contain, next to the orthographic word, POS information, and a phonetic realization of the word in question, including syllable boundaries and lexical stress marking (when appropriate) [1]. The result of the word "bjusterbaarlik" (miraculous) looks like this:

```
("bjusterbaarlik" nil
        ((((b j Y s) 0) ((t & r) 0) ((b a: r) 1) ((l & k) 0))))
```

The lexicon should contain not only the base forms of a word, but all their morphological variants as well. These variants are usually not available in a dictionary. Including all those variants is a large but realistic job. However, it becomes unrealistic when dealing with languages with extensive word compounding or agglutinative languages like Finnish, or Turkish. In that case Black and Kenzo [1] advise to develop a proper morphological analyser to intercept this problem (see also [11]). This was outside the scope of our prototype.

When there is no LtS, and a word does not occur in the lexicon, Festival can give feedback that it does not know the word or it can spell out the word. The recipe for this implementation is found in [1]. Since we do have LtS, this has not been implemented.

4.8.3. Building the prosodic structure of the word

The word was built up from the level ProsWord1 (whole word) down to ProsWord2 (in case of compounds), Foot, Syllable, SylPart (Onset, Nucleus, Coda) and Segment (phonemes) in the relation ProsTree. In NeXTeNS compounds were divided at the level of ProsWord2. For Frisian we were not able to accomplish this, so for Frisian ProsWord2 is equal to the ProsWord1 level. This explains the two pairs of brackets around the whole word in the pronunciation part of the lexicon. All code is implemented by Marsi & Kerkhoff [10].

4.9. Pauses Module

This Pause module inserts the actual pauses. It inserts a silent segment at the beginning and end of the sentence, and wherever the Phrasing Module contains a heavy or medium break.

4.10. Postlexical Module

Postlexical rules are applied for when assimilation occurs between word boundaries and inside words. Also in this module phones are mapped to their Dutch counterparts (see also 4.13.). At the time of writing, we made use of the Dutch postlexical rules. These are mostly the same as for Frisian. Though, we still have to implement some for Frisian as well.

4.11. Duration Module

In this module the duration of every phoneme is defined and special rules can shorten the default duration or lengthen it, e.g., shortening in a consonant cluster. Another option is the default duration module in Festival, where every phoneme has the same length. In our prototype we made use of the duration file as used by NeXTeNS.

4.12. Fundamental frequency control

For F0 assignment NeXTeNS uses the ToDI-intonation (http://todi.let.kun.nl/ToDI/home.htm). Not much research has been done on the prosody and intonation of Frisian. Most grammars assume the Frisian intonation to be the same as in Dutch [4] [13]. One of the few studies on Frisian intonation has been done by Hoekstra [9], who concentrates on sentence accents. He claims that lexical and specific functional prepositions are more frequently stressed in Frisian than in Dutch, and less than in English.

Because of the so-called similar intonation structure in Dutch and Frisian, we used ToDI for the time being and are curious to see if the intonation is good enough for Frisian.

4.13. Waveform synthesis

One of the aims of a TTS prototype system is to create an incentive to construct a language specific voice (diphone set). So no attempt was made yet to create a Frisian diphone-database, the Dutch nl3-database of MBROLA was used instead. The Frisian phonemes were mapped to their closest relatives in Dutch. A similar approach was used by Campbell [3] in creating multilingual TTS. He produced speech in another language (English) than that of the database speaker (being Japanese), though the quality of the resulting speech by mapping alone was not considered good enough. He improved this by using the cepstral information of similar speech of a native speaker of the target language in producing speech with the segments of the prestored voice. For our prototype, this procedure was too involved and was not used.

The Frisian phoneme inventory has more vowels, diphthongs and nasal vowels than Dutch. Most Frisian diphthongs end in a schwa sound. These diphthongs were more or less created by inserting a second vowel (mostly schwa), representing the second part of the diphthong. To allow the correct processing of the inserted segments, this has to be done close to the Word Module, where the grapheme-to-phoneme conversion takes place. All diphthongs except for one (viz. [I&]), are represented in this way. The triphthong (viz. [U>_i]) was not mapped by three phones because this did not improve the quality of the output. Instead it was mapped to the Dutch diphthong [Ui].

Nasal vowels, which also are an important feature in Frisian, are not present in the nl3-database at all. So these vowels had to be restored to their original form again (non-nasal counterpart plus [n]), awaiting a possible Frisian database in the future. In the phone mapping section they were coincided with the nasal again, because otherwise we would loose the nasal aspect in the output; it would sound less like Frisian.

Of course by synthesizing texts the synthesizer will run into diphone-combinations which are not available in the nl3-database. There is no solution for this problem yet and thus an error message will occur.

5. EVALUATION

Eleven native speakers of Frisian were asked to judge 20 sentences, harvested from internet sources as newspapers, party manifestos, internet editions of literature magazines and publications of several youth associations. The subjects had to indicate the intelligibility, general quality and acceptability of the stimuli, each on a 7 point scale. As for acceptability we asked the question whether the synthesized sentences were acceptable as a first attempt for speech synthesis. During this first evaluation, the pronunciation lexicon was not ready, so pronunciation and lexical stress were retained by LtS only. Subjects were informally selected from the contacts of the first author. We want to stress that this is only a pilot study and the results should be seen as indicative only. A formal evaluation is currently prepared.

Three subjects were excluded from the results, because they aborted the test. One of the remaining eight listeners judged only 18 of the 20 sentences in a second attempt. His first trial was not included in the results, because he aborted the test after eight sentences. This means that the total number of responses comes down to 158. The utterance length varied between 9 and 19 words and included Frisian features where synthesis would go wrong, e.g., nasality of vowels (this lacked in the output, see 4.13.), wrong placement of (default) lexical stress (see 4.8.1.), and the feature breaking where vowel change takes place in derived forms, which cannot always be gathered from the spelling.

A division was made between long (>13 w) and short utterances (<=13 w). Both the long and the short set contained 10 stimuli. The averages of the judgements are shown in Table 1.

Table 1: Mean judgements and standard error (between brackets) scale judgements 1-7, higher is better.

	short (N=78)		**long (N=80)**		**total (N=158)**	
intelligibility	3.94	(0.21)	4.00	(0.18)	3.50	(0.14)
quality	3.67	(0.17)	3.78	(0.16)	3.38	(0.12)
acceptability	3.12	(0.16)	3.31	(0.15)	3.13	(0.11)

As expected, the synthesis quality of the Frisian TTS is not stellar. Average judgements are actually below the centre of the scale (4). Six sentences were next to incomprehensible which reduces the scores. The low scores can be attributed to the problems with missing phonemes/diphones and bad modelling of morphological processes. Overall, the fact that the scores are not minimal and even better for some of the utterances shows the potential for improvement, which is the main aim of producing this prototype.

6. CONCLUSIONS

We have demonstrated that it is possible to develop a base-line prototype TTS system for a minority language with minimal speech and language resources. This framework of prototyping TTS allows the fast bootstrapping of speech synthesis. Hopefully, decision makers can then be convinced to spend more money on synthesis. A functioning prototype allows them to estimate the efforts needed for a full scale implementation. Moreover, the organization of the work follows quite logically from the structure of the Festival modules in the prototype.

7. ACKNOWLEDGEMENTS

We would like to thank Erwin Marsi and Joop Kerkhoff for their advice and patience when answering our questions about NeXTeNS. We would also like to thank Hugo Quené for providing us with a Dutch function word list, Dafydd Gibbon for his advices, and Hindrik Sijens, Willem Visser and Durk Gorter from the Fryske Akademy for their advices and for the opportunity to work with the unpublished digital version of the Frysk Hânwurdboek.

Finally, we would like to express our gratitude to the Stichting Spraaktechnologie, the Douwe Kalma Stifting, and the Chair of Phonetic Sciences for their financial support.

8. REFERENCES

[1] Black, A.W. and Lenzo, K.A., *Building Synthetic Voices*, Language Technologies Institute, Carnegie Mellon University and Cepstral LLC, 2003

[2] Breuker, P., *West Frisian in Language Contact*, in *Handbuch des Friesischen,* Munske, H.H. et al., Niemeyer Verlag, Tübingen, 2001

[3] Campbell, N., *Foreign-language Speech Synthesis*, in *Proceedings SSW3*, Jenolan Caves, Australia, p. 177-180, 1998

[4] Cohen, A., Ebeling, C.L., Fokkema, K. and Holk, van, A.G.F., *Fonologie van het Nederlands en het Fries. Inleiding tot de moderne klankleer*, Martinus Nijhoff, 's-Gravenhage, 2nd edition, 1961

[5] Gorter, D. and Jonkman R.J., *Taal yn Fryslân op 'e nij besjoen*, Fryske Akademy, Ljouwert, 1995

[6] Gorter, D., *Extend and Position of West Frisian*, in *Handbuch des Friesischen,* Munske, H.H. et al., Niemeyer Verlag, Tübingen, 2001

[7] Gorter, D. *Nederlands en Fries op gespannen voet?* In *Waar gaat het Nederlands naar toe?* Stroop, J., Uitg. Ben Bakker, Amsterdam, 2003

[8] Hieronymus, J.L., *ASCII Phonetic Symbols for the World's Languages: Worldbet*, AT&T Bell Labs, Murray Hill, USA, 1994

[9] Hoekstra, J., *Oer it beklamjen fan ferhâldingswurden yn it Frysk, it Hollânsk en it Ingelsk*, Us Wurk, Volume 40, p. 67-103, Fryske Akademy, 1991

[10] Marsi E. and Kerkhoff J., *NeXTeNS*, http://nextens.uvt.nl/, 2003

[11] Möbius, B. *Word and syllable models for German TTS synthesis*, in *Proceedings SSW3*, Jenolan Caves, Australia, p. 59-64, 1998

[12] Quené, H. and Kager, R., *PROS*, Research Institute for Language and Speech, Utrecht, 1990

[13] Tiersma, P.M., *Frisian Reference grammar*, Fryske Akademy, Dordrecht Foris Publications, 1985

[14] Zantema, J.W., *Frysk Wurdboek Frysk-Nederlânsk*, 12e druk, Fryske Akademy Ljouwert, O.J. Osinga Uitgeverij, Drachten/Ljouwert, 12th edition, 1992 (1st edition 1984)