Dolphin – a Knowledge base for Transparent Intensional Logic *

Andrej Gardoň, Aleš Horák
Faculty of Informatics, Masaryk University Brno
Botanická 68a, 602 00 Brno, Czech Republic
E-mail: isac@mail.muni.cz, hales@fi.muni.cz

Abstract

This paper describes the design of newly developed Dolphin system for effective implementation of a knowledge base and question answering based on the transparent intensional logic (TIL). TIL is a high-order logic with a hierarchy of types made for representing all natural language phenomena like intensionality, temporality or belief attitudes.

In the text, we will introduce the database acting as a knowledge base for inference in TIL. The time aspect of the truth value of propositions will be mentioned and basic “thinking” capabilities of the Dolphin system will be exemplified.

1 Introduction

Transparent Intensional Logic (TIL) was introduced by Pavel Tichy [1] with the purpose to overcome paradoxes arising from other modern logical systems (FOPL as well as intensional logics [2]). A short summary of the advantages of TIL over Montague’s dynamic logic can be found in [3].

Dolphin is a part of a complex project employing TIL to develop a computer system with data understanding ability. Such system will be able not only to store natural language (NL) structures but also to mine new facts logically resulting from input and data stored before. One of the final benefits of the whole system will be no need for special human-machine communication language – the user will just ask and Dolphin will answer what can be inferred from the knowledge base to the question.

*This work has been partly supported by Czech Science Foundation under the project 201/05/2781 and by Grant Agency of the Academy of Sciences of CR under the project 1ET400300414.
To meet this challenge a special database is needed to store the knowledge encoded in TIL constructions. Currently, the Dolphin system stores labels of NL objects described in the Czech language. However, the actual chosen language has no influence to the functionality of the system. Czech is used in Dolphin because the transcription of NL sentences to TIL constructions is handled by the NTA system [3] that presently provides only the Czech transcription ability.

2 The Dolphin Knowledge Base

When a new input is to be stored in the database, it has to go through several parts of our system. The Figure 1a) demonstrates the process of storing new data to Dolphin. In the first phase of storing a new fact that “Lemon is yellow.” the sentence is translated with the Normal Translation Algorithm (NTA) to the TIL construction expressed by the whole sentence. Dolphin then parses the construction and stores it in the knowledge base (KB). At the level of propositions, Dolphin interprets logical operators and divides the sentence to several basic facts. The last symbol of the sentence (.,?,!) indicates the mode of the database (storing/answering/ordering). NTS stands for the Normal Translation Synthesizer and its role is to produce a sentence from TIL descriptions provided by the output of Dolphin.

3 The Dolphin Storage

As we mentioned before, language has no role in the functionality of the Dolphin KB. On the other hand it is very important in searching for objects. The Dolphin KB thus stores the data in two layers: language and object.
3.1 The Language Layer

This layer provides the language encoding ability of the Dolphin system allowing the rest of the system to be completely language independent.

The implementation of the language layer (LL) has the form of multilingual dictionary providing ILIs (inter lingual indexes) for the object layer. Each LL record contains an object name expressed by words in different languages.\(^1\)

3.2 The Object Layer

This layer is quite complex so only basic features are mentioned, see [4] for details. Each TIL object is given a type from the type hierarchy built over four basic types – \(\iota\) (individuals as object labels), \(\sigma\) (truth values, \textit{true} and \textit{false}), \(\tau\) (time moments and real numbers) and \(\omega\) (possible worlds, rigorously defined in TIL).

One of the essential KB objects for inferring the object properties are the objects of types including \(\iota\) in it. Thus those objects are stored in separate files containing this information:

- the type of the object
- the connection to the language layer, provides naming of objects using words of natural language
- membership – since TIL is a high-order logic, many objects form classes or relations. To provide fast way of searching for data, it is essential to know in which classes/relations an object has a membership
- members – only objects with composed type to identify simpler objects that form it

Objects of type \(\sigma\) and \(\omega\) are currently just stored in the database without any special handling.

3.3 Handling of Temporality

In non-temporal logic system (e.g. FOPL), a fact that “\textit{There is a bird on the roof.}” has only a truth value of true or false. Everything would be good until the bird flies away. If this happens, the system can only change the truth value of the proposition to false. But by doing this, we loose information that “\textit{There was a bird on the roof in some time.}” In TIL the situation is different. The system stores these facts or bird being on the roof and bird flying away as two propositions being true in different time intervals. Let the bird fly away and let new information “\textit{The bird flew away.}” be provided to Dolphin. Although Dolphin has no sense to notice the bird flew

\(^1\)Thus if we know properties of an object (e.g. \textit{lemon}) in English it is enough to provide Czech translations of the properties instead of describing \textit{lemon} in Czech language. If \textit{sour} stands for \textit{kyselý} in Czech, we do not need to be told that “\textit{Ocet je kyselý (Vinegar is sour).}”
away it has a time sense and in the moment of providing the last information it
knows that the bird was on the roof for some time. Then we put our bird back
on the roof and tell Dolphin about it. What is important is the fact that Dolphin
knows the bird is sitting on the roof for the second time. Moreover, Dolphin is able
to answer such questions as “How many times the bird was on the roof today?”
The question is just asking for how many times the truth value of the proposition
changed from false state to true during a day, see the Figure 1b) for the result.

4 Basics of Dolphin Inference Machine (DIM)

Nowadays, DIM is at the start of its development. Currently, DIM is equipped only
with basic Modus ponens like inference mechanism and is able to answer question
from conversation: “Every boy is a child. Peter is a boy. Is Peter a child?” DIM
is able to handle the quantification operators as special objects of Dolphin, Every
and Some, that influence inheritance of relationships of a class to its members.

At first, Dolphin looks if the object has some relationship with Every/Some
objects. In our conversation the sentence “Every boy is a child” suggests relation-
ship between relation Every and objects Boy, Child. After the second sentence is
typed DIM asks Dolphin for all objects that meet True = λx[EveryBoyx]wt. The
database returns object Child and DIM can provide a new fact “Peter is a child” to
the database.

5 Conclusions

We have briefly presented basic features of the Dolphin system for handling struc-
tured knowledge of the transparent intensional logic with the question answering
capabilities.

References

cz/projects/dolphin/...