intElligent Automatic
SIgn languagE tRanslation

D6.6 TOOLS FOR HARMONIZING AVAILABLE ANNOTATIONS TO A COMMON FORMAT

Revision: v1.0

Work Package	WP6
Task	T6.2
Due date	30/06/2022
Submission date	30/06/2022
Deliverable lead	Universität Hamburg
Version	1.0
DOI (latest version)	10.25592/uhhfdm.10262
Authors	Marc Schulder, Sam Bigeard, Maria Kopf, Thomas Hanke (Universität Hamburg – UHH)
Reviewers	Özge Mercanoğlu Sincan (UNIS), Felix Burkhardt (AUD)

Abstract	This deliverable provides tools for harmonizing available annotations to a common interchange format. It defines the interchange format and provides example implementations for two corpora. It also introduces EASIER Notation, a new human- and machine-readable format for data exchange between machine translation and avatar components, and implements a converter from interchange format to EASIER Notation.
Keywords	Data Harmonisation, Corpus Annotation, File Format Conversion, ELAN, iLex, EASIER Interchange Format, EASIER Notation

Grant Agreement No.: 101016982
Call: H2020-ICT-2020-2
Topic: ICT-57-2020
Type of action: RIA

Document Revision History

Version	Date	Description of change	List of contributors
V0.1	28/06/2021	First draft	Marc Schulder, Sam Bigeard, Maria Kopf, Thomas Hanke (UHH)
V0.2	30/06/2021	Internal Review	Özge Mercanoğlu Sincan (UNIS) Felix Burkhardt (AUD)
V1.0	30/06/2021	Camera-ready submission DOI: 10.25592/uhhfdm.10263	Marc Schulder, Sam Bigeard, Maria Kopf, Thomas Hanke (UHH)

DISCLAIMER

COPYRIGHT NOTICE

Project co-funded by the European Commission in the H2020 Programme		
Nature of the deliverable	**R**	
Dissemination Level		
PU	Public, fully open, e. g., web	✓
CL	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to EASIER project and Commission Services	

* R: Document, report (excluding the periodic and final reports)
DEM: Demonstrator, pilot, prototype, plan designs
DEC: Websites, patents filing, press & media actions, videos, etc.
OTHER: Software, technical diagram, etc

Funded by the Horizon 2020 Framework Programme of the European Union

EXECUTIVE SUMMARY

This deliverable provides tools for harmonizing available annotations to a general interchange format, which can then be used to convert them to formats for specific applications. It consists of a report and associated open-source code. The report outlines the schema for that interchange format as well as introducing EASIER Notation, a new human- and machine-readable format for data exchange between machine translation and avatar systems. The software code associated with this report provides converters from BSL Corpus and Public DGS Corpus to the interchange format and a converter from interchange format to EASIER Notation.

Funded by the Horizon 2020
Framework Programme of the European Union

CONTENTS

Funded by the Horizon 2020
Framework Programme of the European Union

LISTINGS

Funded by the Horizon 2020
Framework Programme of the European Union

LIST OF TABLES

Funded by the Horizon 2020
Framework Programme of the European Union

1 INTRODUCTION

This deliverable concerns the harmonisation of annotation data from different sign language corpora. Annotation conventions can differ significantly between corpora[1], creating barriers to the aggregation and combination of datasets for multilingual applications. Apart from being available in different file formats (primarily iLex or ELAN files), corpora may give different gloss naming schemes, sign type hierarchies, annotation tier structures, and more.

At the same time, various applications for the processing of signed languages, such as specific sign recognition, machine translation and avatar tools, may also have different requirements for their data input. To establish compatibility between a corpus dataset and a processing tool, a converter from source to target format is required.

To avoid the duplication of efforts that the development of independent source-target converters for each corpus-tool pairing would cause, we propose the introduction of an intermediate **interchange format**. This interchange format functions as a pivot, allowing the conversion from various corpora to interchange and then from interchange to the desired output format. As such it is designed to hold all information that may be relevant to computational processing.

This report documents the first version of the interchange format. It also introduces the **EASIER Notation**, a format for data exchange between the machine translation and avatar components of the EASIER pipeline.

To illustrate the use of both formats, we release open-source code for the following conversion scripts:

1. Conversion from BSL Corpus (Schembri et al., 2013) to interchange format;
2. Conversion from Public DGS Corpus (Hanke et al., 2020) to interchange format;
3. Conversion from interchange format to EASIER Notation.

The code for these scripts can be found at https://doi.org/10.25592/uhhfdm.10266.

The interchange format and EASIER Notation will be developed further as converters for individual corpora are written and experience with integrating information into the EASIER pipeline is gathered. Regarding the interchange format, while not all of the information that can be extracted from the various corpora will end up being used in the EASIER translation pipeline, all of it must be parsed and understood to some degree to correctly simplify and filter it. We also believe that providing as much information as possible will help with the development of additional output converters and usage of the interchange format as a direct input.

Chapter 2 describes the resulting structure of the interchange format, detailing how sign types (Section 2.1), sign occurrences (Section 2.2), mouthings (Section 2.4) and translations (Section 2.3) are encoded and the hierarchy of subcategories for each of them. Chapter 3 introduces the EASIER Notation.

[1] For an extensive comparison of different annotation conventions in corpora for European sign languages and a general discussion of the design rationale behind the interchange format, see EASIER Deliverable D6.2 *"Specification for the Harmonization of Sign Language Annotations"* (Kopf et al., 2022).

Funded by the Horizon 2020 Framework Programme of the European Union

2 INTERCHANGE FORMAT

The interchange format is designed to represent corpus annotations from various corpora and lexical type information for signs.

In sign language corpus annotations, glosses function as a text-based aid to humans. As such they need to encode information in a succinct way that allows the perusal of sequences of glosses. The EASIER interchange format, on the other hand, is primarily intended for the communication between components of a software pipeline and the conversion into other formats of varying expressivity. Therefore, it has different constraints and priorities. Information does not need to be encoded as succinctly, but needs to be easily and unambiguously parseable. So, instead of representing a sign through a single gloss word, we propose using a JSON container structure that explicitly identifies individual components of the gloss, such as its name and variant, plus relevant meta-information like its source language and dataset.

The exact structure of a container depends on the *kind* of information that it represents. Lexical type entries provide general information for a sign, while token entries provide information on specific instances in a particular recording. Recording tokens can represent a variety of concepts, such as occurrences of signs, occurrences of mouthings and mouth gestures, or translations of sign sequences.

The kind of container can be further specified, e. g. entries for basic lexical sign types provide different information than special types, such as those for buoys, fingerspelling, etc.

The following sections provide examples of how different kinds of containers are structured in version 0.2 of the interchange format. The format will be extended and refined further as converters for more corpora are added and generalisable representations of specific linguistic phenomena are developed.

2.1 TYPES

Type entries represent the context-independent base form of a sign. They represent lexical information that is independent of specific recordings or particular instances of a sign. Depending on the annotation file format, type information may be included in the annotation file or is retrieved from a separate lexical database. To connect types with sign tokens (specific occurrences of a sign), a shared identifier is used, usually either a numeric ID or an ID-gloss.

2.1.1 Type Entry for a Regular Sign

The most common type entry is that of a regular lexical sign, i. e. one whose gloss centres around a spoken language word approximating the sign meaning, as opposed to specially glossed types like buoys or fingerspelling.

The example shown in Listing 2.1 encodes the type gloss *WHIPPED-CREAM1*, taken from the Public DGS Corpus.[2]

[2] https://doi.org/10.25592/dgs.corpus-3.0-type-13156

Funded by the Horizon 2020 Framework Programme of the European Union

```
{
  "id": "type_mydgs_4670",
  "kind": ["type", "regular"],
  "parent": "type_mydgs_13156",
  "language": "gsg",
  "glossname": {
    "deu": "SCHLAGSAHNE",
    "eng": "WHIPPED-CREAM"
  },
  "lex variant": 1,
  "phon variant": null,
  "hands": 1,
  "hamnosys": ["hamfist", "hamindexfinger", "hamfingerhookmod", "
     hamextfingerol", "hampalml", "hamcirclel", "hamsmallmod", "
     hamrepeatfromstartseveral"],
  "wordnet": ["07621388-n"],
  "source": {
    "id": "4670",
    "parent": "13156",
    "dataset": "My DGS - annotated",
    "gloss": {
      "deu": "SCHLAGSAHNE1",
      "eng": "WHIPPED-CREAM1"
    }
  },
  "schema": "0.2"
}
```

Listing 2.1: *Example of a container for the corpus-independent representation of a type entry for a regular lexical sign.*

id: A unique ID of the container. Based on its identifier in the source material, but marked additionally to be unique across datasets as well.

parent: If the type is a child element in a hierarchical type structure, this is the ID of the parent type. The Public DGS Corpus, which employs double glossing (see Konrad et al., 2022) uses this field. For example, the type shown in Listing 2.1 is a subtype of *TO-STIR1^*.

language: The language of the sign, identified by its ISO639-3 language code.

kind: Specifies that the container represents a type entry (as opposed to a token instance) and that the type is a regular sign (as opposed to a specially glossed type like a buoy or fingerspelling). For a list of possible values, see Section 2.5.

glossname: The component of a regular sign that is based on a spoken language word. The spoken language used is identified using its ISO639-3 language code. This permits the inclusion of glosses in multiple languages.

lex variant: A value disambiguating lexical variants: distinct signs that are considered to be semantically synonymous (to the extent this can be determined during glossing). For conventions

Funded by the Horizon 2020 Framework Programme of the European Union

that encode additional information for variants (words, handshape codes) these additional parts are removed from the value and instead encoded in additional fields.

phon variant: A value disambiguating phonetic variants of a sign: Signs differ in some aspects of their form, but are considered to be related, belonging to the same lexical variant.. Value conversion is handled like for `lex variant`.

hands: Integer specifying whether the base form of the sign is performed in a one-handed (1) or two-handed (2) manner. For token entries, a similar field will specify whether a sign is executed with the dominant, non-dominant or both hands.

hamnosys: The phonetic transcription of the sign type in HamNoSys format. As HamNoSys symbols require a special font to display them, they are instead provided as the ASCII names of each symbol. The sequence in this example equates to "".

wordnet: Wordnet sense identifiers, e. g. taken from the EASIER interlingual index (see Bigeard et al., 2022) that integrates with Open Multilingual Wordnet (Bond and Paik, 2012). Lists the different possible meanings the sign can have.

source: A sub-container providing information on the original corpus data on which the type entry is based. These fields are included mainly to allow developers to verify whether the gloss was parsed correctly and to refer back to the data.

id: The unique identifier of the type entry that is used in the original corpus. Unlike the interchange ID, this might not be unique across datasets. In some corpora it is identical to the gloss string.

parent: The corpus ID of the parent type (see EASIER field of same name).

dataset: Identifier of the corpus from which the entry originates.

gloss: The original gloss string(s) of the type entry as they were used in the corpus. Like the `name` field, it allows for multiple language variants.

schema: Version of the interchange format that is used by the container.

2.2 SIGN TOKENS

Sign tokens represent the specific production of a sign in the context of an utterance. They must specify which recording transcript they originate from, locate the token in the time frame of the recording, connect it to a type entry, and specify whether and how the token deviates from its base form.

2.2.1 Token Entry for a Regular Sign

This is the representation format for most sign tokens, except for special cases such as buoys, fingerspelling, etc.

An example of a regular sign container can be seen in Listing 2.2. It shows the container for a token occurrence of the *WHIPPED-CREAM1* sign type that is described in Section 2.1.1. It is structured as follows:

Funded by the Horizon 2020 Framework Programme of the European Union

```
{
  "id": "token_mydgs_1001435",
  "kind": ["token", "lexeme", "regular"],
  "transcript": "transcript_mydgs_1250646",
  "person": "person_mydgs_MVP-05",
  "type": "type_mydgs_4670",
  "is modified form": true,
  "hand": "right",
  "timecode start": [498, 11, 50],
  "timecode end": [498, 36, 50],
  "source": {
    "id": "3250012",
    "dataset": "My DGS - annotated",
    "transcript": "1250646",
    "person": "MVP-05"
  },
  "schema": "0.2"
}
```

Listing 2.2: *Example of a container for the corpus-independent representation of a token entry of a regular lexical sign.*

id: A unique ID of the container. Based on its identifier in the source material, but marked additionally to be unique across datasets as well.

kind: Specifies that the container represents a token (as opposed to a type entry), which kind of token (sign, mouthing, translation,etc) and whether it is a regularly sign or a special kind, such as for a buoy or fingerspelling. For a list of possible values, see Section 2.5.

transcript: ID of the recording transcript to which this token belongs. Based on the identifier or filename of the source transcript, but marked additionally to be unique across datasets as well.

person: ID of the person who produced the sign. Based on the person identifier in the source material, but marked additionally to be unique across datasets as well. Might be omitted for transcripts covering only a single person.

type: The ID of the type entry that represents the base form of the token.

is modified form: Specifies whether the way the sign is executed differs from how it was defined in the type entry. If the corpus also provides information on how it differs (e. g. different handshape or a one-handed sign being performed with both hands), this is specified in the additional field `modification`.

hand: Which hand is the dominant hand in producing the sign. May be "right" or "left" for one-handed and asymmetric two-handed signs and "both" for symmetric two-handed signs.

language: The language of the sign, identified by its ISO639-3 language code.

timecode start: Specifies where in the transcript recording the production of the sign begins. The timecode is given as a list of 3 integers representing seconds, and a numerator/denominator pair to specify a sub-second value. The sub-second denominator can match the frame rate of the video recording (50fps in the example) or be set to 1000 to represent milliseconds. It

Funded by the Horizon 2020 Framework Programme of the European Union

should be noted that depending on the annotation conventions of a corpus, the definition of where a sign starts and ends may differ, particularly regarding whether transitional movements are seen as part of the sign.

timecode end: Timecode for where the production of the sign ends.

source: A sub-container providing information on the original corpus data on which the token entry is based.

id: The unique identifier of the type entry that is used in the original corpus, if available. Not all data formats provide unique IDs for glosses, and unlike the interchange ID, such IDs might not be unique across recordings or datasets.

gloss: If the source data specifies the gloss explicitly for every token (instead of referring to the ID of a type entry), this gloss string is given here. Like the `name` field of type entries (see Section 2.1.1), it allows for multiple language variants.

parent: The corpus ID of the supertype, if applicable (see main container field of same name).

dataset: Identifier of the corpus from which the entry originates.

transcript: Identifier of the transcript in the original corpus, if available.

person: Identifier of the person in the original transcript, if available.

schema: Version of the interchange format that is used by the container.

2.2.2 Token Entry for Fingerspelling

The previous section represented the type and token format for regular signs. As glosses for other kinds of signs encode different information, their containers provide different fields.

Listing 2.3 shows an example of a fingerspelling token. The example is taken from the Corpus LSFB and is a fingerspelling of the family name "Descornet", glossed as *FS:DESCORNTET(DESCORNET)*. The gloss specifies both the letters that were spelled out by the signer and the spelling that the annotators presume was intended. In this particular case, the difference is that the signer accidentally included a second "T" in the name.

The fingerspelling token container uses the following fields that did not occur in previous examples:

spelling: The sequence of fingerspelled letters. Represented as a list of strings to allow for fingerspelling signs that represent multiple letters, e. g. "CH" or "SCH". Correct marking of such multi-letter signs depends on whether they can be identified reliably in the given gloss format or through other methods.

text: If the corpus specifies the correct spelling of a word in addition to the actual spelling, that word is given here.

language: The language that the spelled word is taken from, given as an ISO639-3 language code.

alphabet: Identifier for the fingerspelling alphabet that is used. The identifier will refer to a set of known fingerspelling alphabets. As some signed languages have more than one alphabet (e. g. a one-handed and a two-handed alphabets), identifiers will consist of the language name and additional disambiguating markers.

Funded by the Horizon 2020 Framework Programme of the European Union

```
{
    "id": "lsfb_CLSFBI0301_ 456789",
    "kind": ["token", "lexeme", "fingerspelling"],
    "spelling": ["D", "E", "S", "C", "O", "R", "N", "T", "E", "T"],
    "text": "DESCORNET",
    "language": "fra",
    "alphabet": "lsfb1",
    "hand": "right",
    "timecode start": [56, 957, 1000],
    "timecode end": [60, 384, 1000],
    "source": {
        "gloss": "FS:DESCORNTET(DESCORNET)",
        "id": "456789",
        "transcript": "CLSFBI0301",
        "dataset": "Corpus LSFB"
    }
    "schema": "0.2"
}
```

Listing 2.3: *Example of a container for the corpus-independent representation of a token entry of fingerspelling.*

2.2.3 Compositional Tokens

Most of the time, tokens represent a single sign. However, in some cases a token gloss actually describes a compositional expression composed of a sequence of signs. As an example, consider the following excerpt from file `BF2n.eaf` of the BSL Corpus (Schembri et al., 2017):

```
<ANNOTATION>
    <ALIGNABLE_ANNOTATION ANNOTATION_ID="a22"
    TIME_SLOT_REF1="ts68" TIME_SLOT_REF2="ts70">
    <ANNOTATION_VALUE>SN:JOHN-KING(FS:J-JOHN^KING)
</ANNOTATION_VALUE>
```

The gloss *SN:JOHN-KING(FS:J-JOHN^KING)* represents the sign name "John King", which consists of an fingerspelled J as an initialisation of the first name and a sign meaning *king* for the last name.

In such a case, we decompose the entry into a parent token of kind `["token", "lexeme", "compound"]` and several sub-tokens. The sub-tokens are treated as normal tokens, except that they have a *parent* field that points to the compound token. The sub-tokens may also be missing information on start or end time when time information is only available for the compound token.

The resulting structure for compounds and sub-tokens is shown in Listing 2.4.

Funded by the Horizon 2020 Framework Programme of the European Union

```
{
    "kind": [
      "token",
      "lexeme",
      "compound"
    ],
    "glossname": "JOHN-KING"
    "timecode start": [192,740,1000],
    "timecode end": [193,576,1000],
    "children": [
      "bslcorpus_BF2n.eaf_a22_1",
      "bslcorpus_BF2n.eaf_a22_2"
    ]
    (...)
},

{
    "parent": "bslcorpus_BF2n.eaf_a22",
    "kind": [
      "token",
      "fingerspelling",
      "initialisation"
    ],
    "alphabet-type": "bsl",
    "spelling": ["J"],
    "timecode start": [192,740,1000],
    "timecode end": null
    (...)
},

{
    "parent": "bslcorpus_BF2n.eaf_a22",
    "kind": [
      "token",
      "lexeme",
      "regular"
    ],
    "glossname": "KING",
    "timecode start": null,
    "timecode end": [193,576,1000]
    (...)
}
```

Listing 2.4: *Example of a composite token with two sub-tokens.*

Funded by the Horizon 2020
Framework Programme of the European Union

```
{
    "id": "bslcorpus_BL14n_nurtransl.eaf_a119",
    "kind": ["translation"],
    "timecode start": [322,830,1000],
    "timecode end": [324,810,1000],
    "text": "But I have him to thank for saving me.",
    "language": "eng",
    "source": {
      "transcript": "BL14n_nurtransl.eaf",
      "dataset": "bslcorpus"
    }
  "schema": "0.2"
}
```

Listing 2.5: *Container for a translation token entry.*

2.3 TRANSLATION TOKENS

Translation tokens represent spoken language translations of a signed utterance. An example is shown in Listing 2.5. Their timespan usually covers that of several signs. Translations for different languages are given as separate tokens, as in some corpora they may cover different timespans.

Apart from general fields such as IDs and timecodes, each translation token specifies:

text: The written translation.

language: The language that the signed utterance is translated into, given as an ISO639-3 language code.

2.4 MOUTHING TOKENS

Mouthing and mouth gestures that are used in coordination with manual signs are often annotated independently. This allows the annotation of mouthings that span multiple signs or otherwise diverge from being fully aligned with a lexeme token.

An example of a mouthing is shown in Listing 2.5. Fields of note are:

text: The mouthed word, spelled using the orthography of the spoken language that it is loaned from.

language: The language that the mouthed word is loaned from, given as an ISO639-3 language code.

Funded by the Horizon 2020
Framework Programme of the European Union

```
{
  "id": "mydgs_person_2109043",
  "kind": ["token", "mouth", "mouthing"],
  "transcript": "mydgs_transcript_1209006",
  "person": "mydgs_person_MUE-68",
  "timecode start": [40, 16, 50],
  "timecode end": [40, 38, 50],
  "text": "hamburg"
  "language": "deu",
  "source": {
    "id": "2109043",
    "dataset": "My DGS - annotated",
    "transcript": "1209006",
    "person": "MUE-68"
  },
  "schema": "0.2",
},
```

Listing 2.6: *Container for a mouth token entry.*

2.5 CONTAINER KIND HIERARCHY

Each container of the interchange format specifies what kind of information it contains. Kind values are structured as a hierarchy of increasingly specific subcategories. For example, a token may be a sign token or a translation token and a sign token may be a fingerspelling sign token or a buoy sign token. This section describes the kind hierarchy as defined in version 0.2 of the interchange format.

Each kind has a given set of possible fields. Subcategories of a container kind may add additional fields where necessary. In other cases, the definition of the subcategory itself represents the added information (e.g. fingerspelling must specify the letters that are spelled out, while a location name functions like other signs, but the fact that it is a location may be of use for later processing steps). How deep the subcategorisation of an entry goes also depends on how fine-grained the annotation structure of its source dataset is.

2.5.1 Sign Type/Token Kinds

Signs types always have the "kind" value `["type"]` and sign tokens have the "kind" value `["token", "lexeme"]`. Their subcategories largely overlap, so they are outlined here together:

regular

index

number

name

person

Funded by the Horizon 2020 Framework Programme of the European Union

- ***location***
  - **city**
  - **country**

date

- ***weekday***
- ***month***

buoy

- ***list***
- ***pointer***
- ***fragment***
- ***theme***

fingerspelling

- ***initialisation***

classifier

compound

gesture

- ***manual***
  - **open hand**
- ***nonmanual***
- ***extralinguistic***

cued speech

oral

- ***articulation***

unknown

uncertain

2.5.2 Translation Kinds

All translation tokens are marked as `["token", "translation"]`.

2.5.3 Mouth Kinds

Annotations of mouthed information are marked as mouthings (`["token", "mouth", "mouthing"]`) and mouth gestures (`["token", "mouth", "mouth gesture"]`).

Funded by the Horizon 2020
Framework Programme of the European Union

3 EASIER NOTATION V1

Within the EASIER pipeline the interchange format will be converted into the EASIER Notation. The EASIER Notation is an interface format interposed between machine translation output and avatar input. As EASIER Notation can be easily written by humans, it will also be used in the manual post-editing process. The EASIER Notation schema presented here is a first version and will be updated when needed.

EASIER Notation is a gloss-based notation system. Signs are specified by ID-glosses. EASIER Notation provides a flexible syntactic structure, allowing sign morphology to be specified. The gloss string is enriched with tags specifying linguistic information and affect. Some tags have an implicit scope, others have to be closed explicitly.

Different aspects of signed languages can be represented with EASIER Notation: two-handed constructions, non-manuals, constructed dialogue and more can be specified. Signs can be modified as needed to fit into a specific context. Productive signs or signs not contained in the lexical database can be written using HamNoSys, wordnet-IDs, concepts and other writing systems, allowing fallbacks into other sign languages. Intonation is implemented with commas and full stops leading to longer pauses between signs. Optional timestamps can be inserted into the text, e. g. to use it for timed data such as video "sign-over".

The EASIER Notation writing system is intended to allow encoding utterances in any signed language. This including representing phenomena that only appear in some languages. This means that some patterns provided by the EASIER Notation to encode such phenomena may not be applicable to all languages.

Section 3.1 provides examples for common types of utterances. Gloss formats are described in Section 3.2, followed by a definition of tags for encoding morphology (Section 3.3) and syntax (Section 3.4).

3.1 EXAMPLES

In the following different aspects of the EASIER Notation are discussed on the basis of example sentences.

(1) **<hold> NEWSPAPER READ COFFEE DRINK.**

- The hold instruction means that the final handshape of **NEWSPAPER** will remain held on the non-dominant hand in signing space for each subsequent one-handed sign. If that sequence contains two-handed signs, the held sign is resumed after the two-handed sign finishes.
- As there is no **</hold>**, the hold construct ends with the full stop.

(2) **<topic> BOOK, #B-O-BB-Y(loc:3a) GIVE(src:3a,gol:1).**

- **BOOK** is marked as the topic of the sentence: As there is no closing **</topic>**, the topic extends to the next comma.

Funded by the Horizon 2020 Framework Programme of the European Union

- **#B-O-BB-Y** is fingerspelled at the location 3a in signing space (the ipsilateral side). The double letter indicates a sideways movement instead of a repetition of the letter B.
- **GIVE**'s motion is from 3a to the signer (named 1).

(3) **<topic> BOOK, #B-O-BB-Y $INDEX(loc:3a) GIVE(src:3a,gol:1)**

- The same as before, but with the fingerspelling at its default location, followed by an index to locate Bobby in the signing space.

(4) **<topic> BOOK, <WH-q> YOU WANT HOW-MANY?**

- **BOOK** is again marked as the topic of the sentence, the sentence is a wh-question, marked at the end of the sentence with a question mark and specified with a question-tag in the beginning.

(5) **<indirect=1> HELLO(hd:2) <indirect=2> HELLO(mouth:hi) </indirect>**

- Example for a constructed dialogue with two interlocutors, each signing **HELLO**, one with both hands and the default mouthing for **HELLO**, 'hello', the other with just the dominant hand, but with the mouthing 'hi'.

(6) **<affect=smile:50><mouth=cinema> CINEMA GO-THERE.**

- The whole sentence is performed with a slight smile (50 being 50% of a maximum intensity of 100%).
- Here, a mouthing is stretched over two signs (or actually all the signs up to the next comma or full stop, as there is no closing **</mouth>**.

(7) **YESTERDAY, <affect=sad> $INDEX(loc:3b) LOOSE [ONE + FIVE].**

- Cases with the two hands each showing a sign at the same time, use square brackets and a plus sign to specify first the signs on the dominant hand and then the signs of the nondominant hand.

(8) **[+ ONE TWO THREE FOUR FIVE] ONE TWO THREE FOUR FIVE**

- This pattern is used to have a sequence of signs to be signed with the nondominant hand, by leaving the space for the dominant hand within the brackets empty: Counting 1–5 first with the nondominant hand, then with the dominant hand.

(9) **<list=3><item> FLOUR <item> MILK <item> SUGAR </item>**

- A list buoy specifies the number of items visible on the buoy. Then there is an item tag for each item on the list. The buoy is held until the last item is closed. The default is a list of first, second, third, and so on.

(10) **<list=3><item=1> DAUGHTER <item=3> DAUGHTER <item=2> SON </item>**

- If a deviation from the default is needed the list items can be specified using numbers.

(11) **<list=3><item> FLOUR <item> MILK <neg> COW </neg><nod> OAT, JANUARY I START <emph>VEGAN, <item> SUGAR </item>**

- One item can contain several signs.

Funded by the Horizon 2020 Framework Programme of the European Union

- The list can be paused if a list independent comment is signed. The comma only closes the item, not the whole list. With the next point the buoy is picked up again and the list is closed with the full stop at the end.

(12) **<ground>TABLE <figure>PLATE(loc:2) <figure>PLATE(loc:3a) <figure>PLATE(loc:3b) </ground>**

- **TABLE** is set as the ground on which the following figures are placed in space.

(13) **<time=14:00:00.000><neg> $INDEX(loc:3a) WORK(phs:2), <nod> $INDEX(loc:3b) WORK. <time=14:00:02.500>**

- Headshake and nodding remain active up to the next comma or full stop.
- The timestamps define that the whole utterance shall be performed in not more than 2.5 seconds. As the second timestamp is located after the full stop, the sentence will not be slowed down to exactly match 2.5 seconds if "natural speed" results in a shorter performance.

(14) **CAR $AD-HOC(hns:)**
CAR $AD-HOC(hns:hamflathand,hamextfingero,hampalmd,hammoveor,hamarcr)

- The productive use of signs, as here to show how the car drives around a corner, is written using HamNoSys tokens or characters.

3.2 GLOSS FORMATS

The EASIER Notation allows the use of different gloss formats, such as:

(1) **BANK1B**

- "ID-gloss"

(2) **BANK**

- Free to choose any sign from BANK1A, BANK1B... BANK2..., maybe one where high-quality animation data is available

(3) **{bank,shore}**

- A list of spoken language words is converted to a set of synsets by intersecting the sets of synsets for each of the words specified. Then any sign can be used that can express at least one of the synsets from the intersection.

(4) **{own-100035478,own-10153498}**

- language independent concept identifiers.

The gloss may index any lexical database entry for the target language. If a gloss like HOUSE is not contained in the database, but HOUSE1A, HOUSE1B and HOUSE2 are stored, the system will make an arbitrary choice between the signs available. Beyond lexical signs, $INDEX and some others can be used as follows:

Funded by the Horizon 2020 Framework Programme of the European Union

- **$AD-HOC** is used for productive use of signs.
- # indicates fingerspelling. Capitalized letters indicate single spelled letters, capitalized letters followed by small case letters indicate ligatures or mismatches between manual and written alphabet, e. g. 'Tt' for one T with a sideward movement, 'Sch' for the single manual letter. For cases where first name and surname are fingerspelled a hyphen is placed in between to indicate a shift in location between the two spelled words, e. g. **#JON-DOE**.

The system considers any string of characters up to the next space or opening bracket, comma or full stop as a gloss. Glosses containing any of these characters should be enclose in straight quotation marks.

3.3 MORPHOLOGY TAGS (QUALIFIERS)

As can be seen in several of the previous examples, sign glosses can be followed by one or more qualifiers (in round brackets). A qualifier consists of a keyword followed by a colon and then a value. If there is more than one qualifier, they are separated by commas.

- **src** and **gol** for source and goal, as demonstrated above. Only applicable to directional or orientable signs. If not specified, src defaults to 1 (unless gol is specified as 1 in which case src becomes 2). gol defaults to 2 (unless src is specified to be 2 in which case gol becomes 1). Possible values are 1, 2, 3a, 3b.
- **loc** for locations, as demonstrated above. Possible values are 1, 2, 3a, 3b.
- **mouth** specifies a mouthing or mouth gesture. It uses standard orthography for the spoken word. Pronunciation look-up and mapping to visemes (visible phonemes) are done by the system. For mouth actions stretching over more than one sign, the <mouth> tag is used instead.
- **hd** for number of hands, as demonstrated above. Possible values are 1, 2.
- **phs** for phases, as demonstrated above. The first execution of the sign is phase one, each further repetition is counted on top. A sign with three circular movements would be phs:3. Possible values are 1, 1.5, 2, 2.5, 3, and so on.
- **assim** marks a sign to be assimilated with the preceding sign.
- **hns** for HamNoSys as demonstrated above. Only applicable to $AD-HOC. Possible values are HamNoSys characters and tokens.

3.4 SYNTAX TAGS

Except for <time> and <list>, any tag has an optional closing counterpart, indicated by a slash. When no closing counterpart is used, the scope of the tag extends up to the next comma or full stop. However, if there is another opening tag with the same keyword, the scope ends before this second tag. The scope of <list> is not ended by commas, but only with full stops. Tags either have no parameter, or the parameter is obligatory. The parameter value is specified after an equal sign following the tag name. In some cases, an optional (intensity) degree, — a number ranging from 1 to 100 — can be added to the parameter value, separated by a colon. The list of possible values for the parameter is specified in Table 3.1 for each tag.

Funded by the Horizon 2020 Framework Programme of the European Union

Tag	Function and Parameter Values
<affect>	affect: Parameter specifies kind of affect (smile, sad,...), allowing degree value.
<indirect>	direct speech: Parameter specifies number of interlocutors (1, 2, 3, 4,...), no degree value.
<time>	timestamps: Parameter specifies time in the format hh:mm:ss.msmsms, no degree value.
<list>	list buoys: Parameter specifies number of items (1, 2, 3,...10), no degree value.
<item>	list items: No parameter, no degree value.
<cond>	conditional clause: No parameter, no degree value.
<rel>	relative clause: No parameter, no degree value.
<ground>	ground-figure: No parameter, no degree value.
<figure>	ground-figure: No parameter, no degree value.
<hold>	hold: No parameter, no degree value.
<topic>	topic marker: No parameter, no degree value.
<neg>	negation: No parameter, allowing degree value.
<nod>	head nod: No parameter, allowing degree value.
<WH-q>	WH-question: No parameter, allowing degree value.
<YN-q>	yes/no-question: No parameter, allowing degree value.
<R-q>	rhetoric-question: No parameter, allowing degree value.
<emph>	emphasize: No parameter, allowing degree value.

Table 3.1: *Syntax Tags in the EASIER Notation.*

Funded by the Horizon 2020
Framework Programme of the European Union

REFERENCES

Bigeard, Sam, Marc Schulder, Maria Kopf, Thomas Hanke, Kiriaki Vasilaki, Anna Vacalopoulou, Theodor Goulas, Athanasia-Lida Dimou, Stavroula-Evita Fotinea, and Eleni Efthimiou (2022). *Initial Interlingual Index for DGS and GSL*. Project deliverable D6.3. Version 1.0. EASIER Consortium. DOI: 10.25592/uhhfdm.10171.

Bond, Francis and Kyonghee Paik (2012). "A Survey of WordNets and their Licenses". In: *Proceedings of the 6th Global WordNet Conference*. Matsue, Japan, p. 8. ISBN: 978-80-263-0244-5. URL: https://bond-lab.github.io/pdf/2012-gwc-wn-license.pdf (visited on 12/09/2021).

Hanke, Thomas, Marc Schulder, Reiner Konrad, and Elena Jahn (2020). "Extending the Public DGS Corpus in Size and Depth". In: *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*. Sign-lang@LREC 2020. Marseille, France: European Language Resources Association (ELRA), pp. 75–82. ISBN: 979-10-95546-54-2. URL: https://www.sign-lang.uni-hamburg.de/lrec/pub/20016.html (visited on 06/18/2022).

Konrad, Reiner, Thomas Hanke, Gabriele Langer, Susanne König, Lutz König, Rie Nishio, and Anja Regen (2022). *Public DGS Corpus: Annotation Conventions*. Project Note AP03-2018-01. Version 4.1. Hamburg, Germany: DGS-Korpus project, IDGS, Universität Hamburg. DOI: 10.25592/uhhfdm.10251.

Kopf, Maria, Marc Schulder, Thomas Hanke, and Sam Bigeard (2022). *Specification for the Harmonization of Sign Language Annotations*. Project deliverable D6.1. Version 1.0. EASIER Consortium. DOI: 10.25592/uhhfdm.9842.

Schembri, Adam, Jordan Fenlon, Ramas Rentelis, Kearsy Cormier, Julian Bleicken, Ilona Hofmann, Olga Jeziorski, Lutz König, Susanne König, Rie Nishio, Anja Regen, Uta Salden, Sven Wagner, Satu Worseck, Oliver Böse, Elena Jahn, and Marc Schulder (2017). *British Sign Language Corpus Project: A corpus of digital video data and annotations of British Sign Language 2008-2017 (Third Edition)*. Language Resource. London: University College London. URL: http://www.bslcorpusproject.org/ (visited on 11/26/2021).

Schembri, Adam, Jordan Fenlon, Ramas Rentelis, Sally Reynolds, and Kearsy Cormier (2013). "Building the British Sign Language Corpus". In: *Language Documentation and Conservation* 7, pp. 136–154. ISSN: 1934-5275. URL: http://hdl.handle.net/10125/4592.

Funded by the Horizon 2020 Framework Programme of the European Union