The Open Lexicon Interchange Format (OLIF) Comes of Age

Christian Lieske

SAP AG
Walldorf, Germany
christian.lieske@sap.com

Susan McCormick

SAP AG Consultant for OLIF2 Consortium
Baltimore, MD, USA
smccormick@home.com

Gregor Thurmair

Sail Labs
Munich, Germany
gregor.thurmair@sail-labs.de

Abstract

This paper summarizes the current status of version 2 of the Open Lexicon Interchange Format (OLIF). As a natural extension of the OLIF prototype (OLIF version 1), version 2 has been modified with respect to content and formalization (e.g., it is now XML-compliant). These enhancements now make it possible to use OLIF in a variety of Natural Language Processing applications and general language technology environments (e.g., terminology management systems). At the time of writing, several industrial partners of the OLIF Consortium had already started work on implementing OLIF support. Details on OLIF can be found on www.olif.net.

Keywords

lexicon, lexical exchange, exchange format, machine translation, XML-based

Background Information

Origins of OLIF

OLIF has its origin in the *Open Translation Environment for Localization (OTELO)* project, which worked on a multi-vendor machine translation environment and was funded by the European Commission in the 4th Framework program. The task of sharing lexical resources was a key element within this environment. Sharing capability provides investment protection, since lexical data is no longer tied to a particular system. At the same time, the overall environment develops greater consistency and potentially broader user acceptance. The version of OLIF that was developed in the context of OTELO (OLIF v.1) was a lean and flat format for lexicon exchange (Ritzke, 1999; Thurmair, 1998).

OLIF Consortium

Since the first version of OLIF attracted interest from many parties, the OLIF Consortium was founded with the aim of providing an enhanced version of OLIF that could serve as an industry standard. The consortium consists of:

- Major machine translation vendors and general language technology vendors, as well as research institutes: Systran, Logos, Sail Labs, IBM/Lotus, LinguaTec, PaTrans, Trados, Xerox, German Research Center for Artificial Intelligence, IAI, and others.
- Major users of language technology: SAP, European Commission Translation Service, Lotus, L10nbridge, Microsoft and others.

The consortium is open to all interested parties; its activities are supported in part by the European Commission in the context of the TQPro project (see www.tqpro.de).

OLIF and Other Standardization Efforts

The OLIF endeavor is not the only actor in the field of lexicon exchange formats. Concertation has thus been an important OLIF goal from the start. The two most important lexicon exchange activities with interfaces to OLIF are:

1. The project for *Standards-based Access to Lexicographical & Terminological Multilingual Resources (SALT)*: Among other things, SALT aims to create a lexicon exchange format. Several concertation meetings have defined a division of work between OLIF and SALT such that SALT focuses on the terminological side of the format (in the tradition of the Machine-Readable Terminology Interchange Format (MARTIF), while OLIF focuses on the lexical side. OLIF and SALT's *XML-based formats for Lexicons and*

Terminologies (XLT) define a common set of data categories to enable integration between OLIF and XLT (concrete integration will be discussed once OLIF and XLT have finished their final reviews).

2. The projects *Preparatory Action for Linguistic Resources Organization for Language Engineering (PAROLE), Expert Advisory Group on Language Engineering Standards (EAGLES)*, and *International Standards for Language Engineering (ISLE)* are also engaged in work on lexicon exchange. These projects are more research-oriented, focusing on elaborated lexical descriptions e.g. in the field of semantics, while OLIF is more pragmatic and tries to accommodate existing lexical resources. Care has been taken to make OLIF conformant with the PAROLE proposals.

Thus, OLIF is positioned on the side of lexical exchange rather than terminology, and leans more toward the pragmatic than more theoretical or research-based projects.

Linguistic Information in OLIF

The basic idea of OLIF is to facilitate the exchange of primarily the pivotal information in lexical entries. This information should be easily compilable into the information that is needed by other formalisms (and systems). Since many formats for lexical entries (including proprietary MT system formats) follow a less lean approach and encode even non-pivotal, usually grammatical, information, OLIF also provides the option of a deeper lexical representation. Included in the OLIF format, for instance, is general coverage of inflection patterns, verb argument structure, semantic types, and selectional restrictions. This sort of information is normally coded idiosyncratically for a given system and is often unusable in a different environment (e.g., a different MT system). The OLIF format offers the user a mechanism for encoding the information in a general way that allows portability.

Information in Lexical Entries

The basic unit in OLIF is the lexical entry. Lexical entries represent independent semantic units, e.g., $bank_{river}$ and $bank_{economy}$ are two different entries. With this approach it is possible to use OLIF to support concept-based approaches such as EuroWordNet. An entry itself is structured as a container for monolingual information, with optional information for cross-references and transfers (provided in the form of links).

Each entry is uniquely defined by a set of key data categories: *canonical form, part-of-speech, language code, subject area*, and, in the case of homonyms, a *semantic reading*. In addition to these obligatory key data categories, several groups of optional data categories can be used (see Figure 1 and the detailed discussion of each group below):

- Detailed monolingual information: supplementary (non-key) monolingual information (e.g. *grammatical gender*) that is grouped further based on administrative or linguistic function (e.g. with morphological, syntactic, semantic or administrative information)
- Cross-reference information: indicating related entries in the language of the entry itself (e.g. *abbreviation, synonym*)
- Transfer information: indicating entries in languages different from the language of the entry itself, that may serve as translations if certain conditions hold

Since, on the one hand, the obligatory information is very sparse (only values for the key data categories are needed), and, on the other hand, interrelated monolingual, and even multilingual information can be represented, OLIF can be easily used in a variety of application contexts (e.g. to replace lists of source-target terms in Excel sheets).

OLIF entries comprise the body of an OLIF file, which, in addition to the lexical entries themselves, contains a header (specifying, among other things, metainformation such as the OLIF version) and an optional declaration of shared resources (e.g., bibliographical information for sources), as illustrated in Figure 1:

```
OLIF-file
        header
                fileDescription
                publicationStatement
                dataCategoryRegistry ?
                contentInfo
                workflowInfo ?
                (replacements | note | prop) *
        body
                entry +
                        mono
                                keyDescription
                                        canForm
                                        language
                                        ptOfSpeech
                                        subjField
                                        semReading ?
                                monoDescr ?
                                        monoAdmin ?
                                        monoMorph ?
                                        monoSyn ?
                                        monoSem ?
                                generalDescr ?
                        crossRefer *
                                keyDescr
                                (linktype
                                generalDC )+
                        transfer *
                                keyDescr
                                transferRestr
                                        contextExpr
                                        testExpr
                                        trAct*
        sharedResources *
```

Figure 1: The OLIF file structure and data categories

Monolingual Information

The monolingual information group in an OLIF entry comprises all data categories that are monolingual in nature. The most important data categories are:

- The key data categories described above

- Administrative data categories, e.g., *administrative status*
- Morphological data categories, e.g., *morphological structure, inflection, head, case, number, person,* and *tense*
- Syntactic data categories, e.g., *syntactic (sub) type, syntactic position*, and *argument structure*
- Semantic data categories, e.g., *definition, semantic type*, and *natural gender*
- General data categories (which can also be found in the cross-reference and transfer groups), e.g., *example*, and *note*

Many of these data categories follow proposals from PAROLE. However, OLIF is rather liberal, and allows for certain redundancies. This permits easier implementation, since implementers can readily use their proprietary data categories.

Not all information in the group of monolingual information is completely explicit. Inflectional patterns, for example, can be given by means of examples (*inflects_like*). This approach provides for an easy-to-fill template that requires a certain amount of intelligence in the compilers to and from the proprietary formalisms/systems.

Cross-Reference Information

The cross-reference data group contains information about related entries in the same language as the entry itself (e.g. abbreviations). The main data categories are:

- The key data categories described above[1]
- Relationship type (like *hyponym, hyperonym, meronym* etc.) according to recommendations from EuroWordNet

Cross-reference information is relevant for applications that support content-related functionality (e.g. query expansion in information retrieval applications).

Transfer Information

The transfer data group contains information about entries in languages that are different from the entry language that may serve as a translation if certain conditions hold. OLIF supports directed multilingual transfers (bilingual transfer being a special case), i.e. an entry can have transfers into several target languages. Transfers are not a priori reversible, however. The most important transfer data categories are:

- The key data categories (see above) of the target entry[2]
- The type of equivalence (e.g. *full*)
- A data group for transfer restrictions that define the conditions under which a transfer is valid
- A data group that describes which structural changes hold for a given transfer. This provides a way of formalizing argument mapping (*I like him -> er gefällt mir*), incorporation (*abblenden -> dim the headlights*), head switching, etc.

As mentioned, a single OLIF entry may have many transfers, also into the same language. Thus, OLIF supports 1:1 as well as 1:n transfers for several languages.

Formalization of OLIF

OLIF is currently formalized as a DTD. From the beginning, the vision was to use two representation formalisms for OLIF: that of DTDs and that of XML schemata. Currently, the DTD is the primary (development) representation for the following reasons:

1. The expressive power of DTDs is smaller than that of schemata (for example, wrt. to ordering constraints within content models). This implies that a formalization that uses all of the features of XML schemata (e.g. data typing for element contents) cannot easily be mapped onto a DTD. Going from a DTD to a schema, however, is straightforward.
2. Formalization as a DTD is generally considered to be quicker than formalization as a schema.

Design Decisions

The design of OLIF version 2 was based on the following principles:

- The formalization should be close to the description of OLIF that is provided in the linguistic proposal described above.
- It should be easy to write programs which process OLIF data. Therefore, some technologies (for example, XLink) for which wide tool support does not yet exist, are not used for the formalization.
- The OLIF DTD, as well as OLIF files, should be legible and reasonably clear. Therefore, terseness is not of great importance.
- The design should show quick progress and follow good practice (for example, commenting). In case these two goals conflict with each other, preference is given to quick progress.
- Maintenance and customization of the DTD should be easy. Ease of maintenance is especially important while the formalization is still under review.
- Lexical data should be represented in a natural way. Thus, concatenation of words by means of underscores etc. (like *inside_out*) is banned.
- The linguistic description of OLIF says that elements within groups may appear in any order. Since there is

[1,2] OLIF provides 2 mechanisms for linking entries: via unique identifiers and via key data categories. Linking via unique identifiers has the disadvantage that cross-reference or transfer information is potentially unusable if the entry to which the identifier points is not part of the OLIF file.

no elegant way of modeling this with a DTD, the free-order had to be replaced by fixed ordering.

- The formalization of alternative content for optional elements is (a|b)+. This overgenerates but is a straightforward way of modeling. Furthermore, this style of modeling has the advantage that no special provisions are necessary to realize the required multiple occurrences of e.g. the data category *project*.
- Clearly, the metadata information in the OLIF header should be represented in terms of the *Resource Description Format (RDF*). Due to a heavy workload, however, RDF has not yet been employed.

Overall Structure and Principles

DTD Modularization
OLIF data represents collections of terminological and/or lexical data. In harmony with the *Terminological Markup Framework (TMF)*, this type of data collection is viewed as being comprised of three building blocks: general information (e.g. title of the collection), a list of terminological entries, and complementary information (e.g. shared resources like bibliographical information). The OLIF DTD reflects this partition, since the top-level file (*olif.dtd*) directly references three DTD modules which correspond to these building blocks: *oHeader.mod, oBody.mod*, and *oShareR.mod*.

Uniform Representation of Data Categories
For certain data categories (e.g. *grammatical gender*), OLIF foresees a fixed set of values. Although these data categories lend themselves to being represented as attributes (if this representation is used, then XML parsers can check values automatically), we have chosen to represent these data categories as elements. The reasons for this decision are as follows:

1. The values of some data categories (e.g. particles for verbs) are multiwords (e.g., *inside out*). However, predefined attribute values that are multiwords cannot be declared in DTDs.
2. Coding every data category as an element (rather than some as attributes and some as elements) provides for a structure that is easier to understand.

Two-level Content Models
In principle, it is possible to declare the value of a data category for part-of-speech as follows:

```
<!ELEMENT ptOfSpeech (#PCDATA) >
```

This, however, does not accurately reflect that OLIF foresees a list of fixed values (that might even be customizable by the user) as the content of the data category. A representation that captures this fact better makes use of parameter entities as follows:

```
<!ENTITY % ptOfSpeech.olif.fix.user.ext
"PtOfSpeech CDATA #IMPLIED">

<!ELEMENT ptOfSpeech
(%ptOfSpeech.olif.fix.user.ext;)>
```

This two-level model is the representation style that has been chosen. The section on coding comments details which types of parameter entities have been defined (the different types are reflected in the naming conventions).

The parameter entities for values that are referenced in each of the three main DTD modules have been placed into their individual DTD module files. For example, the parameter entities referenced in *oBody.mod* are stored in *oBodyV.mod*.

XML Representation for Lists of Values
The *everything is represented as an element* approach mentioned above, does not necessarily mean that implementation of checks for validity poses a difficult problem. In principle, nothing more than easy-to-process lists of values for the data categories are needed. If these lists exist, it is fairly easy to code a program that compares the actual value of an element with the values in the corresponding list (coding may for example make use of an XSL style sheet). Therefore, all fixed or proposed values of OLIF data categories have been made available as XML files.

User Extensions
For certain data categories (e.g. *part-of-speech*), users should be able to supply their own values or domains (sometimes as an alternative to a list of recommended or required values). For this, the DTD adopts an approach which is comparable to that of, for example, DocBook.

The data category is defined with the help of a parameter entity whose name reflects that the data category is user-extensible:

```
<!ELEMENT ptOfSpeech
(%ptOfSpeech.olif.fix.user.ext;)>
```

The parameter entity defines a content model that refers to another parameter entity:

```
<!ENTITY % ptOfSpeech.user.ext "">

<!ENTITY % ptOfSpeech.olif.fix.user.ext
"#PCDATA %ptOfSpeech.user.ext;">
```

That other entity ultimately has to be modified by the user, as in the following example:

```
<!ENTITY % ptOfSpeech.user.ext "|user">

<!ELEMENT user (#PCDATA)>
```

In case this mechanism is used, a reference to the user's list of values must be given in the corresponding data category specification in the OLIF header. For example:

```
<ptOfSpeechDCS>
www.user.net/ptOfSpeechInfo.htm
</ptOfSpeechDCS>
```

Coding Conventions
In order to enhance the readability and maintainability of the DTD, coding conventions such as the following have been used:

1. For data categories whose content model is PCDATA but for which OLIF foresees recommended or fixed values, the special suffixes have been used.
2. Elements and attributes have been described by means of comments that have been put into XML-format. For each element or attribute, its type (element vs. attribute), its name, and its definition are given.

The Header
The OLIF header aims at giving value to lexical and terminological data by looking at both practical and theoretical considerations. Many data/information categories that have been proven useful for other exchange efforts have been included. By looking at the header, questions like the following can be answered:

1. Is the file relevant at all (language(s), project,...)?
2. Am I allowed to use it (copyright, distribution,...)?
3. Where can I turn to for more information (contact person, additional resources,...)?
4. Who created the data (creation tool, user,...) when and how?
5. Can I handle it (encoding, size,...)?

The Body
The representation of the OLIF body closely follows the linguistic proposal of OLIF as described above. Among the few minor points of divergence is the grouping of data categories according to type (e.g. *keyDC* for key data categories).

OLIF Software Environment

OLIF at its current stage is already a blueprint for implementation. At the time of writing, several industrial partners of the OLIF Consortium were already beginning to implement OLIF support. Several applications are under discussion (see Figure 2):

- OLIF converters which convert lexical entries in proprietary format from and into OLIF. Such converters (of interest, for example, to MT vendors) are confronted with two main challenges:
  - Some system lexicons are based on a lexical model in which different semantic readings are collapsed into the same lexicon entry. Converters must be able to identify the readings from the entries. This is a non-trivial task.
  - All systems use proprietary representations for, for instance, inflection information. Writing OLIF converters implies not just a simple mapping of data categories, but usually a mapping of a cluster of data categories (the OLIF representation) into another cluster of data categories (the proprietary representation).
- An OLIF editor which permits reading and editing OLIF entries. The editor is one of the deliverables in the TQPro project. One of the partners (Lotus) will develop an OLIF Application Programming Interface (API) that will allow for reading/writing, editing etc. OLIF entries from and to a database

Other types of applications are being discussed as well. Examples are syntax checkers, and entry verifiers (to test, for example, canonical forms). The OLIF Consortium is convinced that software support is a sine qua non for the success of every interchange proposal.

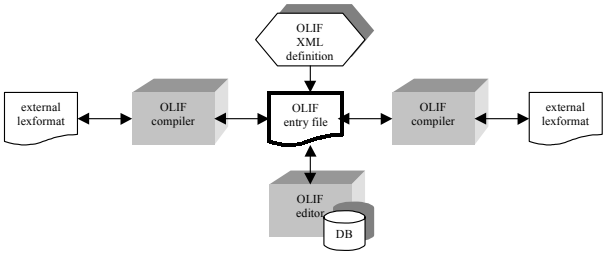

Fig. 2: OLIF Applications

Current Status of OLIF

The current status (April 2001) of OLIF is as follows:

- The linguistic information, and the formalization (DTD) are under review.
- The proposal is being tested by the members of the OLIF Consortium (each one is trying to define how a small set of lexical entries could be represented in OLIF.
- Converters from and into OLIF have been written (by two MT system providers) or are underway (by two more). Tests for exchange should be possible before the end of the year.

After review, and successful tests, further steps are envisioned, among them, the set-up of an infrastructure for the maintenance of the OLIF industry standard, concertation with standardization bodies, and certification.

References

Budin, G. and Melby, A. (2000). *SALT Project – XML Representation of Lexicons and Terminology (XLT).* Proc. Second International Conference on Language Resources and Evaluation, v.II. (pp. 837-844). Athens.

Calzolari, N., Mc Naught, J., Zampolli, A. (1996). *EAGLES Final Report: EAGLES Editors' Introduction*. EAG-EB-EI, Pisa.

McCormick, S. (2001). *Exchanging Lexical and Terminological Data with OLIF.*. Proc. ASLIB.

Ritzke, J. (1999). *Integration of Tools and Engines into a Common Environment: Terminology Exchange between Different MT Systems and Other Resources.* Proc. Lexeter.

Ruimy, N., Corazzari, O., Gola, E., Spanu, A., Calzolari, N., Zampolli, A. (1998). *The European LE-PAROLE Project: The Italian Syntactic Lexicon*, in Proc. First International Conference on Language resources and

Evaluation. (pp. 241-248). Granada.
Thurmair, G., Ritzke, J., McCormick, S. (1998). *The Open Interchange Format OLIF. Terminology in Advanced Microcomputer Applications* – Proc. TAMA '98.
Walsh, N. and Muellner, L. (1999). DocBook: The Definitive Guide. O'Reilly Books.

Acknowledgments

The authors would like to thank the OLIF Consortium and the SALT project for their contributions and support. Maria Jose Munoz Cabanillas and Matthew Chermside receive special mention for providing the initial formatting of this paper, and for proof-reading respectively.