

NAT-term – a authoring tool for terminologies

J. João Almeida Alberto Simões Alvaro Iriarte

In this presentation we discuss NAT-term – a language to author terminologies, and a set of tools to validate, transform and produce a variety of outputs. Initially built for didactic intentions, this tool has also been used in other contexts.

Keywords Terminologies, Dictionaries, NLP

Introduction In translation studies, terminology tasks (understand the concepts, definition, creation and use) is simultaneously (1) a crucial step, (2) something that takes some time to master, (3) often taken not so seriously as it should.

According to our experience, after learning the main concepts, students need to be involved in terminology projects. We experiment several approaches. We start with terminology created on Word (but the often good-looking results have very poor structure and use). We moved on to smartcat's glossaries (Web based tool), where it is possible to define fields (standard or user-specific) – but it was difficult to obtain some advantages or results, and the entries insertion process is painful. We tried MultiTerm (some concepts are clear and elegant, but lots of problems with the tool (installation, Java incompatibilities, use outside Windows))

In the end, we decide to build a specific terminology tool – NAT-term – to be used to teach, create, discuss terminologies, dictionaries and glossaries. According to the indented scope and focus, we hide some notation and features.

Design goals NAT-term was build with the following features:

- Concept-based entries
- Textual input (it is possible to create a nat-term terminology with just a text editor)
- A set of tools to translate the terminology to a variety of output formats: (1) PDF dictionary; (2) xdx (XML format for dictionaries) – to be used with goldendict (or similar) tools; (3) (multi-file) HTML site; (4) other formats planned .
- Transformation (nat-term terminology → output formats) available through a command line script (for programmers) and through a very simple web interface (to be used with no installation effort and dependencies)
- Rich micro structure type of attributes: (1) concept attributes; (2) conceptual relations – it is possible to define relation properties (ex: InverseOf) to get inference and linking; (3) attributes for terms; (4) term / concept attribute values – multimedia attributes, text, numbers, etc.
- syntax for macro-structure definition (multi-level domain trees; maximum level: 3)
- advanced features to (optional): (1) create entries based on tables; (2) external terminologies; (3) directives to control output style: **ignore** (hide a field), **rename** (change the output name of a field), **inline** (compact / reduce size).

NAT-term by example [this section, references, examples, images were omitted]

Conclusions (1) Building real projects helps with terminology concepts comprehension.
(2) NAT-term authoring syntax (or subsets of it) helps clarifying ideas (ex: inverseOf completions achieves good results but fails miserably if the concepts are not well organized).
(3) Good-looking output encourages the students.