

Z is used as a toolkit to specify UTP programs

UTP programs are a set of observational variables that belong to an alphabetised relational calculus that can be used for unification of many programming paradigms.
-> Therefore, it is possible to use Z to represent UTP

--> A natural language to describe the Z notation ease the task to use the Z toolkit.

This language can also be seen as the natural language "specification" of the model, therefore, it shall also be able to describe refinements also in natural language.

* Using Z as a toolkit to specify "languages operational semantics", based on the UTP alphabetised relational calculus, cover the definition of the operational semantics of the "new language".

---> The proposed natural language to represent the Z notation should be able (or have some extension) to also specify the new language syntax. The new syntax constructions should be bound to the language semantics expressed as UTP, through Z, but also in natural language.

Ex.: The Circus language should be processed by the system once it is fully described using natural language based on the Z-NL and the Syntax-NL. Any new language should be understood based on natural language explanations (operational semantics bound with new syntax).

Phases

1) Specify natural language to cover the Z notation

- Only the core definitions should be expressed. Any new definition should be defined using the z-core; its syntax should also be dynamically defined. Therefore, It seems this initial natural language should be abstract enough to represent Z constructions and name new ones (new verbs, nouns, modifiers, etc).

One possible Z-NL should be based on grammatical constructions (verbs, nouns, modifiers, etc) and a core-NL that enable definitions of new grammatical syntax.

= Because the Z notation is a formal language the proposed natural language can not have ambiguities. Therefore, terms can have only one meaning in a given scope.

= The language has the sense of time since in UTP we can have 'decorated variables' representing the value of in a certain time.

** Main ideia: any language can be constructed based on an atomic set of rules and the meaning of its words can be formally defined using an 'alphabetised relational calculus', which can even represent operational semantics.

-> If some sentence is too abstract and cannot be interpreted, it means it does not make sense in a relational calculus. We are treating here of a subset of natural languages that actually "makes sense" independent of anything.

- This formalization of reasoning about natural language (that can actually be understood without ambiguity) may also be useful for accomplishing translation between natural languages.

-> How to define 'context' so the same word can have different meanings?

** There shall exist a tool that validates the natural language text according to the existing definitions and syntax, if a sentence cannot be understood it shall ask the user the meaning of its terms and a one or more new definitions shall clarify the understanding of the new terms.

Solutions 1

It seems that in order to define a language that can describe its own constructions it is necessary to:

- * describe a Z-spec in natural language
- * find 'abstract elements' that define NL grammar
- * define a Natural Language Meta-meta-model
- * define

Problem: if the definition of the language "grammatical rules" is done using natural language itself, how is this previous NL defined? Is there any core-NL? If is there a core-NL, which can be used to define new grammatical constructions, what is its meta-model? Is it possible to define this elements at least for the Z-NL context?