The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching

Authors: Marc Schröder and Jürgen Trouvain

Corresponding Author:
Marc Schröder
DFKI
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
Germany

email: schroed@dfki.de
Tel.: +49-681-302-5303
Fax: +49-681-302-5338

The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching

Authors

Marc Schröder
DFKI GmbH, Saarbrücken, Germany

Jürgen Trouvain
Institute of Phonetics, University of the Saarland, Saarbrücken, Germany

Abstract

This paper introduces the German text-to-speech synthesis system MARY. The system's main features, namely a modular design and an XML-based system-internal data representation, are pointed out, and the properties of the individual modules are briefly presented. An interface allowing the user to access and modify intermediate processing steps without the need for a technical understanding of the system is described, along with examples of how this interface can be put to use in research, development and teaching. The usefulness of the modular and transparent design approach is further illustrated with an early prototype of an interface for emotional speech synthesis.

Keywords

Text-to-Speech, Speech Synthesis, Markup languages, Teaching in Speech Technology, Emotions

Abstract

This paper introduces the German text-to-speech synthesis system MARY. The system's main features, namely a modular design and an XML-based system-internal data representation, are pointed out, and the properties of the individual modules are briefly presented. An interface allowing the user to access and modify intermediate processing steps without the need for a technical understanding of the system is described, along with examples of how this interface can be put to use in research, development and teaching. The usefulness of the modular and transparent design approach is further illustrated with an early prototype of an interface for emotional speech synthesis.

1 Introduction

This paper presents the German text-to-speech system MARY (**M**odular **A**rchitecture for **R**esearch on speech s**Y**nthesis) which is a flexible tool for research, development and teaching in the domain of text-to-speech (TTS) synthesis. [1]

MARY allows a step-by-step processing with an access to partial processing results. In this respect, MARY is similar to the TTS system and interface DRESS developed in Dresden (Hoffmann et al., 1999), also for German. However, as the MARY system uses an XML-based data representation, it does not only display the intermediate processing results, but also allows their modification by the user. Thereby, the user is given the opportunity to interactively explore the effects of a specific piece of information on the output of a given processing step.

MARY is composed of distinct modules and has the capability of parsing speech synthesis markup such as SABLE (Sproat et al., 1998). These features are also found in FESTIVAL (Black et al., 1999), an open source TTS system designed for multi-lingual use. The modular design of FESTIVAL allows everybody to write their own modules which can be plugged into the system. For German, a text normalisation and preprocessing module for FESTIVAL is provided by IMS Stuttgart[2] (Breitenbücher, 1999). FESTIVAL is excellent for getting an in-depth understanding of the technical aspects of text-to-speech synthesis. In contrast, MARY provides a web interface accessible from everywhere with no need to install the system locally. This makes it more suitable for those with an interest in the linguistic aspects of the input and output of the individual modules who do not need access to the technical details of the system.

The article is structured as follows. First, the properties of the system-internal XML representation are described. Then, a detailed account of the system structure is given, including a short presentation of each module. After that, the user interface which allows the user to display and edit intermediate processing results is described. Finally, examples are given to show the use of such an interface for teaching, TTS development, and research.

2 The MaryXML markup language

Throughout the MARY system, an internal, low-level markup called MaryXML is used, which reflects the modelling capabilities of this particular TTS system. MaryXML is based on XML (eXtensible Markup Language) (Harold, 1999). A DTD (Document Type Definition) formally specifies the structure of a correct MaryXML document.

As the use of this internal XML language is fundamental for the flexibility of the MARY system, its properties are discussed before the system as such is presented.

2.1 Positioning the markup language

Because of the growing number of XML-based markup languages related in different ways to speech synthesis, it may be necessary to position MaryXML with respect to these existing markups.

One group of markup languages provides relatively high-level markup functionality for speech synthesis input, intended for the use of non-experts. Early examples for this group include the original SSML (speech synthesis markup language, (Taylor & Isard, 1997)) and STML (spoken text markup language, (Sproat et al., 1997)) as well as Sun Microsystems' JSML (Java speech markup language (JSML, 1999)). Out of these, SABLE (Sproat et al., 1998) was developed, for which parsers exist e.g. in the Bell Labs system (Sproat, 1997) and in FESTIVAL. More recent additions to this family of high-level markups are the XML-based markup language coming with Microsoft's SAPI (Speech API) 5 (Microsoft, 2002) and the new W3C SSML (speech synthesis markup language, (Walker & Hunt, 2001)) which is still in draft status. All of these markup languages are, beyond superficial syntactic difference, functionally similar: They aim at giving a non-expert user the possibility to add information to a text in order to improve the way it is spoken. These markup languages are (at least in principle) independent of any particular TTS system. A specific system is assumed to parse the markup language in its input and translate the information contained in it into a system-internal data representation format which in most cases is not XML-based.

A recent addition to the landscape of markup languages, with a huge commercial potential, is VoiceXML (VoiceXML, 2000). This markup language combines parts of the functionality of speech synthesis markup languages such as SABLE with speech recognition, dialogue management and touchtone dialing functionalities. Its main focus is to provide the necessary tools for a speech access to the World Wide Web.

MaryXML belongs to a different category, and might rather be called a representation language than a markup language. Its purpose is to serve as the data representation format inside the TTS system. For that reason, the concepts represented in it are low-level, detailed, and specific to the design decisions, modules, and scientific theories underlying the TTS system. By means of the Document Object Model (DOM), a standardised object-oriented representation of an XML document, the TTS system modules operate directly on the XML document, interpreting and adding information. Currently the MARY system as well as the BOSS system (Klabbers et al., 2001) follow this approach.

2.2 Advantages and disadvantages

The system-internal XML representation enables the MARY system to provide access to intermediate processing results. Technically, this is realised as follows. Through what is called *serialisation*, a standard operation on DOM XML representations, the current state of the document can be made externally visible in the form of a textual XML document at any stage of processing. As the external XML document contains the complete data which was available at the intermediate processing step where serialisation occurred, the inverse process is also possible: *deserialisation*, i.e. a textual XML document corresponding to an intermediate processing step is parsed by the system, and processing can continue from that step onwards. An expert can edit the textual XML document before feeding it back into the system and thus control all aspects of system data.

A second benefit of using a system-internal XML representation is that it is very easy to parse a speech synthesis input markup language such as SABLE, as this amounts to the translation of one XML format into another (see 3.1).

A structural limitation inherent to XML in general should be mentioned. XML documents enforce an unambiguous tree structure, in which one element is always fully embedded in another one. Therefore, it is not possible to represent two non-embedding structures (e.g., syntactic and prosodic structure) simultaneously via structural XML elements. Instead, one of the structures can be represented via hierarchically structured elements while the other structure must be represented in a "flat" form (see 3.4).

2.3 Syntax

The syntax (in a formal technical, not in a linguistic sense) of a MaryXML document reflects the information required and provided by the modules in the TTS system. Those units of information which can also be encoded in speech synthesis input markup languages, such as sentence boundaries and global prosodic settings, are represented by the same tags as in a standard representant of that group of markup languages. At the time of system design, SABLE was chosen as a model for these tags; in the future, if the W3C SSML becomes an established standard, the MaryXML tags should be adapted to their SSML equivalents.

The majority of information to be represented in MaryXML, however, is too detailed to be expressed using tags from input markup languages, for the reasons outlined in Section 2.1 above. Specific MaryXML tags need to represent the low-level information required during various processing steps. This encompasses mainly tokens along with their textual form, part of speech, phonological transcription, pitch accents etc., as well as prosodic phrasing. In order not to clutter up this paper with technical details, only a selection of tags is introduced as the modules requiring them are discussed. A full DTD for MaryXML is available online.[3]

2.4 Future

The emergence of system-internal markup languages in recent systems such as MARY, BOSS and possibly others opens interesting new lines of thought geared towards connecting TTS systems. If it were possible to define at least a minimal standard TTS architecture with clearly defined XML-based data representations at the interfaces, this would open up the possibility to interconnect modules from different TTS systems and thus work towards a "plug-and-play" TTS architecture. Many problems regarding the details of such work can be anticipated, as each system will differ substantially with respect to the types of data represented internally, both fundamentally (e.g., target-based vs. contour-based descriptions of intonation) and in detail (e.g., the tag sets used for part-of-speech annotation). Still, it would seem worthwile to pursue this idea even if only a subset of system-internal information is transferable via such standardised interfaces.

3 Structure of the TTS System

In principle, the modular design of the MARY system allows arbitrary system architectures. An architectural frame is defined via the notion of *data types*, specifying the data format serving as input and/or output to processing modules. Each module knows about the data type it requires as its input and the data type it produces as its output. Two modules can be "plugged" together if the first module's output type is identical to the second module's input type.

Using this frame, an example architecture has been implemented for German TTS. Nothing limits the system from being extended to other languages: It suffices to define new data types corresponding to the intermediate processing steps sensible for that language (e.g., *text, preprocessed, phonemised* and *audio*), and to provide a chain of processing modules connecting these new data types (e.g., *preprocessor, phonemiser* and *waveform synthesiser*).

In the following, the current German TTS architecture within the MARY system is described. Not surprisingly, it is similar to a typical TTS architecture as described by (Dutoit, 1997). Figure 1 shows the individual processing modules, the flow of information and the intermediate results corresponding to data types defining the interfaces between the modules.

— insert Figure 1 somewhere here —

In the following, each of the modules will be briefly presented.

3.1 Optional Markup Parser

The MARY text-to-speech and markup-to-speech system accepts both plain text input and input marked up for speech synthesis with a speech synthesis input markup language such as SABLE.

The input markup language, presently SABLE and the W3C draft version of SSML, is translated by this module into the system-internal MaryXML format, upon which subsequent modules will operate.

As an example, an `<EMPH>...</EMPH>` SABLE tag requesting moderate emphasis for the enclosed words is translated into low-level settings such as, e.g., a raised F0 level, reduced speed, and an obligatory pitch accent for every enclosed word.[4] These settings are expressed in the MaryXML annotation and reflect the capabilities of the following modules to influence the utterance realisation. This module only determines the fact *that*, e.g., a pitch accent must be present, whereas the corresponding specialised module will determine at a later stage *which* accent to realise on that word.

The realisation indications expressed in the input markup are considered as supplements to the modules' text-to-speech analysis of the input. Each module adds new or more detailed information. For example, if the prosody module does not get information from its input on the locations and types of accents and boundaries, it will use its default rules (see 3.6) to determine them. If it finds partial information in its input, such as the location, but not the type of an accent, it will apply its rules to fill in the missing piece of information.

Technically, the markup parser's task of translating one XML format into another is performed using a specialised XSLT (eXtensible Stylesheet Language Transformation) stylesheet (Harold, 1999). This technique allows a very simple adaptation to new markup languages such as the upcoming W3C Speech Synthesis Markup Language SSML (Walker & Hunt, 2001), as only the stylesheet defining the translation into MaryXML needs to be adapted.

3.2 Tokeniser

The tokeniser cuts the text into tokens, i.e. words, numbers, special characters and punctuation marks. It uses a set of rules determined through corpus analysis to label the meaning of dots based on the surrounding context. In order to disambiguate the meaning of dots, which can be sentence-final periods, decimal number delimiters, parts of ordinal numbers, or abbreviation points, the rules collect evidence from the surrounding context on the role(s) which the dot can or cannot fulfill. For example, a dot preceded directly by a number and followed by whitespace and a lower-case character is not a sentence-final period.

Each token is enclosed by a `<t>...</t>` MaryXML tag. All local information about a token determined by subsequent processing steps is added to that token's `<t>` tag as attribute/value pairs. In addition, punctuation signs, including those dots which are identified as sentence-final periods, are used to determine start and end of sentences, which are marked using the MaryXML `<div>...</div>` tag enclosing a sentence.

3.3 Text normalisation

In the text normalisation module, those tokens for which the spoken form does not entirely correspond to the written form are replaced by a more pronounceable form.[5]

3.3.1 Numbers

The pronunciation of numbers highly depends on their meaning. Different number types, such as cardinal and ordinal numbers, currency amounts, or telephone numbers, must be identified as such, either from input markup or from context, and replaced by appropriate token strings.

While the expansion of cardinal numbers is straightforward, the expansion of ordinal numbers poses interesting problems in German, because of their inflections. On the one hand, the expansion of an ordinal number depends on its part-of-speech (adverb or adjective); on the other hand, for adjective ordinals, the inflection ending depends on gender, number and case of the noun phrase which the ordinal belongs to. In the text normalisation module, none of that information is available, so the ordinal number is simply marked as such, and a stem expansion is given. For example, the ordinal "1." would become "erstens" (Engl. "first (adverb)") in adverbial position ("denn 1. ist das...") and "erste/ersten/erstes/erster" in adjectival position. This module adds the information `ending="ordinal"` and `sounds_like="erste"` to the ordinal's `<t>` tag. Based on this markup, the correct ending will be selected during phonemisation (see 3.5.1 below).[6]

3.3.2 Abbreviations

Two main groups of abbreviations are distinguished: Those that are spelled out, such as "USA", and those that need expansion. The first group of appreviations are correctly pronounced by spelling rules.

The second group is pronounced using an expansion table, containing a graphemic and optionally a phonemic expansion. The graphemic expansion is used for abbreviations such as "bzw.", expanded as "beziehungsweise" (Engl. "respectively"), or "BAföG" (a German government scholarship), expanded as "Bafög" and left to be treated by the default letter-to-sound conversion algorithm (see 3.5.3 below). The phonemic expansion is useful for non-standard pronunciations such as "SFOR" (pronounced [ˈɛs-foːɐ]), and for foreign abbreviations, such as "FBI" which is pronounced as the English spelling [ɛf-biː-ˈaɪ] in German.

One group of abbreviations, such as "engl.", pose a problem similar to ordinal numbers: Depending on the context, they can be adverbs ("englisch"), or to-be-inflected adjectives ("englische/n/s/r"). This group is specially marked in the expansion table and consecutively in the markup (`ending="adjadv" sounds_like="englisch"`) for later processing (see 3.5.1 below).

Tokens which are identified as abbreviations but for which no entry in the expansion table is found are either spelled out, if they consist of no more than five characters, or left to be pronounced like normal words by the phonemisation component (see 3.5) if they are longer.

3.4 Part-of-speech tagger / chunk parser

Part-of-speech tagging is performed with the statistical tagger TnT (Brants, 2000), using the Stuttgart-Tübingen Tagset (STTS) (Schiller et al., 1995), and trained on the manually annotated NEGRA corpus (Skut et al., 1997). A chunk parser (Skut & Brants, 1998) is used to determine the boundaries of noun phrases, prepositional phrases and adjective phrases. In addition to punctuation and part-of-speech, this information about syntactic phrasing is useful for determining correct prosodic phrasing (see 3.6). Furthermore, syntactic phrases are used to delimit the domain for morphological unification, a prerequisite for assigning the correct inflection ending to expanded abbreviations and ordinal numbers (see 3.5.1).

Part-of-speech and chunking information is added to each token's `<t>` tag. For the chunking information, this is not actually a very satisfactory solution, as the local syntactic structure can hardly be considered a property of the individual token. However, the more logical representation of syntactic structure as an XML tree structure would possibly conflict with the prosodic structure, due to the fact that syntactic and prosodic structure cannot be guaranteed to coincide in all cases. As XML only allows for a proper tree structure, with no crossing edges, the only alternative seems to be to give up XML representation in the present form in favour of, e.g., a chart representation allowing more flexible edges. However, the presently used encoding with the XML structure representing prosodic structure and syntactic structure "squeezed" into the token tags seems to be a viable solution.

3.5 Phonemisation

The SAMPA phonetic alphabet for German (Wells, 1996) is used for the phonemic transcription. An extensive lexicon deals with known words, and a letter-to-sound conversion algorithm with unknown words; but first, a dedicated module adds inflection endings to ordinals and abbreviations.

3.5.1 Inflection endings

This module deals with the ordinals and abbreviations which have been marked during text normalisation (see 3.3) as requiring an appropriate inflection ending. The part-of-speech information added by the tagger tells whether the token is an adverb or an adjective. In addition, information about the boundaries of noun phrases has been provided by the chunker, which is relevant for adjectives.

In the lexicon, all entries occurring in noun phrases (determiners, adjectives, and nouns) are annotated with their possible value combinations for the morphological inflection information gender, number and case. In addition, determiners are marked as definite or indefinite. This information was obtained from the morphological analyser MMORPH (Petitpierre & Russell, 1995).

When the inflection endings module finds an ordinal or an abbreviation with an adjectival role, it performs a unification of the morphological variables over the known tokens in the noun phrase to which the ordinal or abbreviation belongs. In many cases, this allows the appropriate values of gender, number and case to be determined for the ordinal or abbreviation, so that the correct ending can be selected and added to the expanded form.

For example, in "mein 2. Angebot" (Engl. "my second offer"), the words "mein" and "Angebot" are looked up in the lexicon, their associated values for gender, number and case are compared, and only the common ones (gender=neutral, number=singular, case=nom./acc.) are retained. Further disambiguation is not necessary, as all remaining possibilities (neutral/singular/nom. and neutral/singular/acc.) correspond to the same adjective ending ("-s" with indefinite determiner "mein"), so the correct adjective ending can be added to the ordinal: "zweites".

3.5.2 Lexicon

The pronunciation lexicon is derived from CELEX (Baayen et al., 1995). It contains the graphemic form, a phonemic transcription, a special marking for adjectives, and the inflection information mentioned above (see 3.5.1).

As the inflection of adjectives is quite regular in German, only the stem form of an adjective is contained in the lexicon, while all inflected forms are generated by the lexicon lookup program.

The lexicon performs a simple compound treatment. If a word is not found in the lexicon but is the concatenation of two or more lexicon entries, the corresponding phonemic forms are concatenated. Optional bounding morphs (*Fugen* or infixes) such as "+s+", "+es+", "+n+", "+en+", and "+e+", typical for German noun compounds, are also allowed. For all parts of a compound except the first, primary word stress is reduced to secondary stress, i.e. the first part is considered the dominant one, which seems to be the default for German. Exceptions to this rule, such as "Bundesinnenminister", "Oberverwaltungsgericht", are part of the lexicon.[7]

3.5.3 Letter-to-sound conversion

Unknown words that cannot be phonemised with the help of the lexicon are analysed by a "letter-to-sound conversion" algorithm. This algorithm is more complex than a simple application of letter-to-sound rules: on the one hand, correct phonemisation relies in many cases on a correct identification of morpheme boundaries. On the other hand, for the phoneme string to be properly uttered, syllabification and word stress information needs to be added.

First, a morphological decomposition is attempted using a statistical morpheme "parser" based on the probability of two adjacent morphemes being neighbours. This had been trained on data extracted from CELEX (Baayen et al., 1995). The resulting morpheme chain is compared to a list of affixes which have a predictable effect on word stress position, either attracting or shifting the stress, or with no effect on stress

(Jessen, 1999).

The remaining morphemes are subjected to a set of generic letter-to-sound rules for German.

The syllabification of the transcribed morphemes is based on standard phonological principles such as the sonority hierarchy of phonemes, the maximal onset principle, the obligatory coda principle and the phonotactic restrictions for the German language (see also (Brinckmann & Trouvain, 2002)).

Finally, a word stress assignment algorithm decides which syllable receives the primary lexical stress. No rule-based secondary stress assignment is attempted at present.

3.6 Prosody rules

Prosody is modelled using GToBI (Grice et al., 2002), an adaptation of ToBI ("Tones and Break Indices") for German. ToBI (Silverman et al., 1992) describes intonation in terms of fundamental frequency (F0) target points, distinguishing between accents associated with prominent words and boundary tones associated with the end of a phrase. The size of a phrase break is encoded in break indices. Within MARY, break indices are used as follows: "2" is a potential boundary location (which might be "stepped up" and thus realised by some phonological process later on); "3" denotes an intermediate phrase break; "4" is used for intra-sentential phrase breaks; "5" and "6" (not part of GToBI) represent sentence-final and paragraph-final boundaries.

The prosody rules module assigns the symbolic GToBI labels. In a later step (see 3.8), these are translated into concrete F0 targets and pause durations and are taken into account for accentual lengthening and phrase-final lengthening in the duration module.

The prosody rules were derived through corpus analysis and are mostly based on part-of-speech and punctuation information. Prosodic boundaries are inserted at punctuation signs, conjunctions which are not part of co-ordinated noun phrases, and after the *Vorfeld* in statements, i.e. just before the first finite verb in a statement. The syntactic information which comes as an output of the chunk parser is used as an additional source for assigning prosodic boundaries, e.g. for special speaking styles. In (Trouvain, 2002) it has been shown that for slow speech, syntactic phrasing information is very useful to determine appropriate locations where to insert additional pauses.

Some parts-of-speech, such as nouns and adjectives, always receive an accent; the other parts-of-speech are ranked hierarchically (roughly: full verbs > modal verbs >

adverbs), according to their propensity for receiving an accent. This ranking comes into play where the obligatory assignment rules do not place any accent inside some intermediate phrase. According to a GToBI principle, each intermediate phrase should contain at least one pitch accent (Benzmüller & Grice, 1997). In such a case, the token in that intermediate phrase with the highest-ranking part-of-speech receives a pitch accent.

After determining the location of prosodic boundaries and pitch accents, the actual tones are assigned according to sentence type (declarative, interrogative-W, interrogative-Yes-No and exclamative). For each sentence type, pitch accent tones, intermediate phrase boundary tones and intonation phrase boundary tones are assigned. The last accent and intonation phrase tone in a sentence is usually different from the rest, in order to account for sentence-final intonation patterns.

3.7 Postlexical phonological processes

Once the words are transcribed in a standard phonemic string including syllable boundaries and lexical stress on the one hand, and the prosody labels for pitch accents and prosodic phrase boundaries are assigned on the other hand, the resulting phonological representation can be re-structured by a number of phonological rules. These rules operate on the basis of phonological context information such as pitch accent, word stress, the phrasal domain or, optionally, requested articulation precision. Currently, only segment-based rules apply, such as the elision of schwa in the endings "–en" and "–em", the backward assimilation of articulation place for nasal consonants, and the insertion of glottal stops before vowels of pitch-accented syllables with a free onset. For the future it is planned to take into account some re-structuring on the prosodic level, e.g. reducing the number of pitch accents and phrase boundaries for fast speech (Trouvain & Grice, 1999).

The output of this module gives the maximally rich MaryXML structure, containing all the information added to the structure by all of the preceding modules.

3.8 Calculation of acoustic parameters

This module performs the translation from the symbolic to the parametrical domain. The MaryXML structure is interpreted by duration rules and GToBI realisation rules.

The duration rules are at present a version of the Klatt rules (Klatt, 1979; Allen et al., 1987) adapted to German (Brinckmann & Trouvain, 2002). They have been shown to yield perceptual results only sightly inferior to a classification and regression tree (CART) trained on a corpus of German read speech (Brinckmann & Trouvain, 2002), while having the advantage of being readily interpretable e.g. by students of speech science.

The realisation of GToBI tones uses a set of target points for each tone symbol. These targets are positioned, on the time axis, relative to the nucleus of the syllable they are attached to; on the frequency axis, they are positioned relative to a descending pair of topline and baseline representing the highest and lowest possible frequency at a given moment. The fact that these lines are descending accounts for declination effects, i.e. overall F0 level is higher at the beginning of a phrase than close to the end. As an example, the GToBI accent "L+H*", associated with the syllable [fʊn] of the sequence [ɡə-ˈfʊn-dən] (Engl. "found") is realised as follows:

- first, the "L+" part is realised by a target on the baseline at the start of the nucleus of the preceding syllable (the schwa of [ɡə]);
- second, the "H*" part is realised by a target on the topline in the middle of the nucleus of the accented syllable (the [ʊ] in [ˈfʊn]).

As is illustrated in Figure 2, this allows the calculation of concrete frequency target values if the segment durations and the frequency values for the start and end points of the topline and baseline are known. Obviously, the latter values need to be set appropriately for the voice to be used during synthesis, in particular according to the sex of the speaker.

– insert Figure 2 about here –

The output produced by this module is no longer a MaryXML structure, but a list containing the individual segments with their durations as well as F0 targets. This format is compatible with the MBROLA `.pho` input files.

3.9 Synthesis

At present, the MBROLA diphone synthesiser (Dutoit et al., 1996) is used for synthesising the utterance based on the output of the preceding module. MBROLA was selected because of the comparatively low degree of distortions introduced into the signal during signal processing. The diphone sets of two German MBROLA voices (one male, one female) are presently used. Due to the modular architecture of the MARY system, any synthesis module with a similar interface could easily be employed instead or in addition.

4 The interface

An interface has been designed which allows the user to easily investigate parts of the MARY architecture tree (see Figure 1). Besides plain text and SABLE- or SSML-annotated text, each intermediate processing result can serve as input, and any subsequent processing result can be output.

In particular, it is possible to only investigate the translation of SABLE into MaryXML, i.e. the interpretation of high-level markup in terms of low-level markup. In the future, the XSLT stylesheet performing that translation is to be made editable from within the interface, allowing the user to experiment with realisation strategies for SABLE markup.

Individual processing steps can be carried out, allowing the user to understand the function of each module, or to investigate the source of an error. In addition, the intermediate results can be modified by hand, experimenting which input to a given module yields which output.

Figure 3 shows an example of such partial processing. The input text pane on the left side contains a partially processed version of the utterance “Ich fliege nach Schottland.” (lit. “I fly to Scotland.”), more precisely the output of the tagger/chunker module (corresponding to the *data type* “MaryXML tagged”). As a well-formed and valid XML document, it contains some header information (not shown in Figure 3, see e.g. Table 1), followed by the document body enclosed in `<maryxml>...</maryxml>` tags. In this example, the document consists of a single sentence (`<div>...</div>`) containing five tokens (four words and one punctuation mark). The tokens have already been enriched with some part-of-speech and syntactic information encoded as attribute/value pairs of the respective `<t>` tags. A “Verify” button allows the user to perform a validating XML parse of the input, making sure that the input is well-formed and valid (i.e., conforms to the MaryXML DTD (Harold, 1999)).

– insert Figure 3 about here –

Output of a given type can be obtained by simply selecting the desired output format (in this case, the output of the prosody module, “MaryXML Intonation”) and pressing the “Process” button. If both input and output are MaryXML, the “Compare” button allows the differences between the two versions of the document to be highlighted, which correspond to the information added by the selected processing steps.

If the output obtained in this step is to be used as input for subsequent processing steps, it can be transferred into the input text pane using the “Edit” button.

4.1 Teaching

The interface allows students to explore the workings of the individual modules in the TTS system. This can be done as a presentation performed by a teacher or interactively by the students themselves.

In order to disentangle the various components of a TTS system, it is helpful for the students to first walk through the individual modules from the very beginning to the very end. After each module, they can see the information that this module has added.

The example in Figure 3 shows an intermediate step in the processing of the sentence “Ich fliege nach Schottland”. As can be seen in the MARY system architecture (Figure 1), only the prosody module is needed to perform the transformation shown in Figure 3. In this case, the added information (highlighted) represents the beginning and end of intonation phrases, the location of prosodic boundaries with their strengths, as well as the location and type of pitch accents and boundary tones.

More advanced students can explore the functioning of a particular module in more detail by modifying specific pieces of information in that module’s input and observe the changes in the output. In the example shown in Figure 3, the effect of changing a token’s part-of-speech on accenting can be observed by changing, e.g., the part-of-speech of the token “fliege” from VVFIN (finite full verb) to NN (noun).

4.2 Development

A possible development task could lie in the domain of speech synthesis markup realisation, i.e. the interpretation of high-level markup (e.g. SABLE) in terms of lower level internal MaryXML markup. As an example, one might be interested in interactively determining an appropriate rendering of “strong” emphasis, which can be expressed in SABLE using the tag `<EMPH LEVEL="strong">`.

Since there does not seem to be a generally accepted definition of “emphasis” one can think of several possibilities how such a rather abstract concept can be interpreted and realised. In one sense “emphasis” can be equated with “perceived prominence”; in a second sense it can intensify an already-existing focus by prosodic means; in a third sense it can stand for a focus shift triggering a prosodic restructuring. Probably there are more interpretations of the meaning of “emphasis”. As it is unclear what sort of interpretation a SABLE tag user intended, a definition as wide as possible seems most appropriate for the task to “put life into” the emphasis tag.

Take again the sentence “Ich fliege nach Schottland” with an `<EMPH>` tag around “fliege” in order to emphasise the fact that one is flying and not driving to Scotland. Linguistic intuition suggests that the following changes could be applied to the default in order to make “fliege” sound emphasised:

- assign an appropriate pitch accent on the word to be emphasised (“fliege”)
- delete the default pitch accent (“Schottland”)
- insert pauses (e.g., 200 ms) around the word to be emphasised
- lengthen the segment duration of the lexically stressed vowel in the word to be emphasised (lengthen [iː] in our example e.g. by factor 2)
- increase F0 excursion size of the F0 peak in the word to be emphasised (e.g. by 15%)
- reset the intonation to a very low level after the word to be emphasised
- a slight F0 falling contour after the nuclear syllable in the word to be emphasised
- increase articulation precision[8]

All of these changes can be requested by using appropriate MaryXML tags.

4.3 Research

Speech synthesis allows the controlled creation of stimuli for perception experiments, be it for applied research (system improvement) or basic research (knowledge increase). The MaryXML markup makes the linguistic units used at any stage of processing accessible. Researchers wanting to modify these units can do this in a controlled way. The fact that intermediate processing results are accessible at all stages of processing gives the experimenters a free choice of the level of abstraction or amount of detailed control which they require for their stimuli. It is also possible to provide input generally on a high level of abstraction (e.g., specifying intonation using GToBI labels), to verify the acoustic parameters generated from these labels, and to fine-tune individual parameters where necessary. This combines the benefits arising from the conciseness of symbolic labelling with the control provided by editing acoustic parameters.

For example, (Brinckmann & Trouvain, 2002) evaluated the role of segmental duration prediction and of phonological symbolic representation in the perceptual quality of synthetic speech, in order to determine priorities for the improvement of speech timing. In view of perception experiments, stimuli were generated using the MARY interface. To that end, intermediate processing results were modified on two different levels: on the one hand, the segmental and prosodic symbolic representation; on the other hand, the acoustic parameters for duration and F0 for each segment. Segment durations were predicted using two standard duration models (Klatt rules and CART) or were taken from a natural speech database. The input to the duration prediction models consisted of a symbolic representation which was either derived from the database or calculated by the MARY system. Results of the perception experiments showed that different duration models can only be distinguished when the symbolic representation is appropriate.

An example for basic research is given by (Baumann & Trouvain, 2001). For a perception test with read telephone numbers, they created stimuli varying in pitch accent and pause structure. The findings of this study supported the idea that strategies for reading telephone numbers found in human speech production are preferred over strategies currently employed in telephone inquiry systems. The advantage of the MARY web interface is that it delivers a comfortable way of preparing the stimuli for such perception tests from everywhere with no need to install the system locally.

5 Synthesis of emotional speech

The modular architecture of the MARY system, based on an internal XML representation, is useful beyond the interface presented above, providing access for expert users. An example for the benefits of the system structure is the ease with which prosodic parameters can be modified using a graphical frontend, e.g. for the synthesis of emotional speech.

The approach to emotional speech synthesis sketched in the following is based on emotion dimensions and was first explored by (Murray & Arnott, 1995) in their HAMLET system. Figure 4 presents a two-dimensional description of emotional states, the so-called activation-evaluation space (Schlosberg, 1941; Russell, 1980; Cowie et al., 2001). The design is similar to Feeltrace (Cowie et al., 2000), a rating tool for emotional expression. The essential properties of an emotional speaker state, namely the activation or arousal of the speaker and the evaluation or valence towards any object (in terms of negative or positive), are represented by a location in a two-dimensional space. Feeltrace presents a graphical representation of this space to users who rate the perceived content of an emotional "clip". The interface presented in Figure 4 adapts the idea to the generation of emotional speech synthesis: A user can determine the emotional colouring of the utterance to be spoken by placing the cursor in the activation-evaluation space.

In the screenshot, the cursor is at a highly active and moderately positive position corresponding to an emotional state such as, e.g., positive surprise. This corresponds roughly to the emotion expressed in the text, "Hurra, wir haben es geschafft!" (Engl. "Yippie, we did it!").

– insert Figure 4 about here –

– insert Table 1 about here –

Through an (as yet minimal) set of rules (Schröder et al., 2001), the co-ordinates of the cursor in activation-evaluation space are translated into global prosodic settings such as the F0 level and range (frequencies of topline and baseline) and overall speech rate. An appropriate MaryXML document containing these prosodic settings is automatically generated (Table 1). The document is sent to the MARY server, causing the text to be synthesised with the specified prosodic settings. The resulting audio file can then be played via a button in the interface (Figure 4). In this way, it is possible to interactively explore the prosodic effects of the shades of emotion according to the underlying prosody rules.

It could be argued that an XML representation internal to the TTS system would not be necessary for obtaining the functionality just presented, as the prosodic variables specified in this early demonstrator are also specifyable in SABLE. However, the types of vocal cues which are relevant for emotion expression in speech synthesis have been shown to be much more numerous than the ones modelled at this early stage, and to include, e.g., voice quality, steepness of F0 rises and falls, and intonation contour type (see (Schröder, 2001) for a review). More sophisticated prosody rules are expected to incorporate these parameters, and therefore are likely to need access to finer TTS control than what is accessible through SABLE.

6 Summary

An overview of the processing components of the German text-to-speech system MARY has been given. It has been described how MaryXML, a system-internal XML-based data representation, can be used to make partial processing results available outside the system. The advantages of MaryXML are three-fold:

1. *All* intermediate processing results can be made visible;
2. these intermediate results can be modified and fed back as input into the system;
3. via the WWW, the interface is accessible from everywhere without a local installation of the system.

These features are very helpful for teaching purposes and for non-technical users. In addition, the benefits of these features for research and development of TTS synthesis were demonstrated using a number of concrete examples.

Acknowledgements

Many persons have contributed to the development of the MARY system over the years. We wish to thank, in alphabetical order, Ralf Benzmüller, Caren Brinckmann, Martine Grice, Tilman Jaeger, Kerstin Klöckner, Brigitte Krenn, Annette Preissner, Diana

Raileanu, and Henrik Theiling. We would also like to thank Markus Becker and Bettina Braun as well as three anonymous reviewers for their helpful comments on an earlier draft of this paper.

Part of this research is supported by the EC Project NECA, IST-2000-28580.

References

Allen, J., Hunnicutt, S., and Klatt, D. H. (1987). *From Text to Speech: The MITalk System*. Cambridge University Press, Cambridge, UK.

Baayen, R. H., Piepenbrock, R., and Gulikers, L. (1995). *The CELEX Lexical Database (CDROM)*. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA, USA.

Baumann, S. and Trouvain, J. (2001). On the prosody of German telephone numbers. In *Proceedings of Eurospeech 2001*, pages 557–560, Aalborg, Denmark.

Benzmüller, R. and Grice, M. (1997). Trainingsmaterialien zur Etikettierung deutscher Intonation mit GToBI. *Phonus 3, Research Report of the Institute of Phonetics, University of the Saarland*, pages 9–34.

Black, A., Taylor, P., and Caley, R. (1999). Festival speech synthesis system, edition 1.4. Technical report, Centre for Speech Technology Research, University of Edinburgh, UK.
http://www.cstr.ed.ac.uk/projects/festival

Brants, T. (2000). TnT – a statistical part-of-speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing*, Seattle, WA, USA.
http://www.coli.uni-sb.de/˜thorsten/publications

Breitenbücher, M. (1999). Textvorverarbeitung zur deutschen Version des Festival Text-to-Speech Synthese Systems. Technical report, IMS Stuttgart.
http://elib.uni-stuttgart.de/opus/volltexte/1999/225

Brinckmann, C. and Trouvain, J. (2002). The role of duration models and symbolic representation for timing in synthetic speech. *International Journal of Speech Technology*, page this volume.

Cowie, R., Douglas-Cowie, E., Savvidou, S., McMahon, E., Sawey, M., and Schröder, M. (2000). ’FEELTRACE’: An instrument for recording perceived emotion in real time. In *Proceedings of the ISCA Workshop on Speech and Emotion*, pages 19–24, Northern Ireland. http://www.qub.ac.uk/en/isca/proceedings

Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., and Taylor, J. (2001). Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, 18(1):32–80.

Dutoit, T. (1997). *An Introduction to Text-to-Speech Synthesis*. Kluwer Academic Publishers, Dordrecht.

Dutoit, T., Pagel, V., Pierret, N., Bataille, F., and van der Vrecken, O. (1996). The MBROLA project: Towards a set of high quality speech synthesisers free of use for non commercial purposes. In *Proceedings of the 4th International Conference of Spoken Language Processing*, pages 1393–1396, Philadelphia, USA.

Grice, M., Baumann, S., and Benzmüller, R. (2002). German intonation in autosegmental-metrical phonology. In Jun, S.-A., editor, *Prosodic Typology*. Oxford University Press.

Harold, E. R. (1999). *XML Bible*. Hungry Minds, Inc. http://www.ibiblio.org/xml/books/bible

Hoffmann, R., Kordon, U., Kürbis, S., Ketzmerick, B., and Fellbaum, K. (1999). An interactive course on speech synthesis. In *Proceedings of the ESCA/SOCRATES Workshop MATISSE*, pages 61–64.

Jessen, M. (1999). German. In van der Hulst, H., editor, *Word Prosodic Systems in the Languages of Europe*, pages 515–545. Mouton de Gruyter, Berlin, New York.

JSML (1999). Java speech markup language 0.6. Technical report, Sun Microsystems. http://java.sun.com/products/java-media/speech/forDevelopers/JSML

Klabbers, E., Stöber, K., Veldhuis, R., Wagner, P., and Breuer, S. (2001). Speech synthesis development made easy: The Bonn Open Synthesis System. In *Proceedings of Eurospeech 2001*, pages 521–524, Aalborg, Denmark.

Klatt, D. H. (1979). Synthesis by rule of segmental durations in English sentences. In Lindblom, B. and Öhman, S., editors, *Frontiers of Speech Communication*, pages 287–299. Academic, New York.

Microsoft (2002). *SAPI 5: Microsoft Speech API 5.1*.
http://www.microsoft.com/speech

Möbius, B. (1999). The Bell Labs German text-to-speechsystem. *Computer Speech and Language*, 13:319–357.

Murray, I. R. and Arnott, J. L. (1995). Implementation and testing of a system for producing emotion-by-rule in synthetic speech. *Speech Communication*, 16:369–390.

Petitpierre, D. and Russell, G. (1995). MMORPH – the Multext morphology program. deliverable report, MULTEXT.
ftp://issco-ftp.unige.ch/pub/multext/mmorph.doc.ps.tar.gz

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39:1161–1178.

Schiller, A., Teufel, S., and Thielen, C. (1995). Guidelines für das Tagging deutscher Textkorpora mit STTS. Technical report, IMS-CL, University Stuttgart.
http://www.sfs.nphil.uni-tuebingen.de/Elwis/stts/stts.html

Schlosberg, H. (1941). A scale for the judgement of facial expressions. *Journal of Experimental Psychology*, 29:497–510.

Schröder, M. (2001). Emotional speech synthesis: A review. In *Proceedings of Eurospeech 2001*, volume 1, pages 561–564, Aalborg, Denmark.
http://www.dfki.de/˜schroed

Schröder, M., Cowie, R., Douglas-Cowie, E., Westerdijk, M., and Gielen, S. (2001). Acoustic correlates of emotion dimensions in view of speech synthesis. In *Proceedings of Eurospeech 2001*, volume 1, pages 87–90, Aalborg, Denmark.
http://www.dfki.de/˜schroed

Silverman, K., Beckman, M., Pitrelli, J., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J., and Hirschberg, J. (1992). ToBI: A standard for labeling english prosody. In *Proceedings of the 2nd International Conference of Spoken Language Processing*, pages 867–870, Banff, Canada.

Skut, W. and Brants, T. (1998). Chunk tagger – statistical recognition of noun phrases. In *Proceedings of the ESSLLI Workshop on Automated Acquisition of Syntax and Parsing*, Saarbrücken, Germany. http://www.coli.uni-sb.de/˜thorsten/publications

Skut, W., Krenn, B., Brants, T., and Uszkoreit, H. (1997). An annotation scheme for free word order languages. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, Washington D.C., USA.
http://www.coli.uni-sb.de/sfb378/negra-corpus/negra-corpus.html

Sproat, R., editor (1997). *Multilingual Text-to-Speech Synthesis: The Bell Labs Approach*. Kluwer, Boston.

Sproat, R., Hunt, A., Ostendorf, M., Taylor, P., Black, A., Lenzo, K., and Edgington, M. (1998). SABLE: A standard for TTS markup. In *Proceedings of the 5th International Conference of Spoken Language Processing*, pages 1719–1724, Sydney, Australia.

Sproat, R., Taylor, P. A., Tanenblatt, M., and Isard, A. (1997). A markup language for text-to-speech synthesis. In *Proceedings of Eurospeech 1997*, Rhodes/Athens, Greece.

Taylor, P. and Isard, A. (1997). SSML: A speech synthesis markup language. *Speech Communication*, 21:123–133.

Traber, C. (1993). Syntactic processing and prosody control in the SVOX TTS system for German. In *Proceedings of Eurospeech 1993*, pages 2099–2102, Berlin, Germany.

Trouvain, J. (2002). Tempo control in speech synthesis by prosodic phrasing. In *Proceedings of Konvens*, Saarbrücken, Germany.

Trouvain, J. and Grice, M. (1999). The effect of tempo on prosodic structure. In *Proceedings of the 14th International Conference of Phonetic Sciences*, pages 1067–1070, San Francisco, USA.

VoiceXML (2000). *VoiceXML 1.0 Specification*. VoiceXML Forum.
http://www.voicexml.org

Walker, M. R. and Hunt, A. (2001). *Speech Synthesis Markup Language Specification*. W3C. http://www.w3.org/TR/speech-synthesis

Wells, J. C. (1996). *SAMPA Phonetic Alphabet for German*. http://www.phon.ucl.ac.uk/home/sampa/german.html

Notes

[1] The system is accessible online under `http://mary.dfki.de`. Notice that the web interface is visually different from the interface described in this paper, but provides nearly identical functionality.

[2] http://www.ims.uni-stuttgart.de/phonetik/synthesis

[3] The DTD can be found at `http://mary.dfki.de/lib/MaryXML.dtd`. An XML Schema-based definition of MaryXML is planned; the Schema will reside at `http://mary.dfki.de/lib/MaryXML.xsd`.

[4] These prosodic settings are meant to realise the abstract concept of emphasis, which does not seem to be a clearly defined concept and which seems to encompass concepts as different as contrasting accentuation and paralinguistic intensification (see 4.2). Not least because of these conceptual difficulties, the parameters selected for the realisation of emphasis are currently based on linguistic intuition rather than hard scientific evidence.

[5] An excellent overview of the phenomena that need to be accounted for in German text normalisation has been given by (Breitenbücher, 1999).

[6] A different solution for this problem, employing a sentence grammar, is used in the SVOX system (Traber, 1993).

[7] A more elaborate approach to productive compounding in German, including morphological decomposition, using weighted affix and stem lexicons, can be found e.g. in (Möbius, 1999).

[8] The change of other parameters such as intensity and voice quality is not yet possible in MARY.

List of Figures

List of Tables

Table 1: A MaryXML document automatically generated by the interface presented in Figure 4. The co-ordinates in activation-evaluation space influence the prosodic settings F0 baseline, F0 range, and speech rate.

```
 <?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE maryxml SYSTEM "http://mary.dfki.de/lib/MaryXML.dtd">
<maryxml>
<div>
<phrase baseline="29%" range="78%">
<rate speed="54%">
Hurra, wir haben es geschafft!
</rate>
</phrase>
</div>
</maryxml>
```

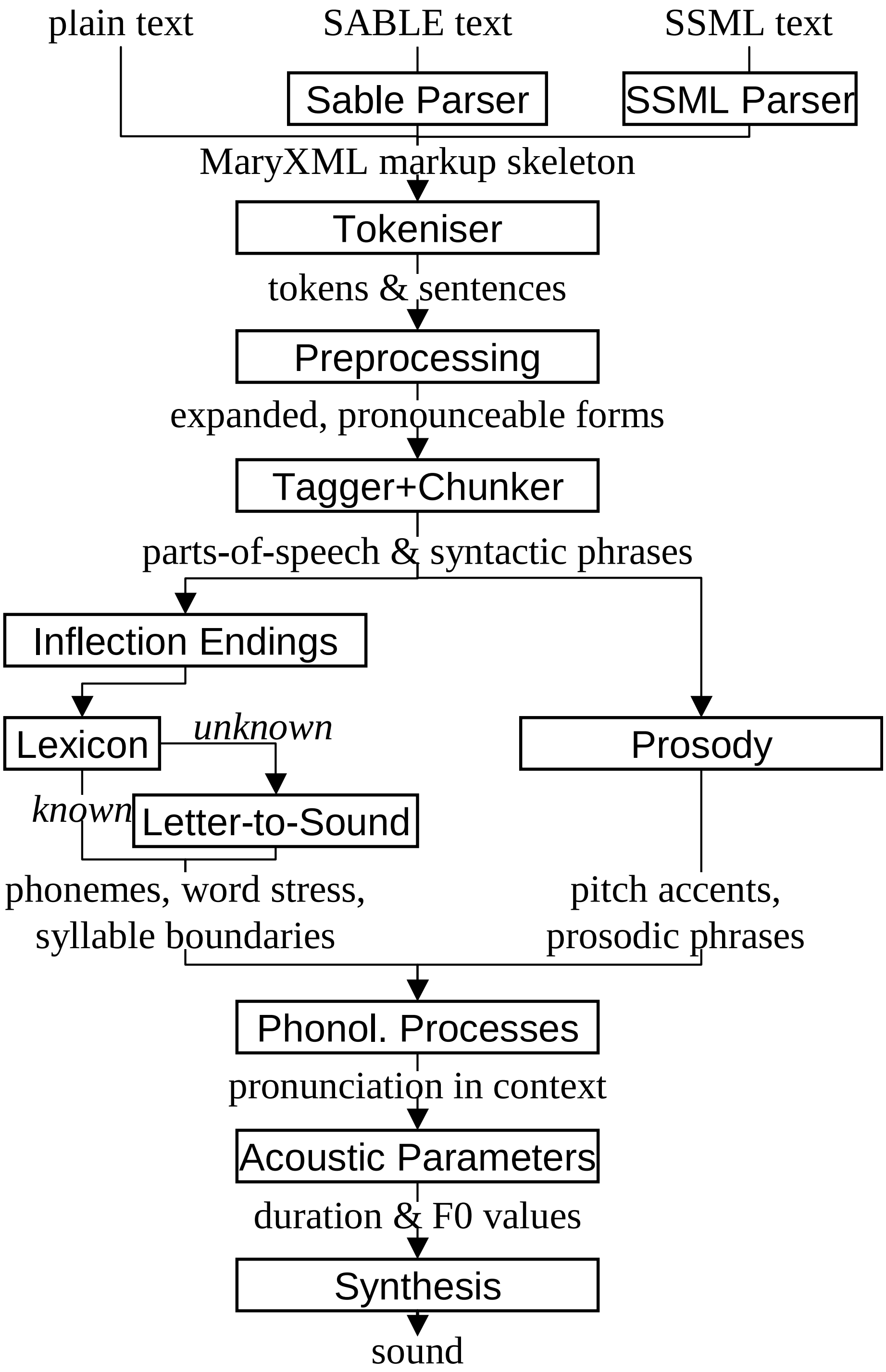


Figure 1: The architecture of the MARY TTS system.

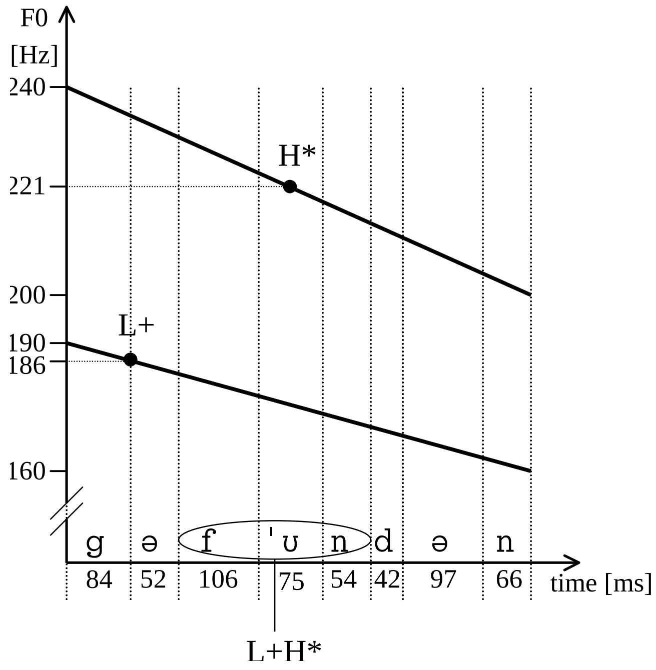
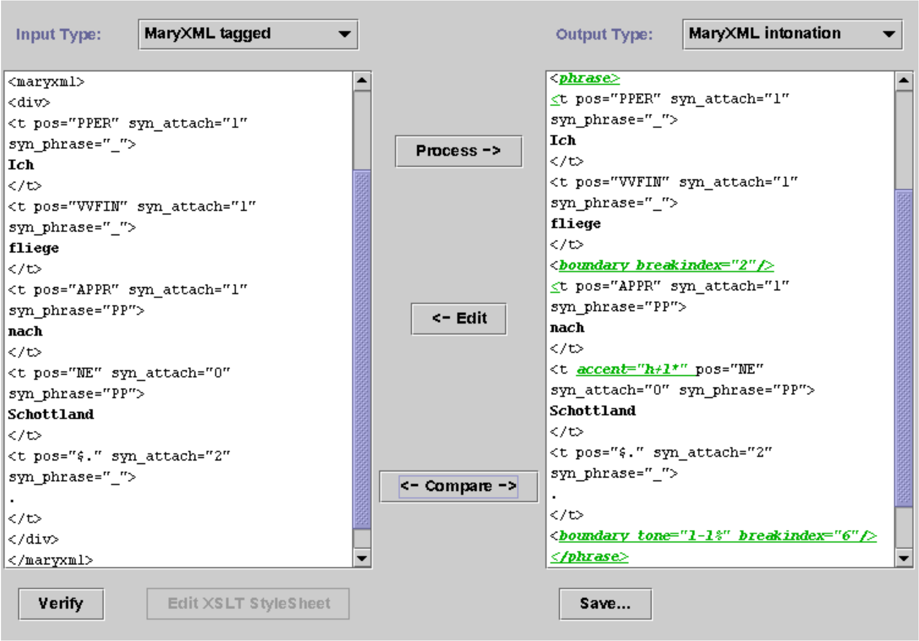


Figure 2: Illustration of the calculation of frequency parameters for target points realising the GToBI accent L+H*.



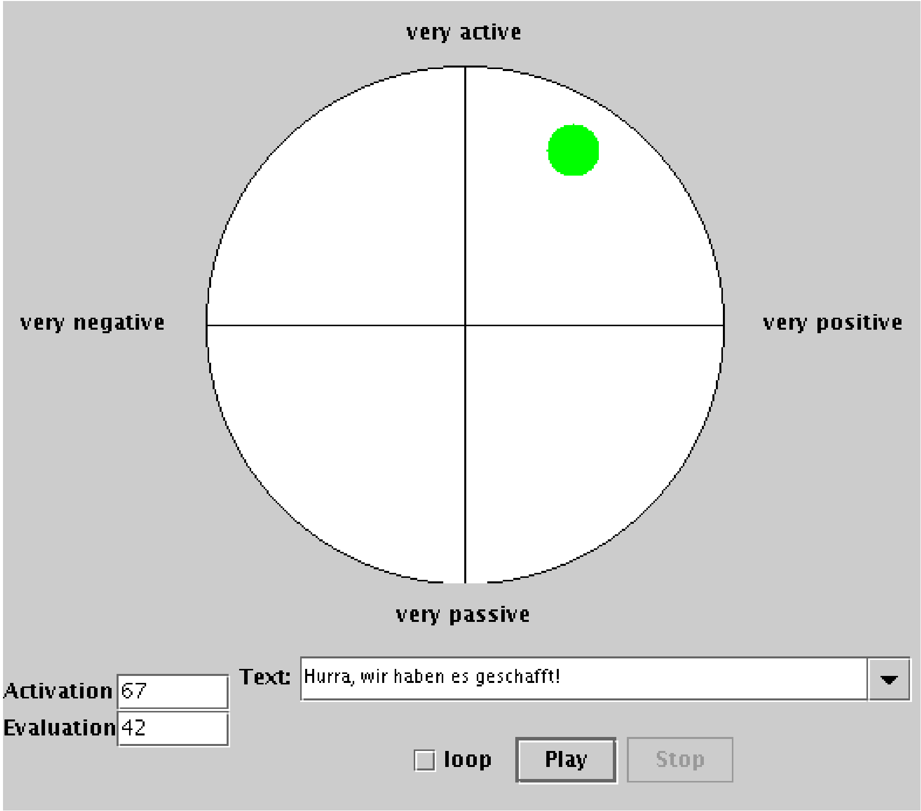
Input Type: MaryXML tagged

Output Type: MaryXML intonation

```
<maryxml>
<div>
<t pos="PPER" syn_attach="1"
syn_phrase="_">
Ich
</t>
<t pos="VVFIN" syn_attach="1"
syn_phrase="_">
fliege
</t>
<t pos="APPR" syn_attach="1"
syn_phrase="PP">
nach
</t>
<t pos="NE" syn_attach="0"
syn_phrase="PP">
Schottland
</t>
<t pos="$." syn_attach="2"
syn_phrase="_">
.
</t>
</div>
</maryxml>
```

```
<phrase>
<t pos="PPER" syn_attach="1"
syn_phrase="_">
Ich
</t>
<t pos="VVFIN" syn_attach="1"
syn_phrase="_">
fliege
</t>
<boundary breakindex="2"/>
<t pos="APPR" syn_attach="1"
syn_phrase="PP">
nach
</t>
<t accent="h+l*" pos="NE"
syn_attach="0" syn_phrase="PP">
Schottland
</t>
<t pos="$." syn_attach="2"
syn_phrase="_">
.
</t>
<boundary tone="l-l%" breakindex="6"/>
</phrase>
```

Figure 3: Example of partial processing with the MARY interface. See text (4) for explanations.



Activation 67

Evaluation 42

Text: Hurra, wir haben es geschafft!

Figure 4: An interface for emotional speech synthesis, representing activation-evaluation space. Values for activation and evaluation reach from -100 to 100.