Mizar and the Machine Translation of Mathematics Documents

Grzegorz Bancerek
Institute of Mathematics
Warsaw University in Białystok

Patricia Carlson
Teachers' College of English
Białystok

Abstract: The Mizar language and the machine translation of mathematics documents have been in the process of development for the past sixteen years at the Białystok branch of Warsaw University. The original objective of the project was, and still is, concerned with "doing" mathematics. The second is concerned with the machine translation of mathematical papers which have been written in Mizar. And the last is involved with the formatting of those papers for publication in the journal, *Formalized Mathematics* [6], which is put out in Louvain, Belgium. This translation project includes the following software: the Mizar language checking devices and processors, a module of Pascal programs and procedures that perform the machine translation, and TEX and LATEX that do the formatting.

The situation described in this report is the situation that we hope to reach in the future rather than the actual state of affairs which is described in [1, 2] and also [4, 5].

Contents

1 Introduction

The Mizar language is a formalized language for mathematics. It has a unique grammar in which every mathematical fact can be expressed (the diagrams of grammar of Mizar abstracts[1] can be found in the Appendix B); however, expression is possible only in a limited number of ways. In this language, there is a collection of documents (Mizar articles), which form the Main Mizar Library (MML), and which successively develop mathematical knowledge. The object of the translation module is to render these documents in an English-like output that would be understandable to mathematicians.

There are several specifications for these documents. First, all concepts and facts used are introduced in the article or linked from another that was written previously. Hence, the environment portion of the document refers to other documents accepted previously in MML, and linked concepts and facts are not repeated. The articles in the MML are ASCII files, and they do not have the capacity to express all mathematical symbols in addition to others which individual authors might wish to add. More about Mizar you can find in [3] and [7].

The specifications for output include translation into readable English with the appropriate use of all mathematical symbols. These translated documents are to include introductory portions which cite the references to previous documents. In order to ease the dryness and content-laden mathematical portions, reasonable filler is to be used. The result is to be a reasonably well-written mathematical document.

2 Translation

The approach to translation is rationalistic rather than empirical. The module, MizTEX, performs the translation of the abstracts and utilizes the object oriented methods in Pascal. This translation portion of the project is based on the following five objectives: to parse, to analyze, to translate, to enhance, and to TEX. (The processing diagrams can be found in the Appendix A.)

2.1 Parsing

The first objective is to parse the Mizar article which is to be translated. An abstract tree, with a unique grammar and specific to the article, is constructed.

2.2 Analyzing

The second objective is to analyze the tree that has been created specific to the Mizar article that is to be translated. This involves a sequence of passes through the tree from root to

[1] The Mizar abstracts are results of the mechanized process which eliminates from the Mizar articles proofs and auxiliary steps and leaves global items: reservations, theorems, schemes and definitions.

leaves. The approach is deductive, proceeding from the most general to the most particular in the mathematical formulae. The analysis is performed in order to translate all articles that are referenced in the particular one being translated. There are four processing stages in analysis: statistics, concepts used, references to theorems and schemes, and variables respectively.

2.2.1 Statistics

The first step in analyzing the tree is to find some statistical information which will be used in succeeding steps. This information consists of the following:

- **Global items.** How many - definitions, theorems, and schemes - occur in the article and in which segment they appear. This data on placement must also include their size and how they are grouped. Additionally, there is a need to know which and how many definitional items there are in every definition, and how many and which kind of parameter together with how many premises there are in every scheme.
- **Concepts.** Which concepts from which articles and how many times they are used and in which segments they appear. There is the need to know how deep they occur in the tree structure.
- **Theorems & schemes.** Which are referred to and in which article they appear, how many times these references occur in every section and in every proof.
- **Variables.** Which and how many times they occur and in which segments.

2.2.2 Concepts Used

In addition to the above information, the next step uses the results gained previously from the use of MML and the Mizar database. This step includes the following points:

1. Completion of concepts wherein one concept implies or is dependent on another,
2. Selection of concepts to be cited in a global preliminary section and those to be cited in local preliminaries that appear in each section,
3. The decision on whether a definition of a concept must be fully cited or must have merely a reference without an extensive definition.

2.2.3 References to Theorems & Schemes

This process uses the information gained from MML. Certain decisions must be made:

1. Which theorems and which schemes included in other articles and referenced in the present article have to be cited.
2. In which place (preliminary, in which order, and between or after which concepts) theorems or schemes should be cited.

2.2.4 Variables

In this process, we have to choose which variable font styles are to be used and in which place in the translated text they will be introduced.

2.3 Translating

To translate the tree that has been constructed, larger structures are selected through the use of connectives in the mathematical logic portion of the tree. This logically follows a complete analysis of the atomic formulae represented in the leaves. There is attention paid to deep structures, and, secondarily, there is a focus on the selection of form with stylistic variants.

The *translation process* can be represented by the following diagram:

$$T \xrightarrow{P} E$$

where T represents the abstract tree completed by "analyze", P represents all translation variants for patterns of tree constructions in every level of grammar, and E represents an English-like output with mathematical symbolism.

Translation variants can be provided by the author-independent part and the author-dependent part. The first is a set of translation variants for such constructions which are common for every mathematical article - the grammatical level of definitions, second order propositions, and propositional and quantifier calculus. The author-dependent part is a set of verbal and symbolical explanations provided by the author for every concept that he/she introduced.

In the translation process, a decision must be made on that which is to be translated verbally (and how much) and that which is symbolic (and in which notation). To make this decision, the real size of every term in particular translation variants, priorities of verbal and symbolical variants, statistical information (constructors - place, frequency, and depth), and possibilities of using variable abbreviations are used. Moreover, there are external parameters, maximal and minimal limits of verbal translation, which the editor can control.

2.4 Enhancing

Having translated the tree, the next objective is to enhance the output. This segment creates a TeXed file. The translation is transposed into TeX statements in order to provide formatting instructions for printing the article in *Formalized Mathematics.*

2.5 TeXing

This last segment in the process is the printing of the TeX file. This is the final version of the translated Mizar abstract and is fully formatted for publication.

3 Appearance of Output

In this section we want to explain how the output (mathematical article) looks. The article includes the title, the name(s) of the author(s), the affiliation, a summary, and the main text. The summary is written by the author(s), but the main text is a result of translation of the Mizar article. The Mizar article has at standard format which includes the environment and text proper which is divided into sections, each one having its own title. The machine translation of the document must preserve the sectional divisions. (Note the appearance diagram in the Appendix A.)

4 Additional Information from MML

Dependency Relation. The first step is to find which concepts depend on other concepts for their definitions. This will be called dependency relation. This relation is accomplished by starting from the empty set. Next, for every concept C_1 we add pairs (C_1, C_2) where is a concept appearing in the sentence which defines C_1. In this way, we have a net of dependence which can be used to complete sets of concepts used in the article.

Statistics. It is known how many times each concept is used. Those concepts which have a high frequency form basic knowledge; the editor can manipulate and control the level and limit of the basic knowledge.

Clustering. It is desirable to group concepts according to subject, but it is first necessary to determine the subjects. This is dependent upon the dependency relation and the level of basic knowledge. If a particular subject is large, it is necessary to subdivide into smaller areas. This is done by deciding on how many concepts must be included in the dependency set. If one imagines a net structure of concepts that are inter-related, the strategy is to isolate the middle area; in such a method, those at the bottom of the net are eliminated, as well as the those above. Further clustering is done by analyzing the subjects in the middle area of the net. First we try to discover disjoint areas.

5 Concepts

In order to have an output that is clear and understandable to mathematicians, certain concepts must be explained in the preliminaries. By explanation, we mean that the definition of the concept must be cited or only a reference to the article in which the concept is introduced is used. There is no need to explain all concepts but to choose only those that are necessary. Next, those concepts that are selected must appear in an order and in places that preserve the logical construction of the article.

The processing of concepts works in following way. We start from the set of concepts used in the article. We have a level of basic knowledge for the subject of the article in addition to the dependency relation between concepts and in which article they are introduced. The first stage in processing concepts is to eliminate those concepts which form basic knowledge. For those remaining concepts, we must find the linear order determined by dependence among concepts and their appearance in the article. The ordering of concepts is done in the following way. First, for every concept, we have its first appearance, thus beginning the linear order desired. Then, we move every concept A which is required to define concept B to a position before concept B and we iterate this step until there are no pairs of concepts in reverse order of dependency. Thus, order is determined by dependence and in some measure by the appearance of the article. Then, we try to find segments of this order which are independent and can be moved with preservation of dependency but this movement must fit more logically in terms of appearance. At this point, it is possible to make decisions about which concepts should appear in the global preliminary and which in local preliminaries. Besides this, it is possible to make decisions about citation or reference and which concepts can be eliminated by grouping. Moreover, if the decision about citation is made, we can decide to complete explanations by new concepts.

6 MML References

Every statement in the article has justification, which can refer to facts from other articles. Statements form reasoning, the sense of which has to be translated. To have understandable output, we have to explain the theories which are used there. This means that some theorems, lemmas, and second order sentences from other articles have to be cited. Moreover, it is necessary to say from which articles the facts are used and with which concepts they are concerned.

The processing of references works in the following way. We start with the set of facts used in justifications and with the results of the previous process dealing with concepts. Similar to the previous process (viz concepts), we eliminate facts which concern basic knowledge and try to order facts. The ordering of facts is a result of reference placement in the article and the order of selected concepts. At this point, it is possible to make decisions about which facts can be explained in the global preliminary and which can be done in the local preliminaries of particular sections. Furthermore, we can decide how we can mix the explanations of facts and concepts in order to achieve a logical structure. By using frequencies of justifications, common concepts, and information from which article are facts, we can eliminate some of them by grouping. Those facts are explained only by note of article and concepts used.

7 Translation Process

After the analyzing processes, we get an abstract tree which carries a structure of the article completed by preliminary segments. This tree has to be translated to English-like output which preserves the mathematical sense of the article. Mathematical articles (accepted in MML) are based on common fundamental theory and they, therefore, have common constructions in the level of definitions, propositional and quantified calculus, second other sentences, and so on. The translation in those levels has to be similar in every article, i.e. there are variants of translation independent of the author but taken from natural language (and used by mathematicians). There are general translation variants and, when translation of the general structure has some exceptions, there are more particular variants, and the same for the exceptions to the exceptions. The most exceptions are concerned with the structure of sentences, because when the tree of a sentence is translated, it has to preserve its structure without usage of brackets as grouping appears in formal language, but by usage of specially chosen connectives. Moreover, for every structure we can have more than one variant with information as to how often it can be used. Besides common constructions of mathematical articles, we have constructions concerning concepts introduced which cannot be translated without the author's explanations. It means that the author has to give to editors the translation variants for concepts he introduced in the article. Those variants have to be done for every context foreseen by editors and, if it is possible, in verbal and symbolical forms. Of course, the author can propose for every context more than one variant and then he/she can also qualify priority of every variant. Those priorities are used to choose how often synonymous variants can appear in translated output.

In the translation process, the decision is made regarding the limit between verbal and symbolical translation of the tree. To do it, first the limit is put between the propositional level and the structure of atomic sentences. Next, the limit is moved deeper by usage frequency of constructors for which the decision must be made. If the constructor is used not so often, then the verbal variant of translation is chosen, otherwise, symbolism is preferred. There is also a rule to choose as small a number of different variants as well as possible. But this rule is not for constructors which form basic knowledge. Simultaneously, for terms which have large real size in chosen variants, variable abbreviations are made. These same abbreviations are made for constructors which are smaller in real size but are used in sentences (in proof, in definition, ...) more than n times, the number n being determined by the real size and the option given by the editor.

The most difficult part of translation is to translate proofs. Here, we do not want to have such exact proofs as in translated formal language, but we want to have understandable output which carries the sense of reasoning and does not lose readability in considerations extorted by formalization. The first stage in translation is to eliminate those segments of reasoning which deal with basic knowledge or which have too many nested proofs. Next, we try to reduce those steps of reasoning which use only local statements in justifications. As far as it is possible, we try to find in reasoning the most important lemmas necessary to justify final conclusion(s). After that, we analyze sets of references used to justify every lemma and try to find common denominators for every set. These common elements can be concepts, a subject or a "sub-subject" or an article.

References

[1] Grzegorz Bancerek and Patricia Carlson. Semi-automatic translation for mathematics. In J. Darski and Z. Vetulani, editors, *26 Colloquium of Linguistics - Abstracts*, pages 17–18, 1991.

[2] Grzegorz Bancerek and Patricia Carlson. Semi-automatic translation for mathematics. In J. Darski and Z. Vetulani, editors, *Proceedings from 26 Colloquium of Linguistics*, 1991. to appear.

[3] Ewa Bonarska. *An Introduction to PC Mizar.* Fondation Philippe le Hodey, Brussels, 1990.

[4] Roman Matuszewski. Preface. *Formalized Mathematics*, 1(1):5–6, January 1990.

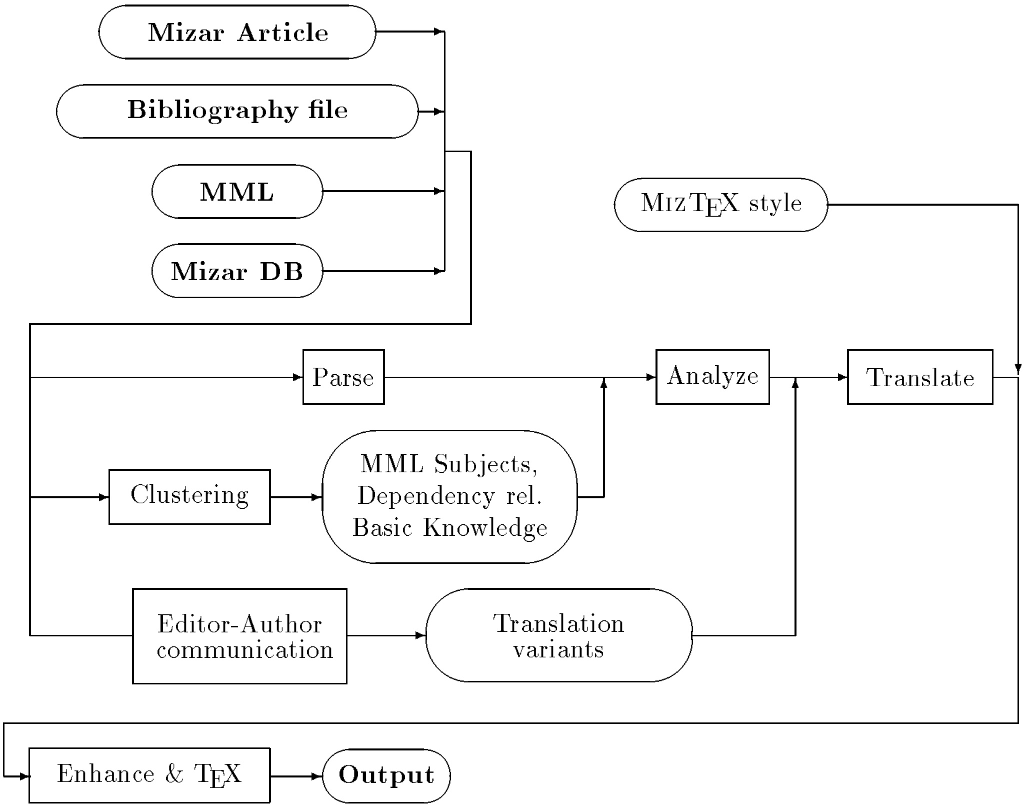
[5] Roman Matuszewski. Preface. *Formalized Mathematics*, 1(4):623–624, September–October 1990.

[6] Roman Matuszewski, editor. *Formalized Mathenathics: a computer assisted approach*, volume 1–3. Université Catholique de Louvain, 1990–1992.

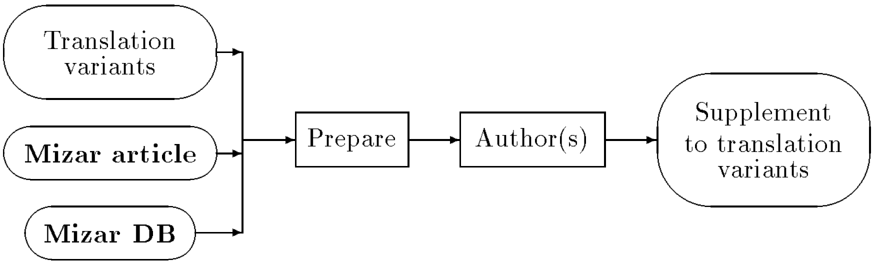
[7] Andrzej Trybulec. Introduction. *Formalized Mathematics*, 1(1):7–8, January 1990.

Appendix A. Translation Processes Charts

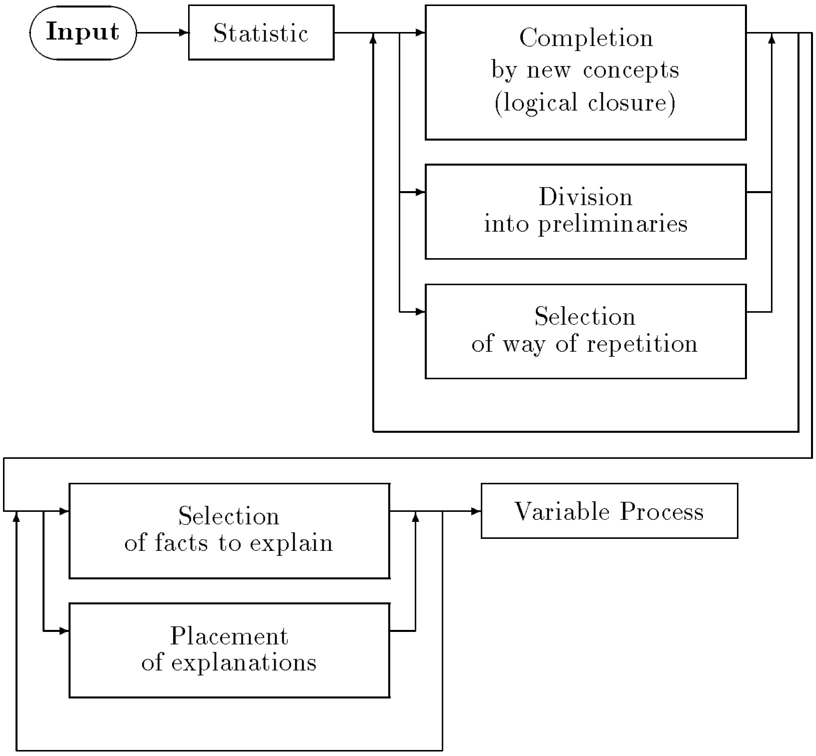
Editor Processes



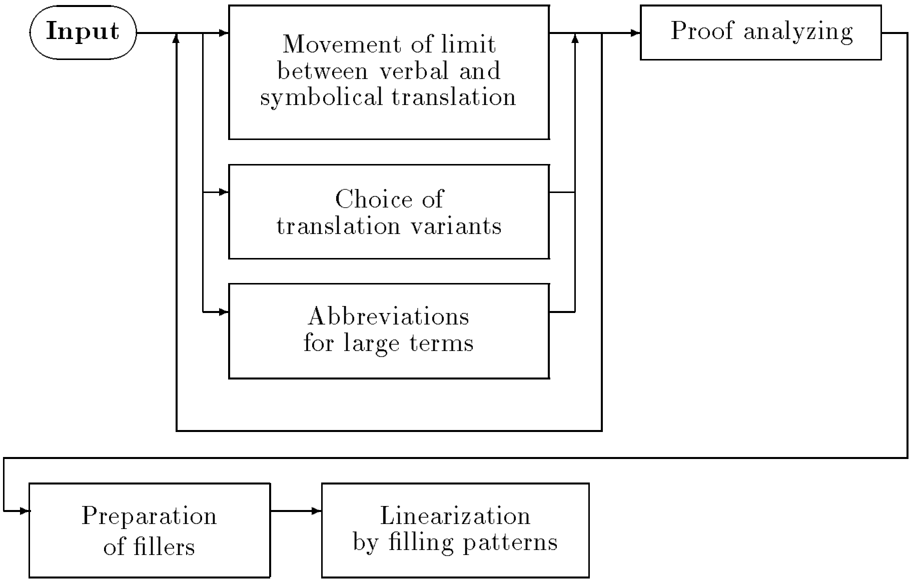
Editor-Author communication



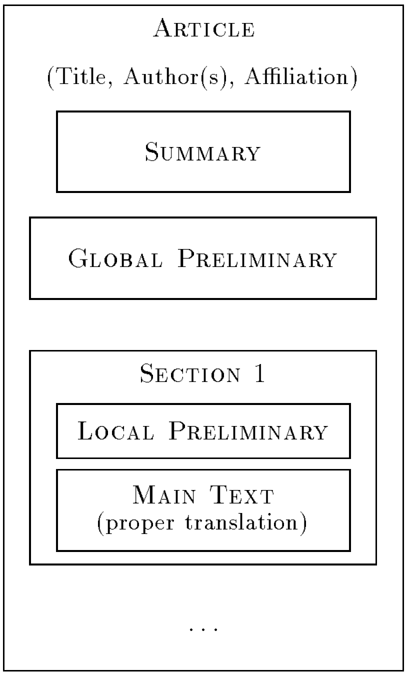
Analyze


Input
Statistic
Completion by new concepts (logical closure)
Division into preliminaries
Selection of way of repetition
Selection of facts to explain
Placement of explanations
Variable Process

Translate


Input
Movement of limit between verbal and symbolical translation
Choice of translation variants
Abbreviations for large terms
Proof analyzing
Preparation of fillers
Linearization by filling patterns

Output structure



Appendix B. Grammar of Mizar Abstracts

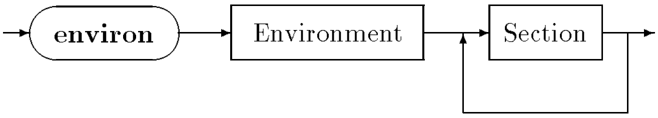
B.1 Reserved Words

aggregate
and
antonym
as
assume
attr
be
begin
being
by
canceled
case
cases
cluster
coherence
compatibility
consider
consistency
contradiction
correctness
def
deffunc
define
definition
defpred
end
environ
ex
exactly
existence
for
from
func
given
hence
hereby
holds
if
iff
implies
is
it
let
means
mode
non
not
now
of
or
otherwise
over
per
pred
prefix
proof
provided

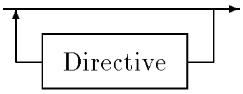
qua **reconsider** **redefine**
reserve **scheme** **selector**
set **st** **struct**
such **suppose** **synonym**
take **that** **the**
then **theorem** **thesis**
thus **uniqueness** **where**

B.2 Syntactic Diagrams

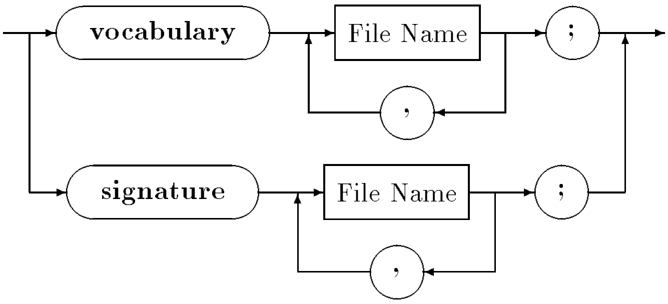
Article



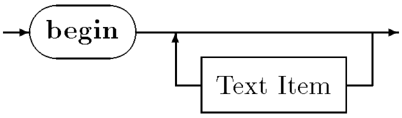
Environment



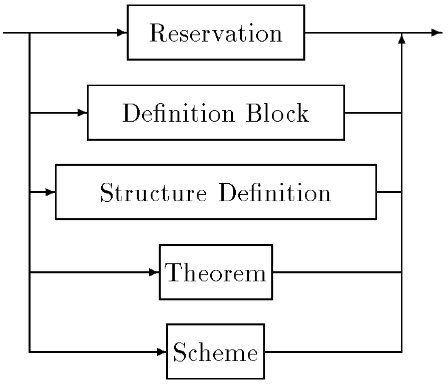
Directive



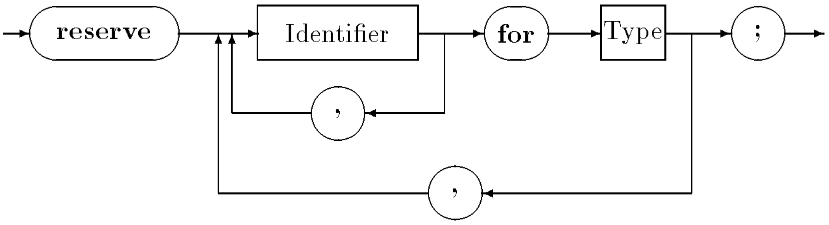
Section



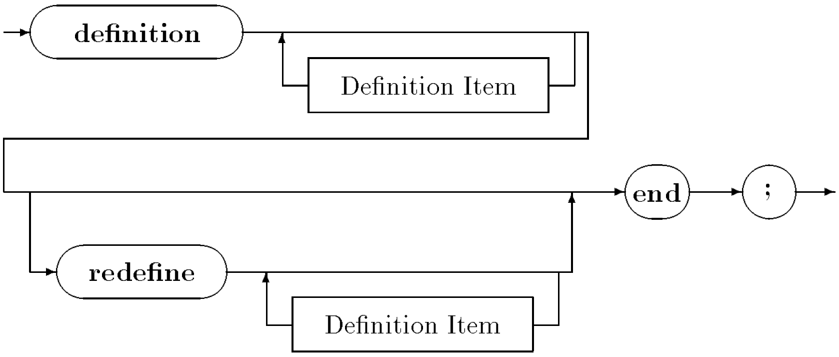
Text Item



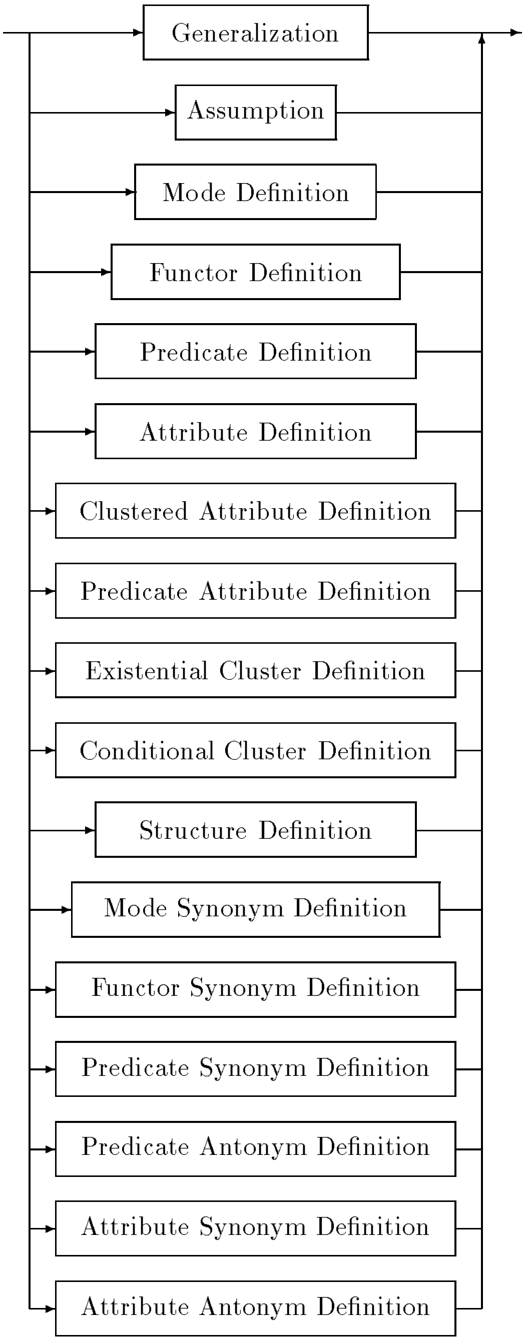
Reservation



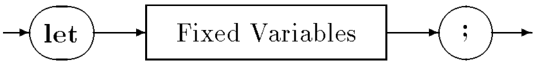
Definition Block



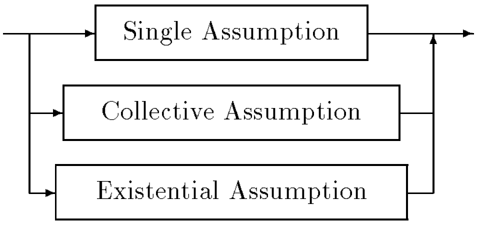
Definition Item


Generalization
Assumption
Mode Definition
Functor Definition
Predicate Definition
Attribute Definition
Clustered Attribute Definition
Predicate Attribute Definition
Existential Cluster Definition
Conditional Cluster Definition
Structure Definition
Mode Synonym Definition
Functor Synonym Definition
Predicate Synonym Definition
Predicate Antonym Definition
Attribute Synonym Definition
Attribute Antonym Definition

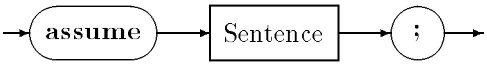
Generalization



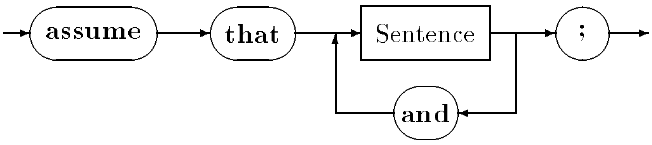
Assumption



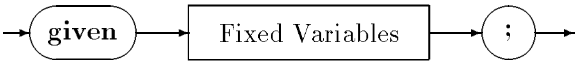
Single Assumption



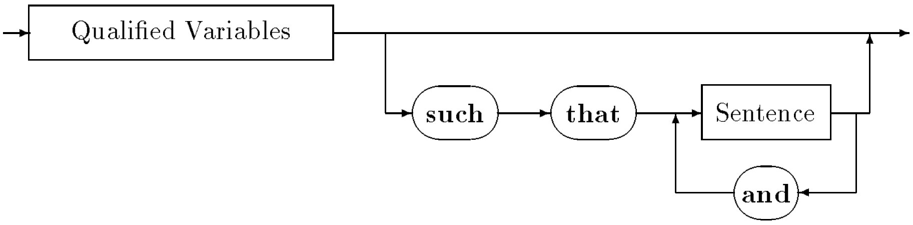
Collective Assumption



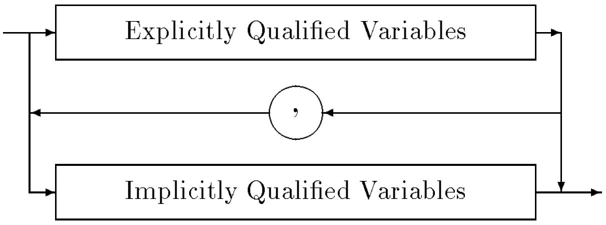
Existential Assumption



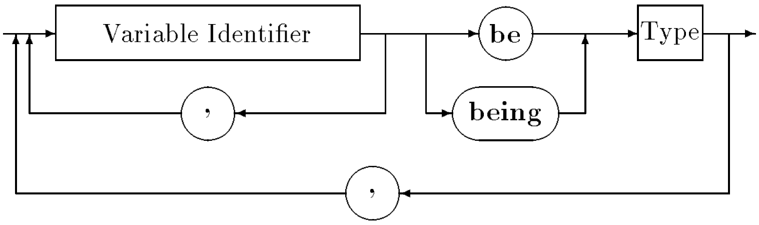
Fixed Variables



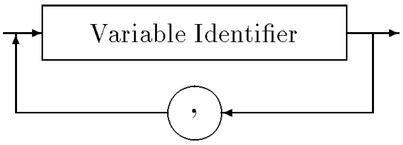
Qualified Variables


Explicitly Qualified Variables
,
Implicitly Qualified Variables

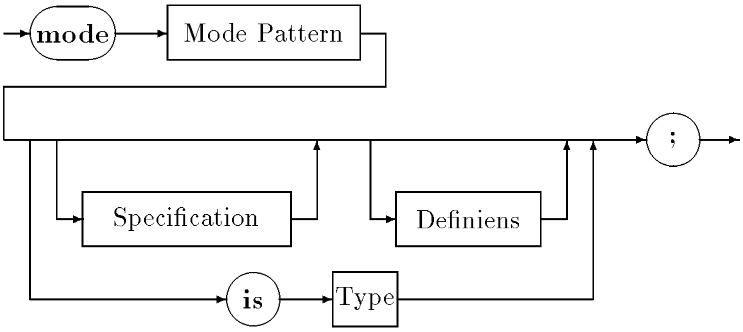
Explicitly Qualified Variables



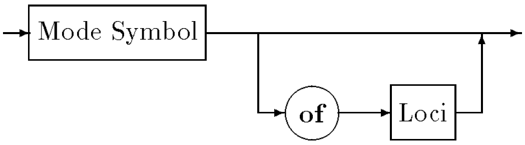
Implicitly Qualified Variables



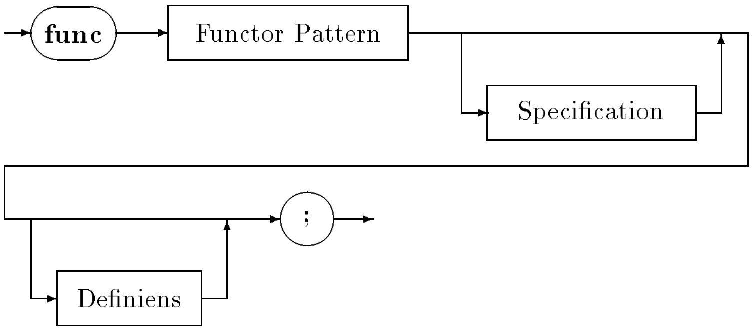
Mode Definition



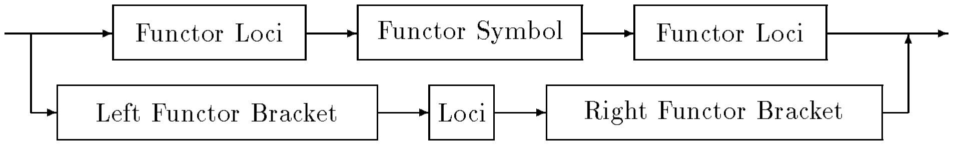
Mode Pattern



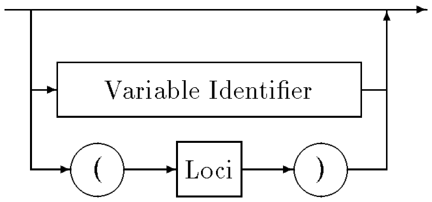
Functor Definition



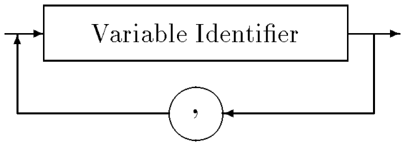
Functor Pattern



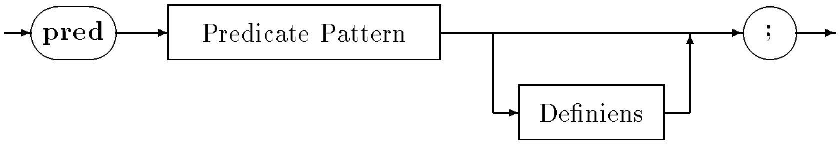
Functor Loci



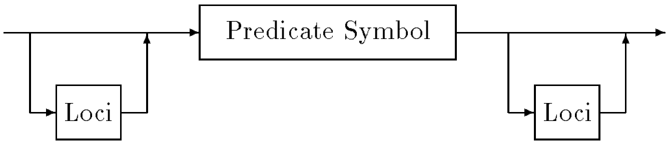
Loci



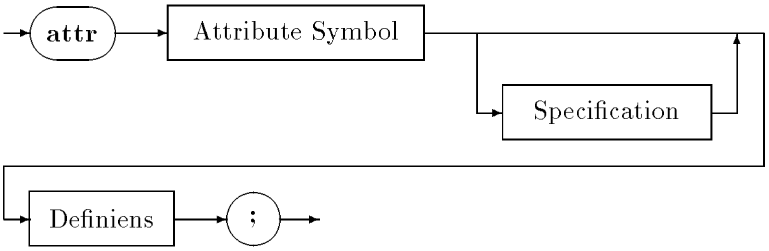
Predicate Definition



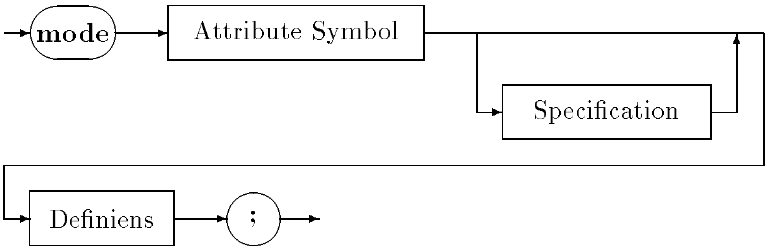
Predicate Pattern



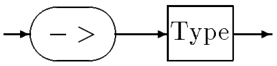
Attribute Definition



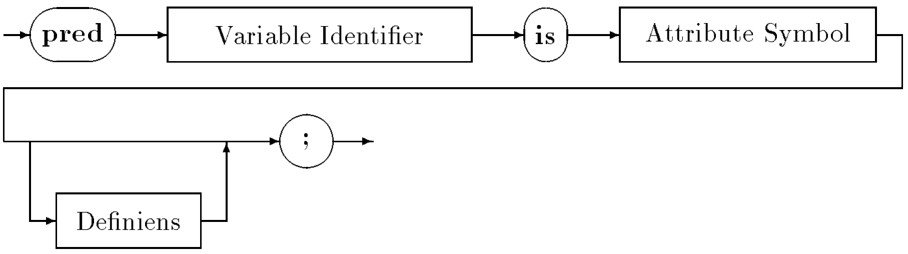
Clustered Attribute Definition



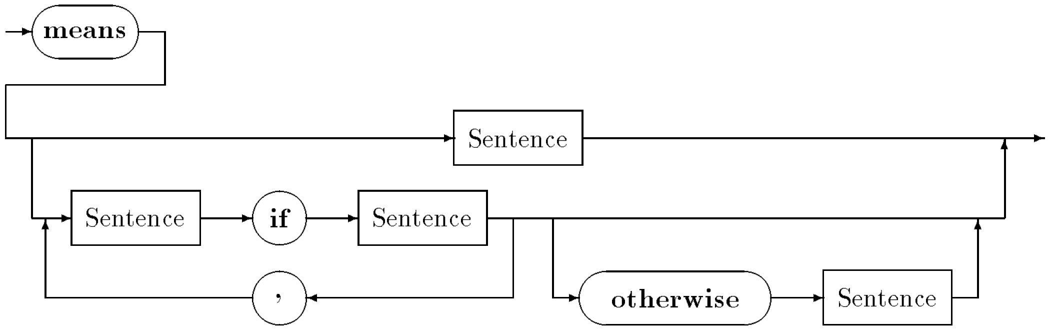
Specification



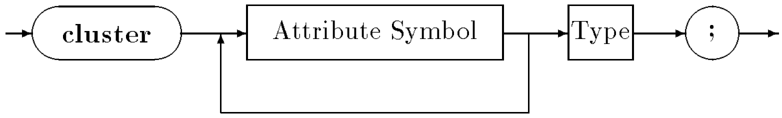
Predicate Attribute Definition



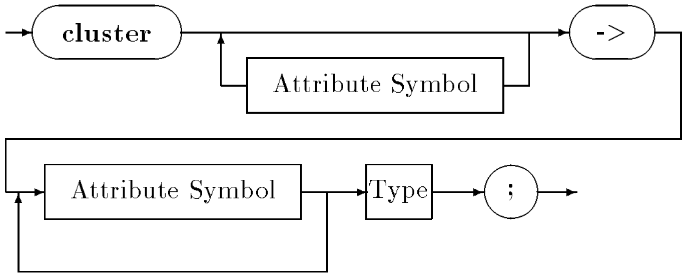
Definiens



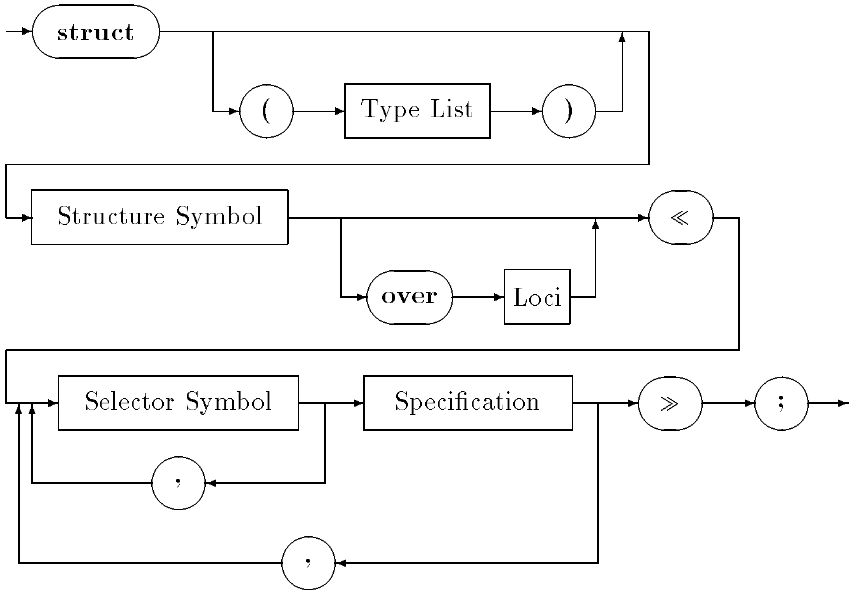
Existential Cluster Definition



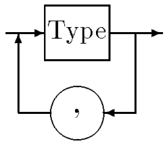
Conditional Cluster Definition



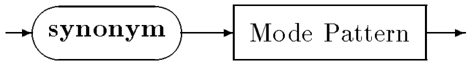
Structure Definition



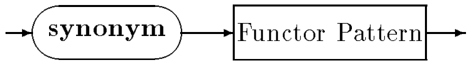
Type List



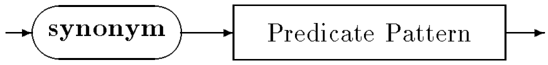
Mode Synonym Definition



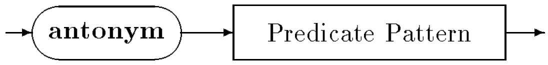
Functor Synonym Definition



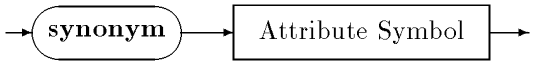
Predicate Synonym Definition



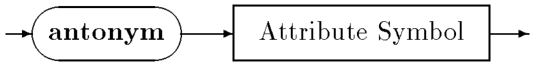
Predicate Antonym Definition



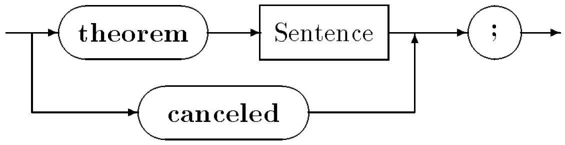
Attribute Synonym Definition



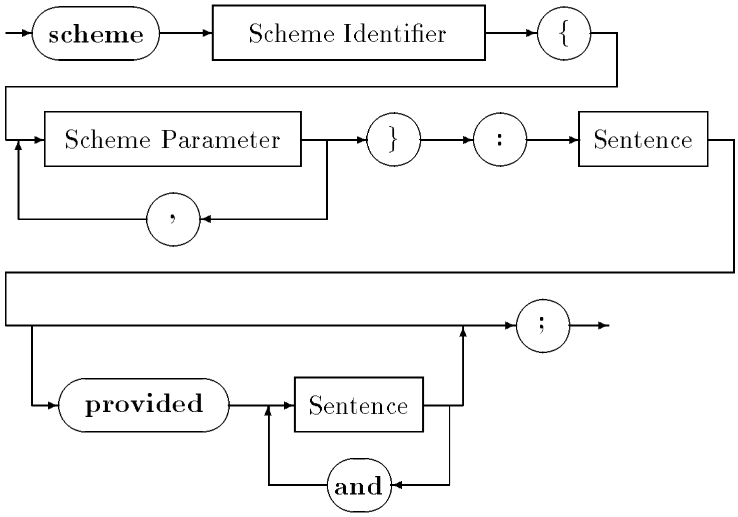
Attribute Antonym Definition



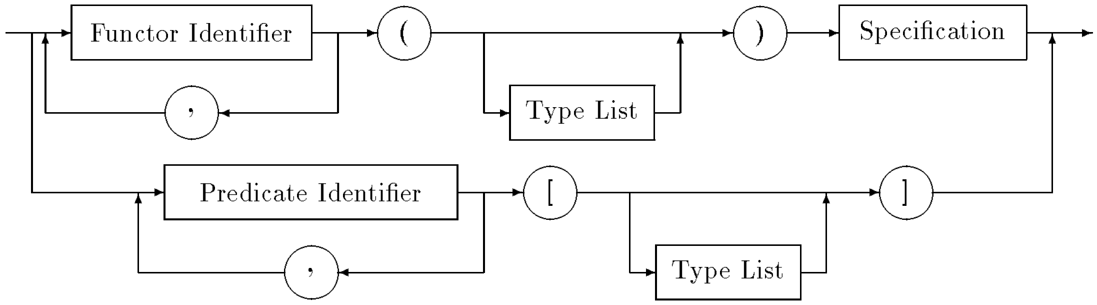
Theorem



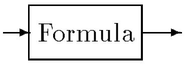
Scheme



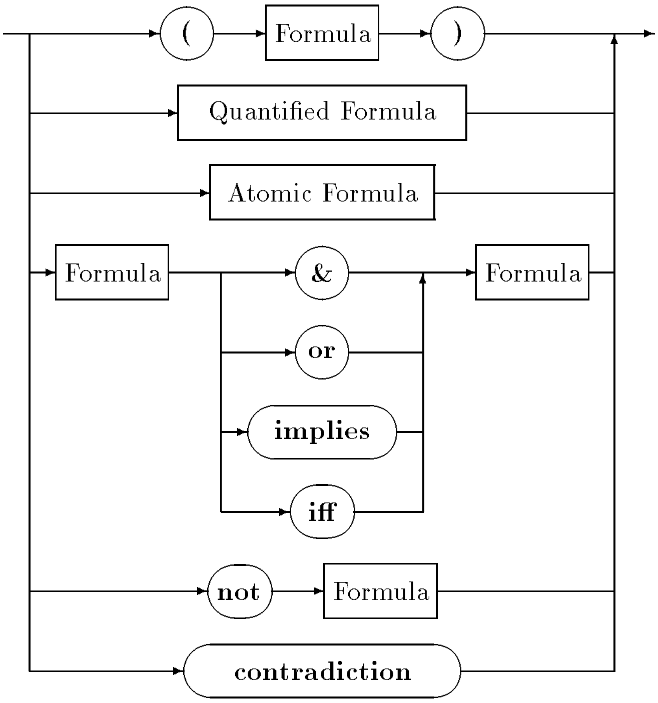
Scheme Parameter



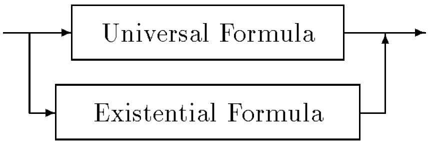
Sentence



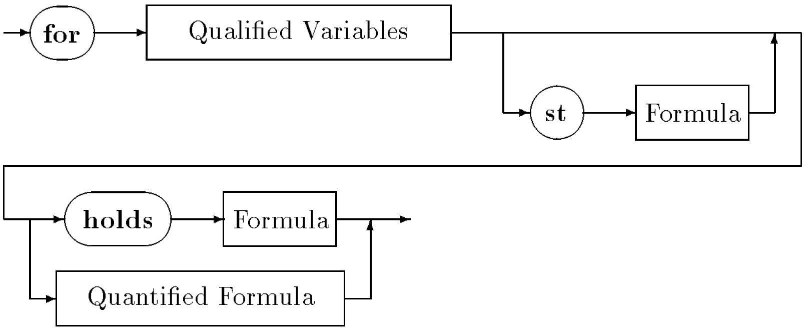
Formula



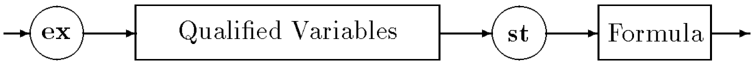
Quantified Formula



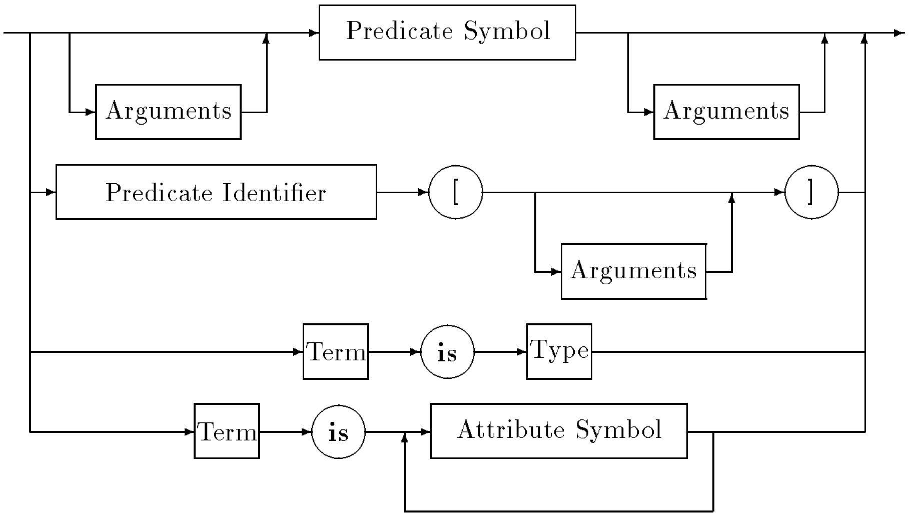
Universal Formula



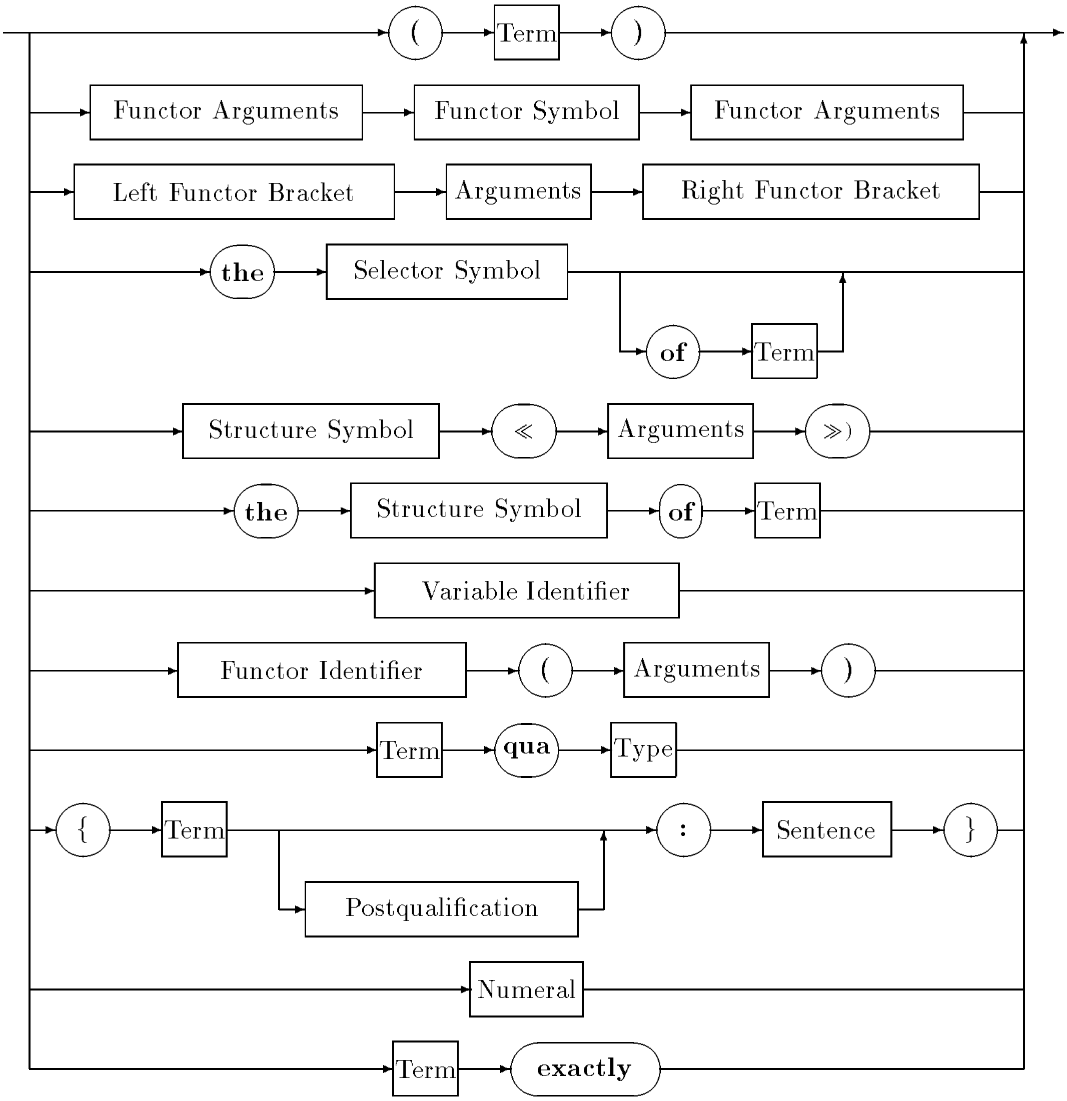
Existential Formula



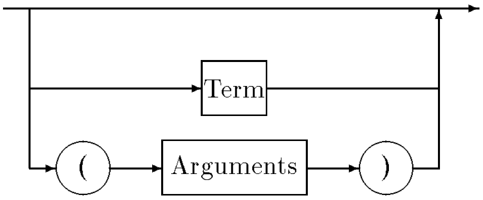
Atomic Formula



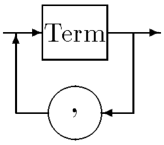
Term



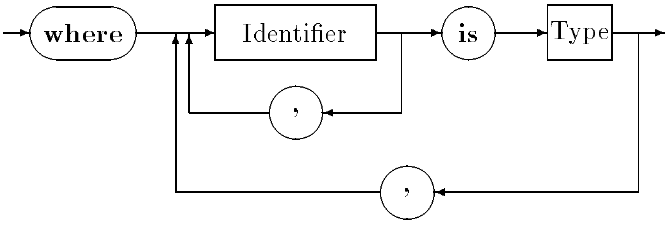
Functor Arguments



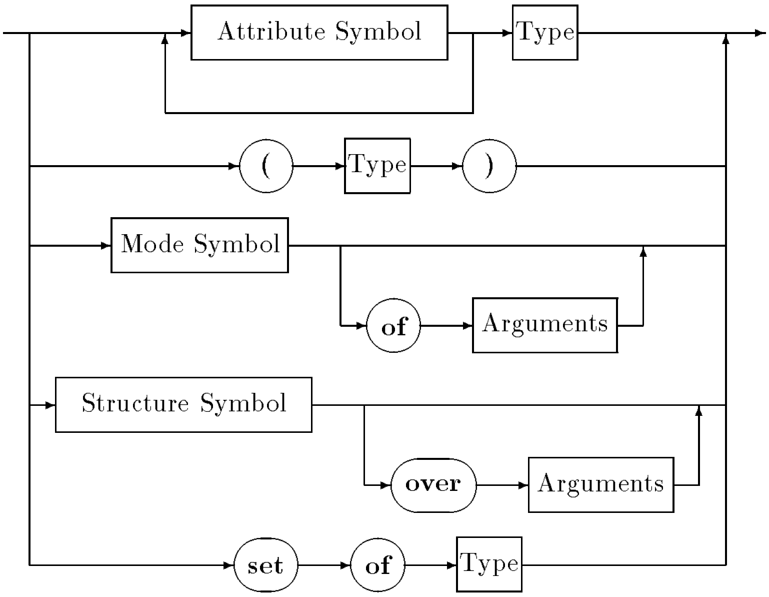
Arguments



Postqualification



Type


Attribute Symbol
Type
(
Type
)
Mode Symbol
of
Arguments
Structure Symbol
over
Arguments
set
of
Type