# Recommendations to the CLARIN Standards Committee

*Jan Odijk*

## Introduction

This document summarizes some recommendations to the CLARIN Standards Committee. The recommendations have been derived from experiences in the CLARIN-NL project, but are proposed by the author alone.[1]

It is not fully clear to me what the scope of the CLARIN Standards Committee's work is. I assume here that this committee is able and authorized to commission the development of software. This might be a wrong assessment. If it is a wrong assessment, all such activities should probably be delegated to the CLARIN BoD. For this reason, I also sent this document to the CLARIN BoD and the CLARIN NCF.

**Recommendation 1** The Standards committee must make standards much more visible on the CLARIN ERIC website. Currently there is not even a menu item for CLARIN standards, so visitors have to actively search for keywords such as 'standards', 'CLARIN standards', etc.

**Clarification** I was looking for CLARIN recommended or required standards, but cannot find anything directly via the menu. The closest is the item "Standards Committee" under governance, but that describes the committee, but not standards, and it has no reference to standards.

A string search for "standards" on the web site yields many results, but the top item refers to a page that excels in vagueness. ('We are working on it' but nowhere is there any reference to results or concrete guidelines)

There is a reference to an overview of CLARIN related standards (CLARIN **related,** which means nothing**),** at the IDS website, which has a fancy (but unusable) graphical interface, and anyway, it says nothing on what the CLARIN supported standards are.

Daan Broeder informed me that something usable for CLARIN on the basis of the existing IDS standards registry was promised. It can perhaps be achieved by a 'flag' or a search option that gives us CLARIN required or recommended practices.

Only as the 7th item in the search results, we find "What standards are recommended by CLARIN", and there at least there is a concrete list, but it is an open list, is incomplete I think, and has some odd characterisations (e.g. text format: XML, fine and true but very unspecific unless you mean 'text' as a medium; representation of primary sources: TEI very odd, if not incorrect)

Only on page 2 we find:

[1] I would like to thank Daan Broeder and Steven Krauwer for useful feedback on an earlier draft of this document.

http://www.clarin.eu/content/standard-recommendations

that I was actually looking for, but that probably is outdated (I will refer to this document as the '*CLARIN-PP Standards Document*').

So I recommend at least improving the menu structure so that people can quickly find which standards CLARIN supports, and perhaps also take measures to get pages on CLARIN recommended standard on top when querying for "standards" or "CLARIN standards", or "CLARIN recommended standards".

**Recommendation 2** Maintain a clear and regularly updated list of CLARIN-recommended and required standards with some clarification of the rationale behind the choice, intended application, etc.

**Clarification** An actualized version of the *CLARIN-PP Standards Document* is desperately needed, and it should be regularly updated. There should also be a procedure for users to suggest additions or modifications of this list to the Standards Committee, and for the Standards Committtee on how such requests should be dealt with.

**Recommendation 3** The Standards committee must coordinate the development of wrappers for a variety of CLARIN-supported formats: inventory what is already being done, prioritize supported formats for which wrappers will be developed, set requirements for wrappers, perhaps also have some such wrappers developed at the ERIC level, maintain a registry of available wrappers (or make these wrappers visible via the VLO), etc.

**Clarification** Many tools that operate on text only allow plain text as input format (and sometimes also an idiosyncratic format that is not a CLARIN-standard). But all tools that take plain text as input should also allow formats for text supported by CLARIN as input, such as TEI, HTML, PDF/A, CSV[2], LMF, ISO/DIS 1951, TMX, EAF, as well as XCES (perhaps also CES), RTF, CHAT, Shoebox/Toolbox, Tipster if CLARIN is serious about support for its standards and about interoperability. Furthermore, in various countries other formats have arisen as *de facto* standards, e.g. in NL the FoLiA format, which CLARIN-NL surely will support and CLARIN ERIC will be requested to support. With respect to this, the standards committee should give an overview of the 'accepted' practices from all the other national projects.

In most cases a user wants the results of some tool then correctly integrated in a copy of the input data and in the same format as the input data. To that end, *wrappers* should be developed, i.e. a piece of software that extracts the relevant text from an input format, sends this text to the tool, and integrates the results of the tool in a copy of the input extended with the right mark-up. Alternatively, if the input format allows stand-off annotation in physically different files, the wrapper might just generate an additional annotation file in the right format and with the correct links to the input data. There may be non-trivial issues (e.g. multiple conflicting annotations; input structuring may be incompatible with what the tool

[2] On CSV, see also below.

requires, etc.) with developing such wrappers, but these should be investigated for specific cases, and solutions should be proposed. Some information loss might be acceptable but should be made explicit.

Surely some such wrappers are already being developed by individual CLARIN members. By inventorying what is going on and defining requirements on the wrappers, this work can be coordinated, duplication of work and incompatibility of results can be avoided, and our users will be better served. A software registry (on github or in the CLARIN EU software source code repository) would be welcome for such things. As an example, for CMDI there is an inventory of useful conversion scripts.

**Recommendation 3** The Standards committee should make a list of formats that occur in the 'real world' and that CLARIN claims to support for their use as input format (and only as such) by providing converters from these formats to a CLARIN supported standard format. The Standards Committee should also coordinate the development of converters for such formats.

**Clarification** Many tools for textual data apply to plain text files only. But humanities researchers come with data from the real world, which includes formats such as Word format (.doc and .docx), RTF, .odf, HTML, ePub, etc. In fact, we should be glad that they come with e.g. Word files, because if they come with plain text files we have to ask them which character encoding has been used! Many do not even know what this is, and only very few know which encoding was used. In a .doc or .docx file this is known. In addition to that, we need also analysis / checking tools analoguous to JHOVE.

A first version of such converters could simply convert these formats into plain text, or some other CLARIN standard format; more advanced versions could try to extract as much structure from the input documents as possible (headings, titles, subtitles, sections and subsections, footnotes, etc. etc.)

**Recommendation 4** The Standards Committee should require that tools in the CLARIN infrastructure do not apply to one file only, but to multiple files, of multiple types (see recommendations 2 and 3), to whole folders, and to a set of files in archive files such as zip, tar.gz,tar.bz2, etc.).. It coordinates the development of dedicated modules for achieving this.

**Clarification** Many tools currently allow only one file to be treated. That is understandable if the tool is part of the ongoing research. However, as soon as the tool is going to be used for other purposes, it should provide decent user services such as allowing its application to multiple files, folders, etc. It must also be possible to provide a URL as the location of the file(s) to be processed.

**Recommendation 5** The Standards Committee must require that tools always allow export of the aggregated data that underlie a visualization (graph, diagram etc.), as well as an export of the visualization itself (e.g. as a picture or animation) in a high resolution format.

**Clarification** Many tools provide output that you can only look at but not process further. That is unacceptable. Pictures are perhaps good for seeing main trends but mask many other

aspects of data: therefore the data underlying such a picture must be available and exportable in some standard format for processing in statistical tools, other visualisers, etc. Similarly, it should be possible to extract each picture generated in a high resolution format (screen shots will not do) for inclusion in (possibly paper) publications such as journal articles.

**Recommendation 6** The Standard Committee should make clear what the status is for the following formats: CSV, maps, cross tables

**Clarification**

- **CSV** The *CLARIN-PP Standards Document* mentions CSV in its list but states nothing about it. In CLARIN-NL we have assumed it is a CLARIN-recommended standard and I recommend it as a CLARIN Standard. In several respects the CSV format is better than XML for the representation of single tables, because of the explicit semantics of its format, few or no representation alternatives, its compactness, and the wide availability of tools operating on this format. It has also some disadvantages but XML has all of these as well, as far as I can see. It can also be considered to define a specific format in XML for the representation of CSV files, though I am not sure it is worth the effort.[3]
- **Standards for maps:** Which format does CLARIN assume as its standard? KML? Something else?
- **Standards for web service exchange:** CLARIN say nothing about JSON, though *de facto* it is frequently used. Shouldn't it be a CLARIN-supported standard?
- **Pivot tables** Some tools use pivot (cross) tables as input. Do we have a standard for that?

**Recommendation 8** Tools must be able to process CMDI metadata as one of its inputs and generate CMDI metadata for the resulting data as one of its outputs.

**Clarification** For all data that have been generated by a tool (whether fully automatically or with human intervention) CMDI metadata (incl. provenance data) must be made available, and the best and easiest way to ensure this is by having the tool generate these CMDI metadata on the basis of the CMDI metadata of the input data, human input, and information automatically generated by the tool. These metadata must contain (a link to) configuration files of the run of the tool (see recommendation 9). This is especially important for tools that require large computational resources (parsing, PoS-tagging, etc.) the results of which will be stored as data for further re-use.

**Recommendation 9** Each tool must be able to generate a configuration file as one of its outputs with a listing of all settable parameters, and each tool must be able to use such a configuration file as one of its inputs to set the relevant parameters. This could be part of the CMDI metadata but can also be a separate file to which a link in the CMDI metadata is provided.

[3] In fact, I am even more in favour of a format that I call XSV-format (eXtended Separated Value format), but that is perhaps too much of a personal hobby: it is like CSV (but with any string as possible value separator), but also allows separation of values by a different separator string *inside* CSV cells (one level deep). This format is very convenient for many applications, and is slightly more powerful than CSV without the need to directly go to multiple tables. It was (and perhaps still is) used in Lotus Notes (though not under the name XSV), and I can offer a tool for manipulating such files called AXE (Analysis of XSV Encodings) that I have been using successfully for more than 10 years.

**Clarification** One of my students tried to replicate similarity measure calculations on Wordnet of (Patwardhan and Pedersen, 2006) and (Pedersen, 2010). He did this in an excellent team: Piek Vossen and his research group. He did it with the help of one the original authors: Ted Pedersen. They used the exact same software and data as in the original paper. Nevertheless, they failed to reproduce the original results! And the reason for this is that 'properties which are not addressed in the literature may influence the output of similarity measures' (Fokkens et al., 2013). Many experiments and Pedersen's unpublished intermediate results were required to determine the original settings of all parameters (e.g. treatment of ties in Spearman ρ), and which aspects of the data had been used and how. Having configuration files as described above is a small step towards avoiding such problems (and is additionally very convenient for any user of the tool).

Fokkens, A., van Erp, M., Postma, M., Pedersen, T., Vossen, P., and Freire, N. (2013). Offspring from reproduction problems: What replication failure teaches us. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1691–1701, Sofia. ACL.

**Recommendation 10** For phonetic transcriptions also an ASCII-encoding must be supported in CLARIN, e.g. X-SAMPA

**Clarification** Currently only IPA Unicode is supported in CLARIN, but almost all tools and data for speech processing actually work with ASCII encodings of IPA. Allowing only IPA Unicode would exclude all these tools from the CLARIN infrastructure.

**Recommendation 11** Certain metadata elements must be made obligatory, in particular title and name of a resource, version of a resource, and language.

**Clarification** Metadata are usually made by individual researchers or research groups, who are often not aware of the wider CLARIN context in which their data and metadata will become available. They are often so focused on their own work that they forget to mention in their metadata properties that are or have become 'too obvious' for them, e.g. which language their resource covers (e.g. because they only work with one language). Having titles and names of resources is essential to easily refer (informally, for humans) to the right resource, and explicit versioning is required for replicability and verifiability of research results. For 'language', not only a language code and/or name must be added, but also the time period that the resource covers, and whether the resource deals with the standard language, with dialects / sociolects of the language, or with both.

**Recommendation 12** Input and output of tools and services should be organized by these tools and services in a user-friendly manner.

**Clarification** The recommendation should be obvious, and almost everybody agrees with it, but in practice this is often not done for research software, which is only used by a single researcher or research group. When such tools are made available for the wider humanities research community, user-friendliness of the tools is essential. It is also important for improving verifiability and replicability of research results.

Concrete means to achieve user-friendliness in the sense intended here are inter alia: it must be clear where output results can be found; there must be a clear distinction between input and output files and a clear connection between multiple corresponding input and output files (e.g. by applying systematic naming conventions); logging files should be stored separately, made visible in interfaces only upon request**,** etc.