Project ref. no.	*EDC-22124 ESIGN / 27960*
Project acronym	***eSIGN***
Project full title	***Essential sign language information on government networks***

Security (distribution level)	*CO*
Contractual date of delivery	*T12 = August 2003*
Actual date of delivery	*30. Sept. 2003*
Deliverable number	*D2.3*
Deliverable name	*Intelligent Sign Editor*
Type	*Demonstrator*
Status & version	*final (1.0d23)*
Number of pages	*27*
WP contributing to the deliverable	*WP 2*
WP / Task responsible	*UH*
Other contributors	
Author(s)	*Thomas Hanke, Hortensia Popescu*
EC Project Officer	*Evangelia Markidou*
Keywords	*Avatar, Virtual Human, Animation, Sign Language, HamNoSys, eSIGN editor*
Abstract (for dissemination)	*This deliverable describes the eSIGN editor software which allows the user to compose signed text to be performed by the eSIGN Avatar.*

Table of Contents

1   Executive Summary

The eSIGN Editor software allows the user to compose signed text to be performed by the eSIGN Avatar. The editor gives the user an economic approach to create signed sequences by selecting signs from the lexicon and then modifying them with the assistance of specialised editors (that focus on different aspects of the sign's phonetics and morphology) where necessary.

The tool is delivered for Windows 32bit and MacOS platforms in three user interface languages (German, English, and Dutch). Modules to support features specific to individual sign languages (such as fingerspelling and counting) can be plugged in.

2   Introduction

A key component to ensure equal access of Deaf people to information society is the provision of web pages in their native language, sign language. As there is no established written form for any sign language, and digital video is too expensive to produce and consume, sign language animation is a promising approach.

Machine translation into sign language has only recently started as a research discipline and will need years to mature to the extent that it could be usable in this context except in very restricted services where a language model can be created. For free text, the approach taken in eSIGN is to support a human translator in order to make the translation process from spoken language into animation input as economic as possible. The central tool for that task is the eSIGN Editor.

The main idea of that program is to pick signed sentences sign-by-sign from the lexicon, and to apply morphological changes to individual signs or strings of signs as necessary. Different "assistants" are available to deal with different types of morphological and other modifications. This includes pronunciation databases to ease the description of mouth pictures and mouth gesture databases. While "writing" the signed text, the user has the possibility to immediately check the contents by sending the text to the Avatar.
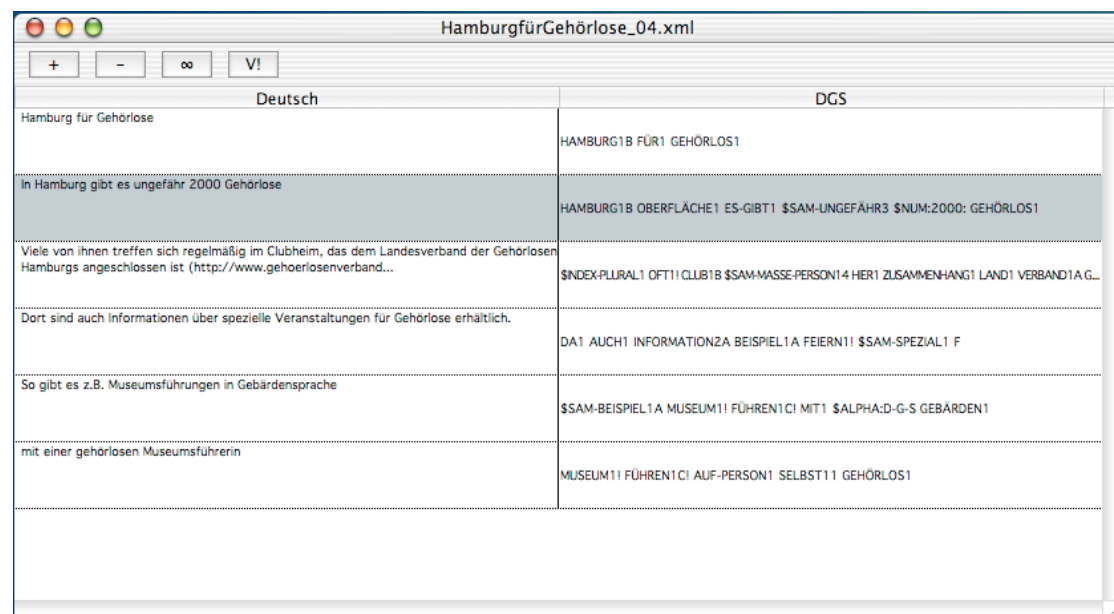
Depending on the working environment at the content creator's site, the creator can use lexicon maintenance functionality within the eSIGN Editor or use external programs for that purpose.

Documents created with the editor can be exported to SiGML. The integration of SiGML into web pages still is a manual process at this point of time.

As the tool is now used for all three content sites instead of one as originally planned, the focus for the tool had to be shifted towards delivering a cross-platform tool that allows plug-ins to allow specific editors for each of the target languages, e.g. fingerspelling. While the tool does provide a rather flexible assistance in combining parametrised signs in linguistically meaningful ways, we still have lots of ideas to make working with the tool more efficient. These ideas will find their way into future revisions of the tool.
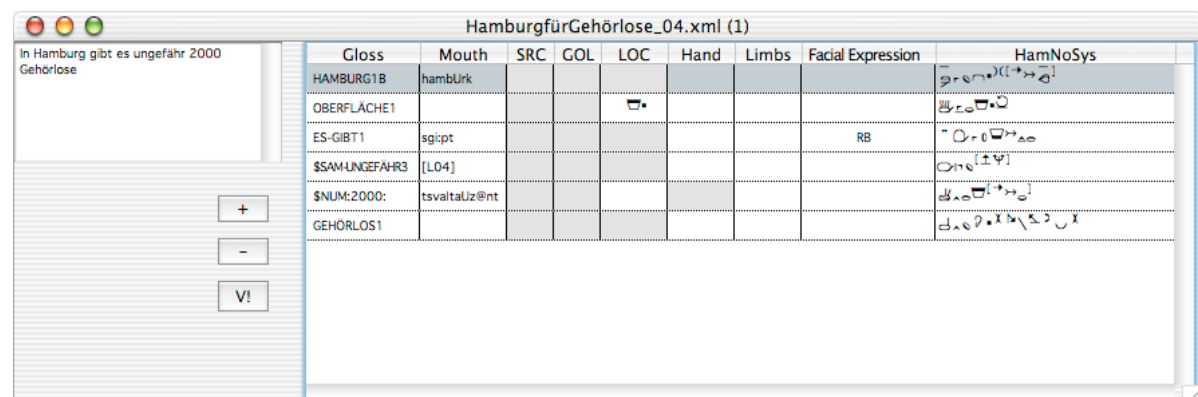
3   Quick tour through the eSIGN Editor

An eSIGN Editor document represents a signed text. In the document window, the signed text is represented by sequences of glosses parallel to the (optional) spoken language text that serves as the translation basis.

For creating the signed contents, each utterance is handled separately. In order to create or modify an utterance, the user opens a window to specify the sequence of signs.


Signs are represented by glosses[1] as well as a couple of form aspects. Some columns are governed by the lexical entries, i.e. the lexical entry determines whether a field may be filled by the user (otherwise it is grey) and which default value to use. The

---

[1] Glosses are spoken-language labels for signs that match the semantics of a sign as closely as possible. Glossing is widely used not only in academic contexts, but also for note-taking by deaf people. However, the danger of using another language for describing utterances in one language has to be kept in mind (cf. Pizzuto/Pietrandrea 2001).

user can view the utterance as a whole or sign-by-sign by using the Avatar playback controls.

New signs are usually chosen from the lexicon. A dialog window allows the user to specify search criteria (parts of the gloss, HamNoSys (Schmaling/Hanke 2001) form aspects, or part of speech) and then select from the result set.
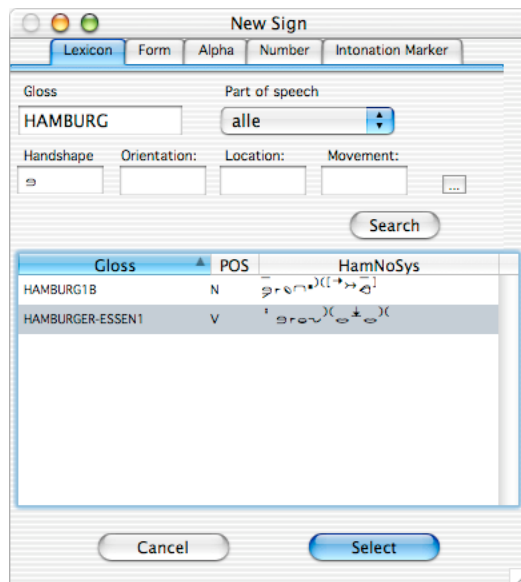

If a sign is not in the lexicon, it can be specified by directly entering a HamNoSys string:
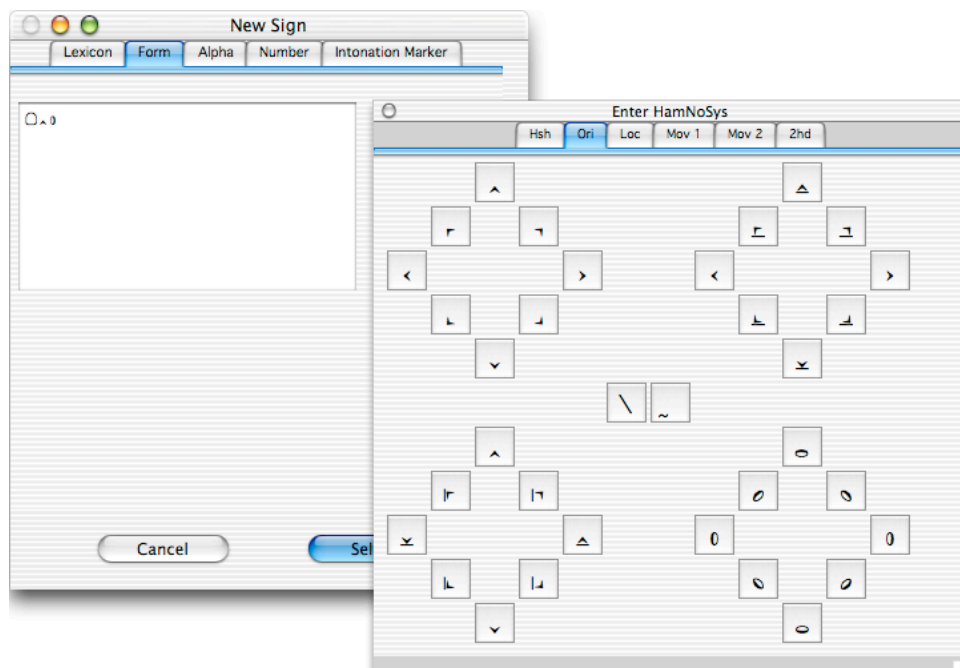
For numbers and fingerspelling, special editors are available.

Lexical entries usually have a mouthing, either a mouth picture or a mouth gesture. As there is no strict one-to-one mapping, the user has the option to specify an alternate form. For mouth pictures, a spoken language text can be entered and then be converted to the SAMPA (Gibbon et al. 1997) pronunciation encoding used in ViSiCAST for viseme description.[2]
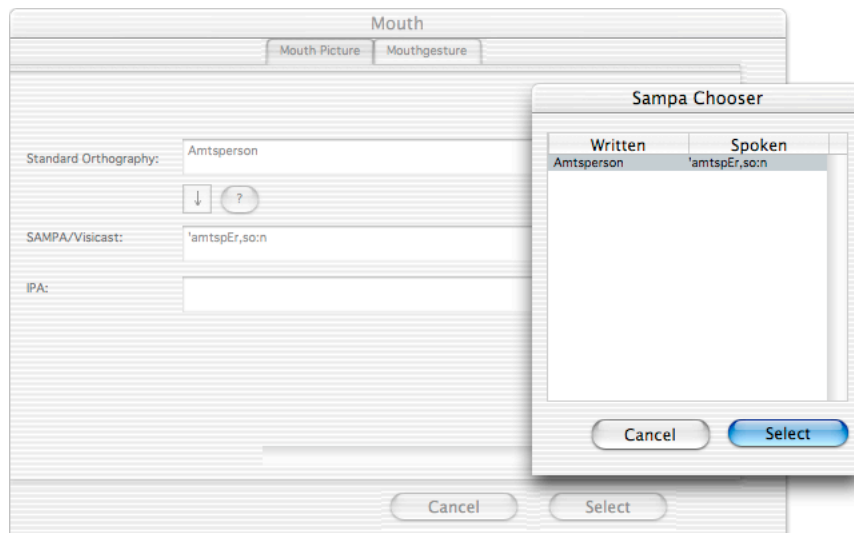
Mouth gestures are selected from a list. For each mouth gesture, the code is shown along with a picture, a movie, and a description.

---
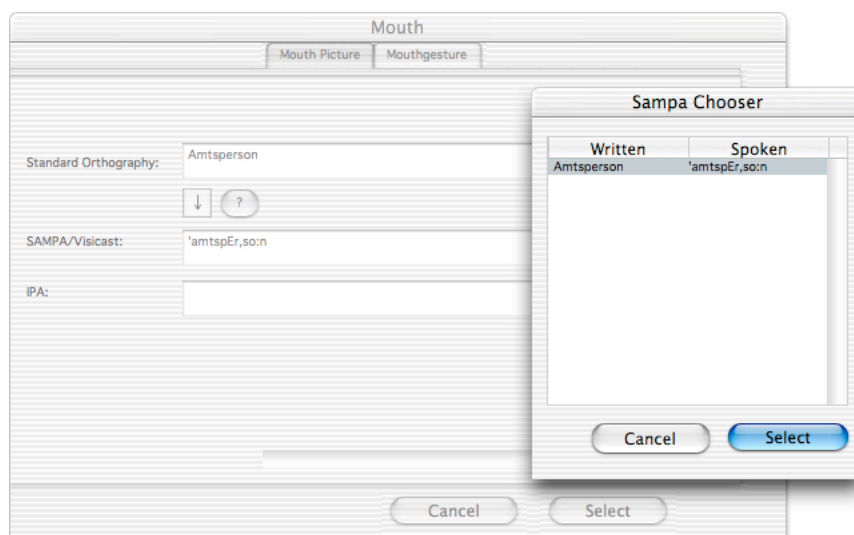
[2] The pronunciation database builds on the Bonn machine-readable pronunciation dictionary (Portele et al. 1995).
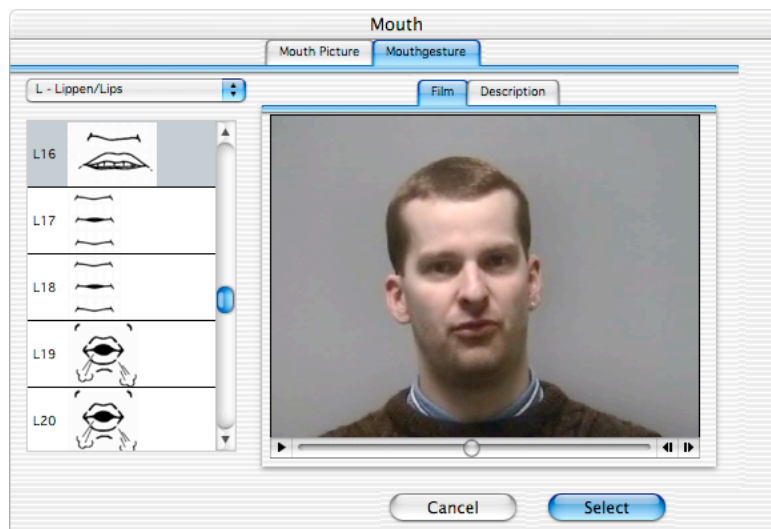
The lexical entries for directional signs as well as locatable signs are not specified for their location in signing space. Default positions are assumed until the user defines them:
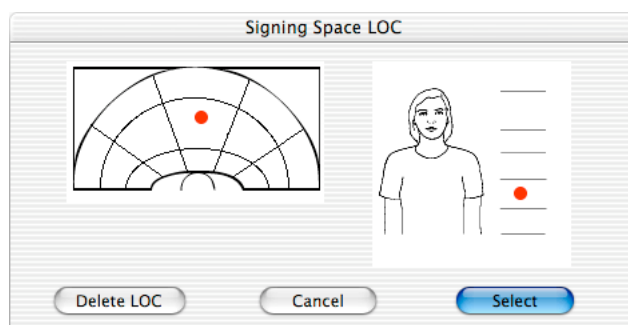
In the same way, the user can add handshape/classifier information for those signs that depend on their arguments to set handshape, orientation, and handedness:
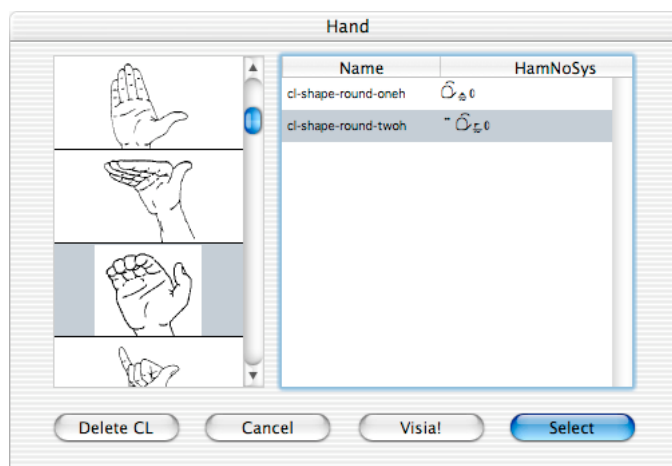
For convenience reasons, discourse structuring signals can be added to utterances in a single action even if they consist of sequences of signs:
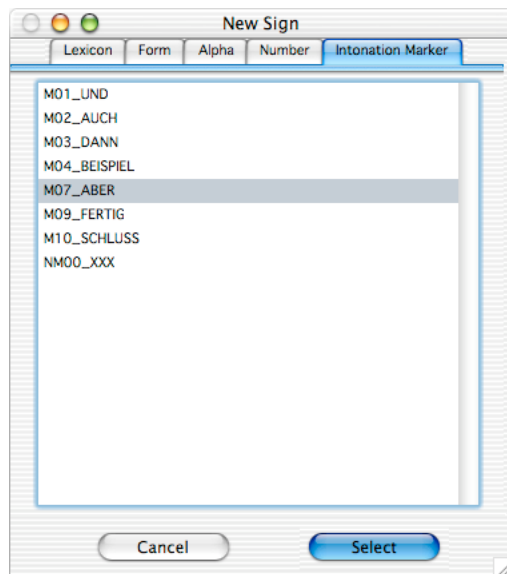
The inventory of these signals can be built inside the eSIGN Editor itself.

4   eSIGN Editor user documentation

4.1   eSIGN Editor documents

Documents are used to store one signed text each. It is up to the user to decide what a text is: There is no limit to the amount of utterances or signs in a document. But as it is possible and convenient to work with multiple documents at the same time, a text can also be split into a number of documents.

A new document is created with the "New Document…" command in the "File" menu. A window will open containing the contents of the new document. As soon as a document is open, "Save" and "Save as…" commands are available.

Initially, the new document consists of zero utterances.
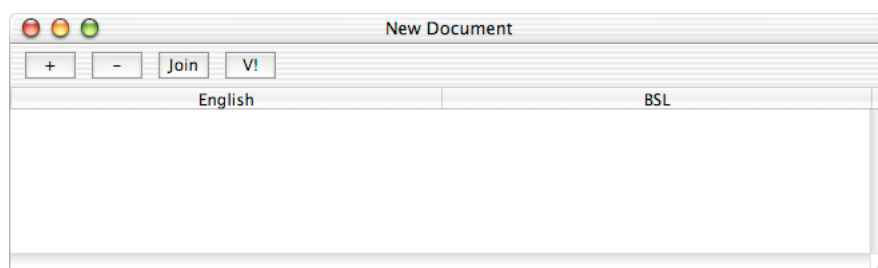
Utterances can be produced either manually by clicking on the "+" button or by dragging a text file containing a spoken language text onto the window. In the latter case the editor will create an utterance for each paragraph in the source text.
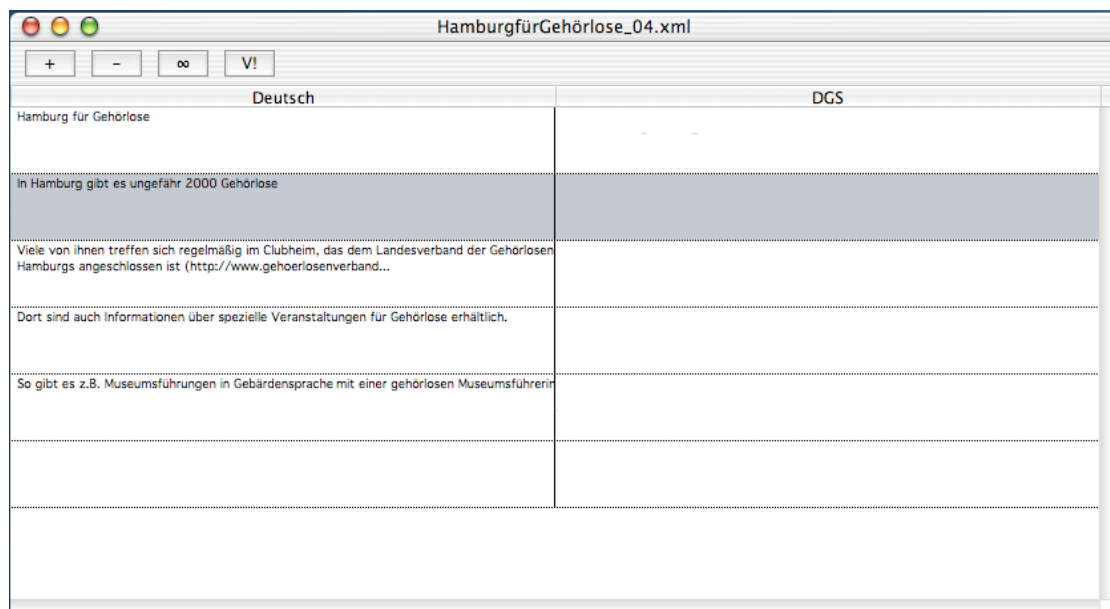
To prepare for the actual translation, the utterances can be reordered (by dragging them around after clicking on the right column of a row), joined (button "∞"/Win: "Join"), or split : To split an utterance into two, place the insertion mark in the left

column text and press the Enter key: The text after the insertion mark will be moved into a new utterance.

Double clicking a row (in the right column) will open an extra window for the corresponding utterance.

With the "-" button, selected utterances can be deleted. The "V!" buttons sends the selected utterances to the avatar, cf. below.
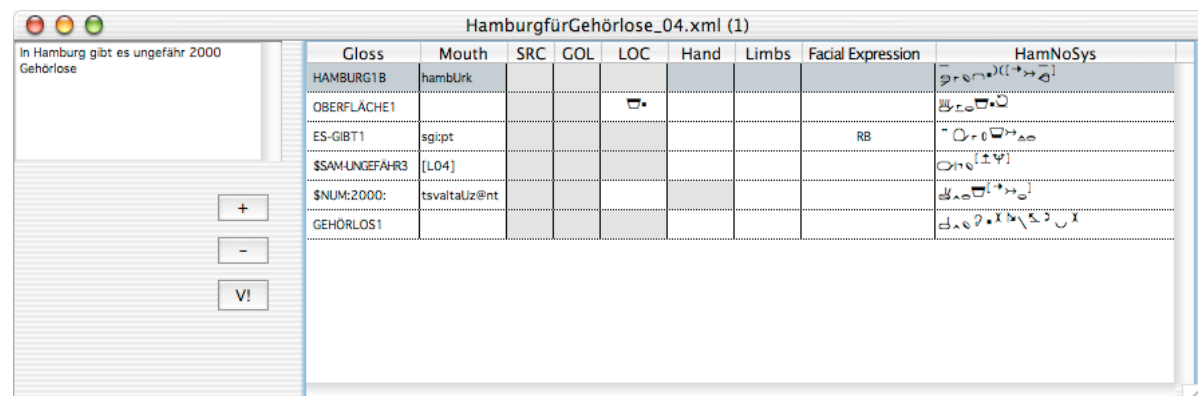
4.2 Utterance windows

The text field on the upper left side of the window shows the spoken language text corresponding to the utterance. The list on the right shows the signs the utterance consists of. New signs can be added by clicking on "+" (or by dragging in signs from other places), the selected signs can be deleted with the "-" button. By dragging the rows, their order can be changed. The selected signs can be sent to the avatar by clicking on "V!".

To add a sign to the utterance,

click on "+".

click on the appropriate tab for the kind of sign you want to add:

- o Lexical signs can be retrieved from the "Lexicon".

  Define the search criteria with the "Gloss", "Handshape" etc. text fields and the part of speech menu and click "Search". In the lower part of the window you will see the search results. Select one sign from the list.

  For entering HamNoSys into the "Handshape" text field and its siblings, click on "…" to view the HamNoSys input palette.

Please note that the search routine simply looks for matches of all four HamNoSys fields, and does not care whether the contents really is what the text field label says. Thereby you can look for signs with two handshapes, by entering one into "Handshape" and the other one into one of the three other fields. If you placed the two handshapes into one field, only those signs where the notation contains the two symbol strings concatenated would be found. (Which in most cases would mean incorrect HamNoSys.)

o Productive signs and other forms not contained in the lexicon can be added by entering HamNoSys: Click on "Form".

While on the Macintosh the keyboard switches automatically to the HamNoSys layout, no comparable assistance is available on Windows. However, you can once again use the HamNoSys input palette to assist you in typing HamNoSys.

These signs will show up in the utterance's list of signs with the gloss "$PROD".

o "Alpha" allows you to add fingerspelling to the utterance.
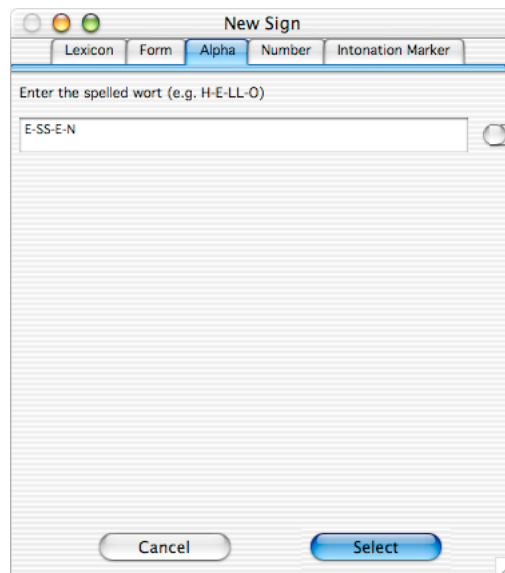
Type in the word letter by letter and press on the button to the right, or type in the letters separated by hyphens to signal what to sign, e.g. H-E-LL-O if you want a double-L sign to be shown. The button tries to achieve the same: It uppercases the letters (except ß) and inserts hyphens between the letters except between doubles without movement.

The fingerspelling will show up in the utterance's list of signs with a gloss beginning with "$ALPHA:".

o "Number" allows you to conveniently add numbers and number incorporation signs. Please note that you will probably also be able to add the number signs by selecting them from the lexicon.
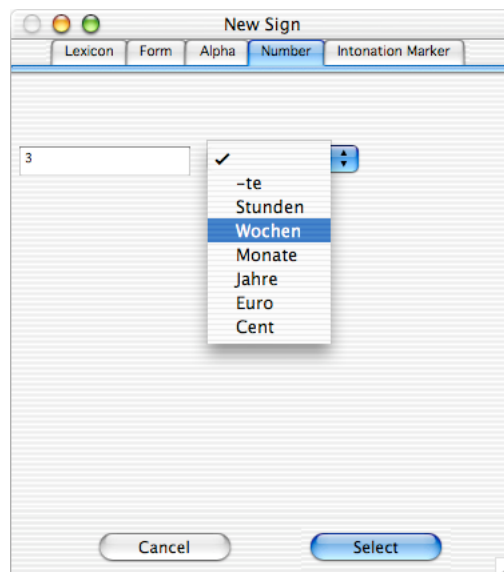
Type the number into the text field. The pop-up menu then allows you to select number incorporating signs that would fit with the number entered. Leave the menu option blank if you just want the number sign.

These signs will show up in the utterance's list of signs with a gloss beginning with "$NUM:".

o "Intonation Marker" allows you to add discourse structuring signals into the utterance, independent of whether they consist of a single sign or a sequence.
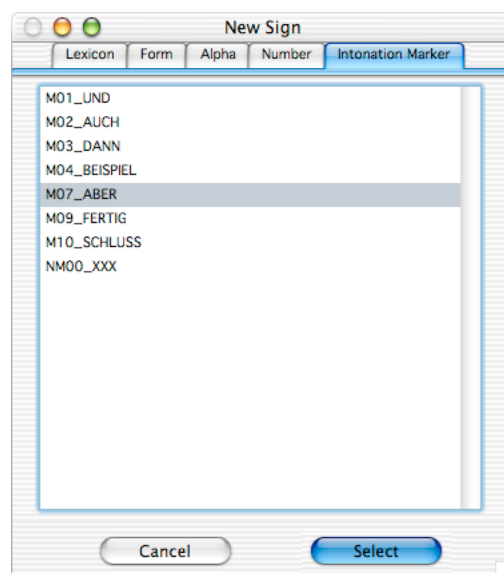
Just select the signal you need from the list.

Please note that the list of markers shown here can be edited with the eSIGN Editor program itself. Cf. appendix.

Now click "Select" to close the window and insert the sign selected into the utterance, or "Cancel" to close the window without inserting anything into the utterance.

By doubleclicking on the gloss, you can add comments regarding the selected sign.

Next you can change a number of parameters of the sign: Mouth, source and goal or location (for signs that are tagged to allow these modifications), hand (classifier, actually more than just the handshape), facial expression, and HamNoSys: Just doubleclick on the cell that you want to change. (Cells that are grey indicate that the lexicon considers selecting a value of that kind inappropriate for the sign. If you want to assign a value nevertheless, hold the ctrl key when doubleclicking).
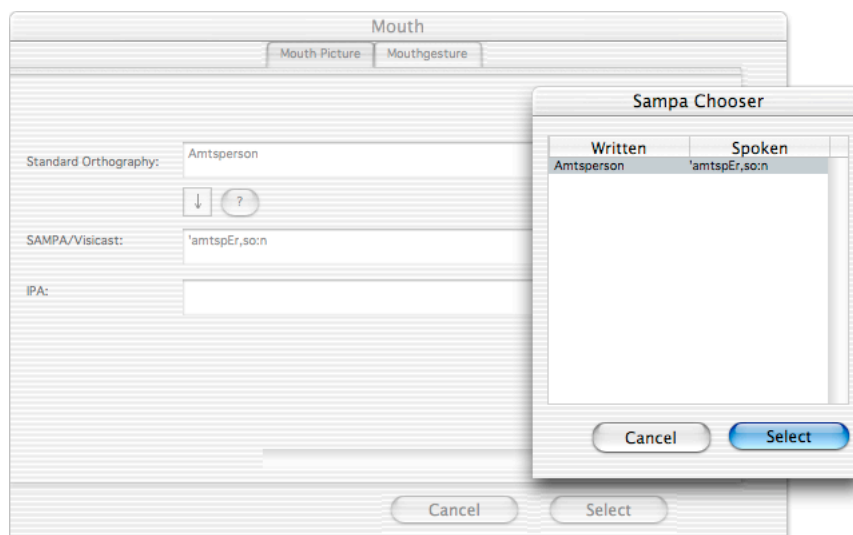
- o Mouth: You will open this dialog if the lexical entry has not been specified with a mouth picture or mouth gesture or not that one you prefer in the given context. By clicking on the tabs you can choose between entering a mouth picture or a mouth gesture:

To specify a mouth picture, you can either directly enter it in SAMPA encoding (cf. ViSiCAST deliverable 5-1) or use the pronunciation database: Enter a spoken language word in the upper text field, and have it convert into SAMPA by clicking the downwards arrow button. If you need a shortened mouth picture, still enter the full orthography, and then shorten the resulting SAMPA code appropriately. If you enter a partial word, you can use the "?" button to list the pronunciation database entries matching. Select one from the list to have SAMPA code inserted.
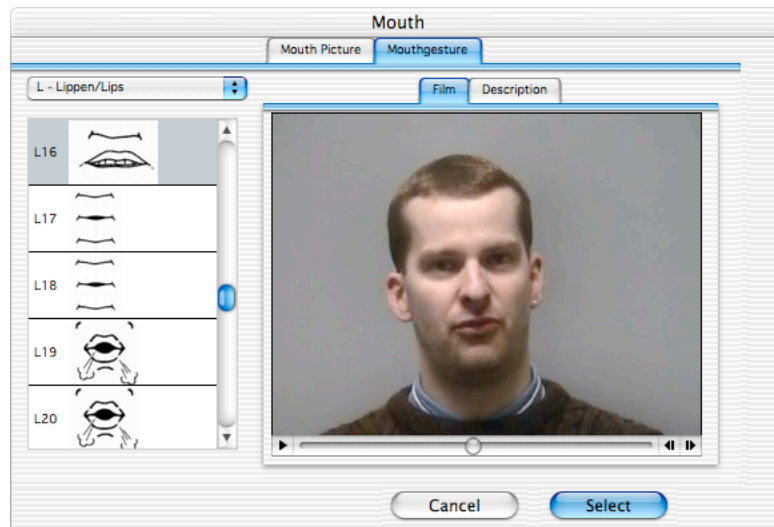
Mouth gestures are grouped by the main articulator: Teeth, jaw, lips, cheeks, or tongue. If you select one of these from the pop-up menu on the upper left of the window, all available gestures show up in the list below with their codes and drawings. Click on one item in that list to get a movie showing the mouth gesture and a textual description. (Switch between the two by clicking on the "Film" and "Description" tabs. The descriptions match those in the ViSiCAST deliverable 5-1.)
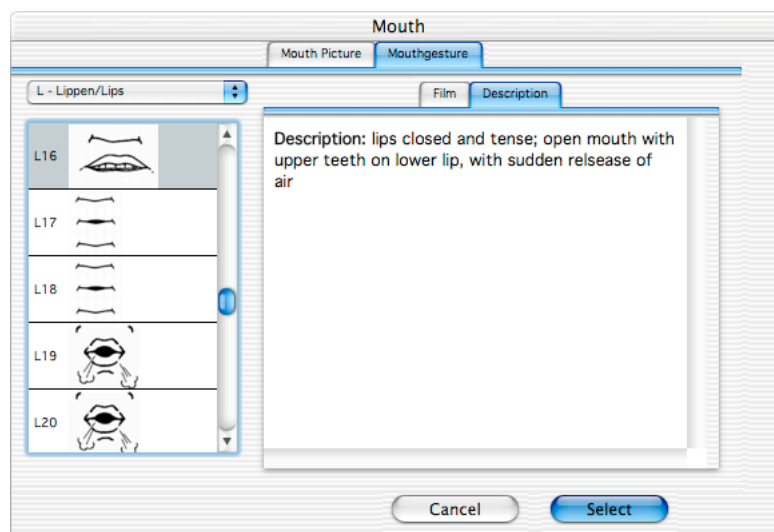
- o SRC (Source) and GOL (Goal): Specify a position in signing space to be used for source/goal by clicking: Your click will be rounded to the next available position representing 1st person, 2nd person, 3rd person on the ipsilateral side, and 3rd person on the contralateral side.
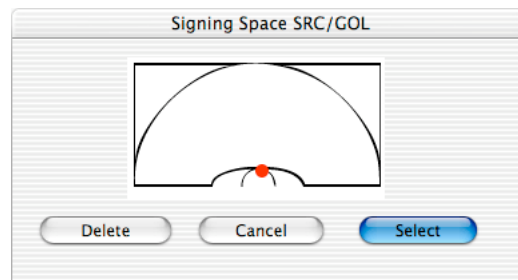
- o LOC (Location): Specify a position by clicking on both the horizontal and the vertical grid: Your click will be rounded to the next available position expressible in HamNoSys. Click on "Delete" to return to the default location.
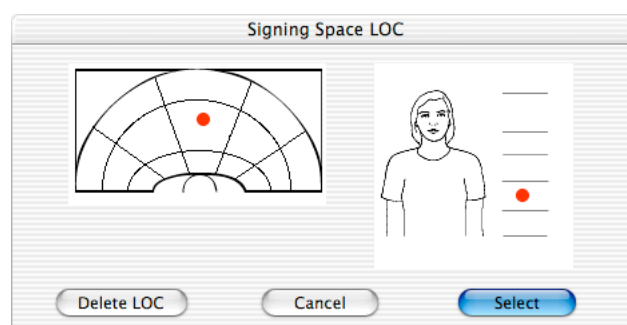
- o Hand: If you click on a handshape in the left list, the right list will show all classifiers in the database with that handshape. The classifier consists of the handshape, orientation, and handedness, as shown by the HamNoSys values. To learn how exactly the classifiers looks without reading the HamNoSys, have the avatar show it to you by clicking on the "V!" button. (For more information on the classifier definition used here as well as on the names, refer to the ViSiCAST WP5 deliverables.
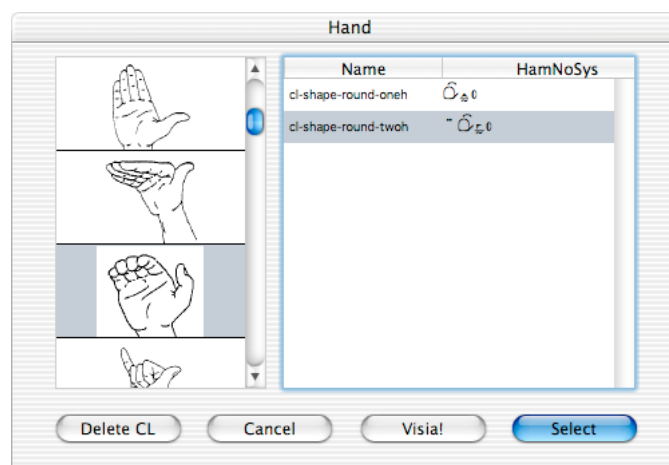
- o Limbs: Here you can specify body, shoulder, and head movement by selecting values from the menus right to the text fields. The values

stem from ViSiCAST D5-1 where you can find extensive explanations.
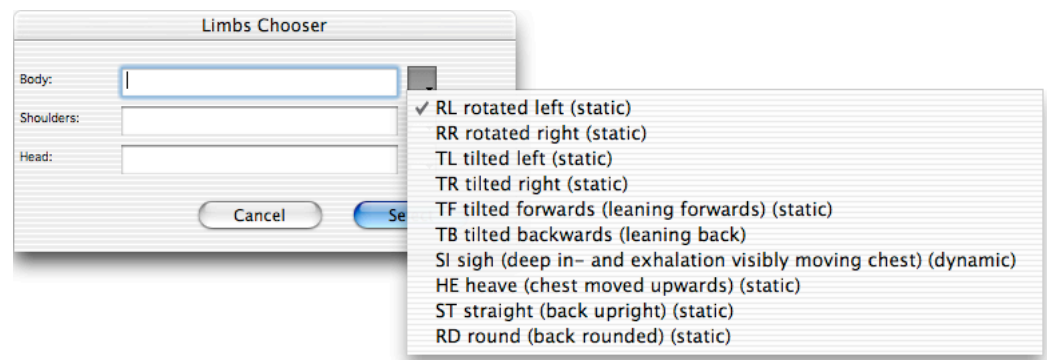
- o Facial Expression: Here you define the avatar's facial behaviour except for the mouth. Select values from the menus right to the text fields to specify eye gaze direction, eye brows and lids movements as well as movements of the nose. Once again, the values stem from ViSiCAST D5-1.
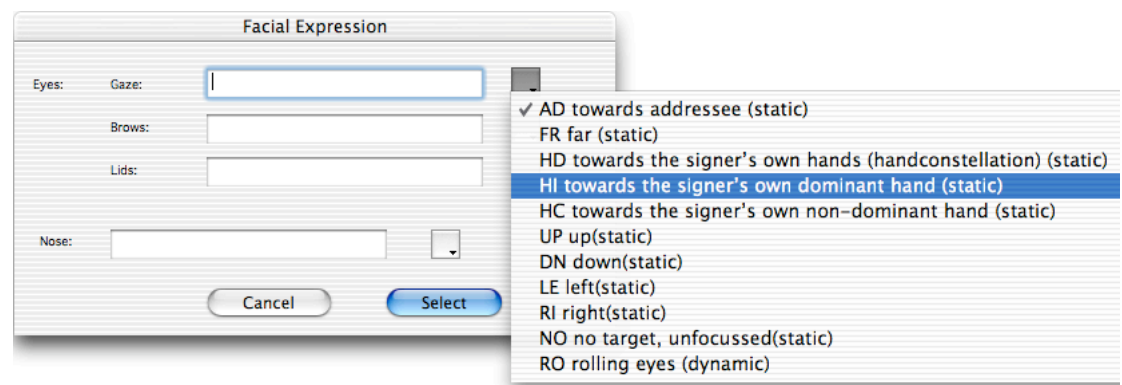
- o HamNoSys: If you want to change manual aspects of the sign not covered elsewhere, you can directly manipulate the HamNoSys value. To assist you in typing in HamNoSys, the palette opens if you click on "…".
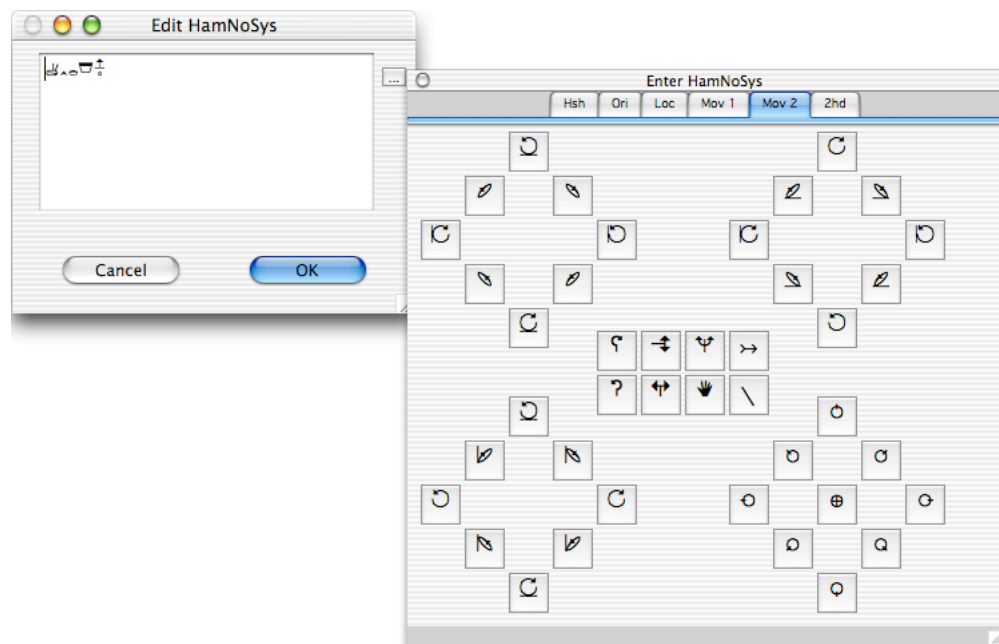
Please note that if you have manually changed a HamNoSys value, this entry will not automatically update if the sign's lexical entry is revised.

The easiest method to fill in a value that is not yet specified is to "copy" it from somewhere else. Option-click (Win: alt-click) into the receiving field and drag a line into the defining field.

Dragging can be within a column to copy a value from another (possibly temporally preceding) cell, or between compatible cell in different rows and columns. The latter applies to source, goal, and column. What precisely is added to the receiving cell depends on the columns involved: It might simply be a copy of the value (e.g. location to location), or a temporal extension of the previous value (e.g. /HOLD/ for mouth).

4.3   Talking to the avatar

Utterances or parts thereof can be sent to a SiGMLAnimation Service to immediately view the avatar playing back the constructed signed text by clicking the "V!" button available at the appropriate places in the program.

The IP address of the SiGMLAnimation service talked to is stored in the text file *VisiaIPAddress.txt* which in the package distribution is preset to 127.0.0.1 (localhost). The service can, however, reside on another machine. (In fact, this is currently required if the editor is running on a Macintosh.)

If you hold the Option key (Win: alt) pressed when clicking on "V!", the program will open a window with the SiGML string to be sent. You can then review or even alter the text before actually sending it.
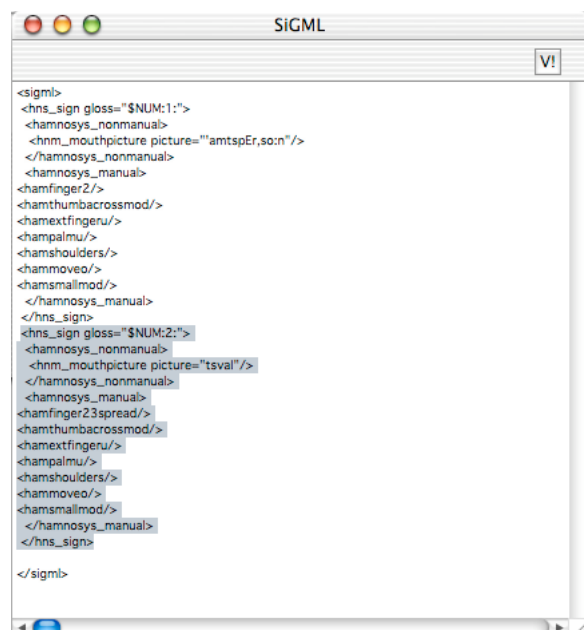
While it is possible to copy SiGML from that window to be used elsewhere, you can also use "Export as SiGML…" in the "File" menu to save SiGML corresponding to the selected utterances.

4.4   Database maintenance

While the signed texts created with the eSIGN Editor are stored in document files, a database is used to store the lexicon, the list of classifiers as well as the SAMPA pronunciation data. The editor provides functions to batch populate the database:

If you choose "Database Maintenance…" in the "Maintenance" menu, a window will open offering three import functions. Each will empty one table in the database and then populate that table from the contents of a text file with a predefined name available in the same directory as the program. For the formats and names of these files, please refer to the appendix.

Attention: Importing signs will mean that all sign entries that you created inside the editor will be lost!

To work as a stand-alone solution, the editor also provides the functionality to extend and revise the lexicon, i.e. the database table of signs:

Choose "Update Signs…" from the "Maintenance" menu.

To insert a new sign into the lexicon, choose "Insert".

In the window opening now, fill in the required data:

> The gloss is used a convenient name for the sign. In most cases you will use a gloss name approximating the semantics of the sign. The idea is that glosses are unique in the lexicon. This is, however, not endorsed by the editor.

> The POS (part of speech) value currently is just for information of the user when browsing through a list of possible signs. It is recommended to use abbreviations such as "N", "V" etc. or to leave the field empty.

> In order to enter a HamNoSys description of the sign (in its citation form), you can open the HamNoSys palette window by clicking on the "…" button. For locatable signs, you can still enter complete HamNoSys including location information. The location information will then be used as the default value until you specify another location for the sign within an utterance.

> To enter a default mouth picture or a mouth gesture associated with the sign, click on the "…" button to the view the editor described above.

> Check the appropriate boxes for directional signs (taking source and/or goal) and locatable signs. Checking the box here will allow you to use the

corresponding editors to specify source, goal, or location once you insert this sign into an utterance.

Click "Insert" to store the new sign in the lexicon and to close the window, or "Cancel" to close the window without storing the sign.

Before you can update or delete a lexical entry, you have to find it: Enter a (partial) string in the "Gloss" field and click "Search". Then select the entry to be changed or deleted and press the appropriate button. For updating a sign, doubleclick on the value to be changed. Boolean values will toggle. For other data, a dialog will open where you can change the data.

Please note that from the list of signs resulting from a search you can drag a sign and drop it onto an utterance window to have the sign inserted into the utterance.

4.5   Help

If you need help with the eSIGN Editor, have a feature request or bug to report, please send us an email (mailto:esign-editor@sign-lang.uni-hamburg.de). If you have questions about HamNoSys or how to encode lexical entries, please turn to mailto:hamnosys-helpdesk@sign-lang.uni-hamburg.de.

5   References

Gibbon, D., et al. (eds). 1997. *Handbook of Standards and Resources for Spoken Language Systems*. Berlin: de Gruyter.

Hanke, T., and C. Schmaling (eds.). 2002. ViSiCAST Milestone M5-15: User documentation.

Pizzuto, E., and P. Pietrandrea. 2001. The notation of signed texts: Open questions and indications for further research. *Sign language and linguistics* 4 (1/2):29-45.

Portele, T., J. Krämer, and D. Stock. 1995. Symbolverarbeitung im Sprachsynthesesystem Hadifix. In: *Proc. 6. Konferenz Elektronische Sprachsignalverarbeitung 1995, Wolfenbüttel*, 97-104.

Prillwitz, S., et al. 1989. *HamNoSys Version 2.0: Hamburg notation system for sign languages: An introductory guide*. Hamburg: Signum.

Schmaling, C., and T. Hanke. 2001. HamNoSys 4.0. (http://www.sign-lang.uni-hamburg.de/Projekte/HamNoSys/

6   Appendices

6.1   Appendix: eSIGN Editor file format

eSIGN Editor documents are stored in XML format that closely matches the data exposed to the user during the editing process: A signed text consists of several utterances, each consisting of a number of signs. Each sign is described by several attributes, the most important being HamNoSys phonetic description. An annotated DTD reads as follows:

```
<!--  SignedText100.dtd

      2003-08-13 Thomas Hanke, University of Hamburg
      2003-09-30 Thomas Hanke, University of Hamburg: added attlists for
src, gol,          loc, hamnosys
-->

<!ELEMENT signedtext (utterance*) >
<!ATTLIST signedtext
      database CDATA #IMPLIED
      language CDATA #IMPLIED
      spokenlanguage CDATA #IMPLIED
      title CDATA #IMPLIED
>
<!--  database for us contains "ilex.sign-lang.uni-hamburg.de", i.e. an
unqualified        URL to identify the database host.
      language should contain a value such as "sgn-de", "sgn-nl", "sgn-gb"
etc.         from RFC 3066 (extensions to ISO 639).
      spokenlanguage uses ISO 639-2 to identify the written text on the left
side         of eSIGN Editor documents, i.e. "deu" etc.
-->

<!ELEMENT utterance (sign*) >
<!ATTLIST utterance
      spoken CDATA #IMPLIED
>

<!ELEMENT sign (signdescription?, mouth?, src?, gol?, loc?, limbs?,
facialexpression?, hamnosys?) >
<!ATTLIST sign
      gloss CDATA #REQUIRED
      signid CDATA #IMPLIED
>

<!ELEMENT signdescription (#PCDATA)* >
<!--  Any text the user wants to store along with this sign, e.g. why a
      modification was applied. -->

<!ELEMENT mouth (#PCDATA)* >
<!--  Sampa for mouth pictures or ViSiCAST mouth gesture codes enclosed in
square        brackets -->

<!ELEMENT src (#PCDATA)* >
<!ATTLIST src
      editable CDATA #IMPLIED
>
<!--  src is 1, 2, 3, or 3c. editable=false means the field is displayed
grey in      the editor and a value cannot be set. -->

<!ELEMENT gol (#PCDATA)* >
<!ATTLIST gol
      editable CDATA #IMPLIED
>
<!--  loc is 1, 2, 3, or 3c. editable=false means the field is displayed
grey in      the editor and a value cannot be set. -->
```

```
<!ELEMENT loc (#PCDATA)* >
<!ATTLIST loc
      editable CDATA #IMPLIED
>
<!--  loc is the location of the (locatable) sign, encoded in comma-
separated    HamNoSys tokens. editable=false means the field is displayed
grey in the   editor and a value cannot be set. -->

<!ELEMENT limbs (body?, shoulders?, head?) >

<!ELEMENT body (#PCDATA)* >

<!ELEMENT shoulders (#PCDATA)* >

<!ELEMENT head (#PCDATA)* >
<!--  all use the Visicast nonmanual codes (two-letter abbreviations) -->

<!ELEMENT facialexpression (#PCDATA)* >
<!--  uses the Visicast nonmanual codes (two-letter abbreviations) -->

<!ELEMENT hamnosys (#PCDATA)* >
<!ATTLIST hamnosys
      manual CDATA #IMPLIED
>
<!--  HamNoSys encoded in comma-separated tokens -->
<!--  manual can be true or false. Manual=true means that the HamNoSys value
of the        entry has been manually entered and therefore should not be
automatically      updated from the database. -->
```

6.2   Appendix: Supporting files

The eSIGN Editor requires a number of files to be located in the same directory (folder) as the application:

EditorData is the RealDatabase file to store signs, SAMPA pronunciation, and classifiers. It is not required if another database adaptor than RealDatabase is used.

HamSeparators.txt, *HamSymbols.txt*, *Layout.txt* and *VisiaIPAddress.txt* have the same content as in the Ham2HPSG package. Their format and meaning is described in the accompanying documentation. Please note that the identical format of the *Layout.txt* file means that customised layouts of the HamNoSys palettes can be shared between the two tools.

The *CL* subdirectory (folder) contains GIF files for some hundred handshapes as displayed in the Hand (classifier) editor. Filenames are derived from the HamNoSys description of the handshapes. A converter between filenames and HamNoSys handshape descriptions is part of the Ham2HPSG program.

The *Mouth* subdirectory (folder) contains pictures, movies, and descriptions of the ViSiCAST set of mouth gestures that are displayed in the mouth picture/mouth gesture editor.

The file *IntonationMarks.xml* containing the list of discourse structuring signals available in the corresponding editor is an eSIGN Editor XML file, i.e. it can be edited in the eSIGN Editor itself: Intonation Marks are treated as utterances, i.e. as sequences of signs. The spoken language phrases linked with the utterances are used as the labels appearing in the editor.

6.3   Appendix: Import formats

On the database maintenance panel, buttons allow the user to populate the database from text files. This appendix explains the expected format of the text files. Please note that the respective database file is completely emptied at the beginning of the import process. The character encoding of the import files is auto-guessed during the import process, so anything like plain ASCII or UTF8 should be ok. The files are supposed to be located in the same directory as the application.

classifiers.txt is a tab-text file with two columns:

1	name of the classifier
2	HamNoSys for the classifier in comma-separated token format

sampa.txt is a tab-text file with two columns:

1	standard orthography of a spoken language word
2	pronunciation of that word in SAMPA encoding

import1.txt is a tab-text file to batch-enter signs into the database. The file consists of the following columns:

1	signID: uniqe numerical identifier
2	gloss: assumed to be unique, but not really required
3	HamNoSys for the sign in comma-separated token format
4	part of speech descriptor, may be empty
5	default mouth picture/mouth gesture for the sign. Mouth pictures are encoded in SAMPA, mouth gestures are defined by the ViSiCAST code enclosed in square brackets.
6	allowSRC: 0 or 1. If 1, the sign can be modified by specifying a source.
7	allowGOL: 0 or 1. If 1, the sign can be modified by specifying a goal.
8	allowLOC: 0 or 1. If 1, the sign can be modified by specifying a location.

If another database other than the single-user RealDatabase is used, the database can of course be populated by more sophisticated processes.

6.4   Appendix: Platforms supported

The eSIGN Editor currently is available for Win32 and MacOS X. As it internally works with Unicode only, older versions of the operating systems (i.e. Win95, Win98, WinME, and MacOS 9) are not supported. In the future, a Linux version might become available if required by one of the partners.

The eSIGN Editor has been implemented using RealBasic. In this environment, a couple of database connectors are available. It is therefore possible to replace the default single-user RealDatabase with one of the following:

MySQL

Frontbase

Postgresql

Any database supporting ODBC (including FileMaker and MS SQL).

In addition, modules are available for purchase to connect to Valentina, Sybase, OpenBase, PrimeBase. Please note that a change of the database connector currently requires recompilation of the tool. Therefore, please address any request for special versions to mailto:esign-editor@sign-lang.uni-hamburg.de stating the combination of user interface language, target sign language, operating system, and database vendor that you need.