



# Natural Language User Interface to Topic Maps with Prolog

Thomas Flemming

Email: [thomas.flemming@usit.uio.no](mailto:thomas.flemming@usit.uio.no)  
University of Oslo, Center for Information Technology Services , Norway

## Objectives

Can Natural Language Processing techniques be used to unleash the potential of Topic Maps for end users? Can TopicMaps be more useful with Natural Language user interface?

## Using Prolog as backend for Topic Maps

To develop a prototype the programming language Prolog was found quite useful for doing operations on TopicMaps like querying and updating.

### Benefits of Prolog

Prolog is a mature programming language, and has been in use since 1972. Prolog can be used for storing, querying and updating a Topic Map. It has a useful built in syntax for defining Definite Clause Grammars. Prolog also has many similarities to TML query language.

### Prolog conventions used

Original TopicMap in LTM syntax:

```
[traviata : opera = "La Traviata"]
[verdi : composer = "Giuseppe Verdi"]
has_composed(verdi:composer,
             traviata:opera)
```

Same TopicMap can be coded like this in Prolog:

```
basename(traviata, 'La Traviata').
topic_type(traviata, opera).
basename(verdi, 'Giuseppe Verdi').
topic_type(verdi, person).
has_composed(verdi, person,
             traviata, opera).
```

Then we can add data using prolog asserts:

```
assert(basename(verdi,
               'Verdi, Giuseppe', sortname)).
```

Rules in Prolog can be compared to views and triggers in a relational database. The rule beneath makes it possible to query for associations of the type 'has\_composed' without specifying roles:

```
has_composed(Person, Work) :-
    has_composed(Person, Role, Work, WorkRole).
```

It has been shown that Prolog can be used for constraints by Santosaputri (2003).

## Tools developed

I have written a small Java API to simplify operations like adding a topic, searching for a topic and do queries that returns a list for Topics. The queries and update statements are written in Prolog. The open source Prolog engine JLog was used for Prolog. For searching in basenames, Lucene is used to optimize search.

## Utilizing linguistic information in Topic Maps

### Mapping from TopicMap elements to natural language elements

Instead of defining a lexicon with verbs and nouns and other word classes, a mapping from TopicMap elements to natural language elements like verb phrases and noun phrases was created.

Basenames are mapped to proper noun phrases when the topics has not been used as a topic type, association type, occurrence type or association role. For instance is the topic with the basename "Johan Sebastian Bach" a proper name, a proper noun and a noun phrase. Association types normally consists of Verb Phrases like "has composed".

Linguistic term	Topic Map term	Examples
Nouns	Topics used as types for topics	"person", "fugue"
Proper Nouns	Topics not used as types	"Johan Sebastian Bach", "St. Thomaskirche", "Leipzig"
Verb Phrases	Topics used as types for associations or occurrences	"was born in", "compesd"
Transitive Verbs	Associations. Verbs that take an object.	"belongs to"

### Adjectives in Topic Maps

To be able to understand the natural language question "what cost the cheapest mp3 player", our question answering system needs to understand that "cheapest" is an adjective meaning lowest value of price.

To do this, our Topic Map can be extended with the name "cheapest" as a name with the scope "adjective-smallest" for our occurrence type "price". In LTM syntax:

```
[price = "Price"
 = "cheapest" / adjective-lowest]
```

This can again be used to recognize the pattern "NounPhrase/TopicType adjectiveLowest" to interpret questions "the cheapest mp3 player"

## Information Extraction

### Using a pipeline of recognizers

For expanding a TopicMap automatically with information from a corpus, a set of different stages in a pipeline can be used to extract information and bring it to the next stage..

- 1. Identify Proper Names**  
In Topic Map terms this means identifying basenames for topics not used as a topic type, association type, occurrence types or role types.

For instance "Johan Sebastian Bach" and "Leipzig".

- 2. Identify Noun Phrases**  
For instance "the composer"

- 3. Identify Verb Phrases**  
For instance "was composed"

The Topic Map can have several scoped names for the same association type, all meaning "composed by".

- 4. Identify patterns of interest**  
The pattern "ProperNoun VerbPhrase NounPhrase", can be used to add information about a musical work to the Topic Map that already contains information about Bach and musical works.

For instance the sentence "Johan Sebastian Bach composed the cantata the Coffee Cantata" contains the Subject-Verb-Object pattern.

The technique is also called cascaded finite-state Transducer (Hobbs et. Al 1995).

### Question answering

The pipeline can also be used to recognize questions like "Who composed the Coffee Cantata?". This can be rephrased as the Prolog query:

```
composed(X, _);basename(X, 'The Coffee Cantata')?
```

## Usage

### Increasing search quality

If a search query is recognized as a question, this can be used to give the actual answers to the question in addition to normal search results. A search for "convert 56 inches to centimeters" in Google, gives the answer first and then search results for the search string.

### Information extraction

The bottleneck in Natural Language Processing is to initially teach the system knowledge about the world and a lexicon of words. This could be simplified if we could use Topic Maps.

### Natural Language Topic Map editor

Natural language can be used to let users enter information to a Topic Map. This require a TopicMap with enough knowledge to understand the users input.

## Discussion

The potential for Natural Language Processing techniques in cooperation with TopicMaps seems promising. There are many questions to be answered. Can Natural Language Processing be used in a useful way to give users more possibilities to query, create or update Topic Maps? If users can store their data in more formal ways, like in Topic Maps, what can that information be used for?

### Related work

The question answering system Metalog has been made for RDF (*Massimo 2004*). Metalog uses a restricted subset of English for querying RDF data.

There is much literature about Natural Language Interfaces for Data Base (*NLDB*). Topic Maps with a general standard data model should be more suitable for natural language input than conventional relational databases.

Some work has been done on using natural language processing for Semantic Wiki (Nissim 2006).

## References & Acknowledgements

Thanks to Are Dag Gulbrandsen, Lars Marius Garshol and Cecilie Dèsirée Widsteen for hints and comments:

1. Erica Santosaputri 2003 "Prolog for Topic Map constraints"
2. Hobbs et al 1995. "FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text"
3. Massimo Marchiori 2004 "Towards a People'sWeb: Metalog"
4. Nissim, Malvina & Presutti, Valentina 2006 "Wikipedia meets natural language processing". Wikimania proceedings.