many-sorted language

Canonical name	ManysortedLanguage
Date of creation	2013-03-22 17:44:40
Last modified on	2013-03-22 17:44:40
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	12
Author	CWoo (3771)
Entry type	Definition
Classification	msc 03B70
Classification	msc 03C07
Classification	msc 03B10
Synonym	many sorted language
Synonym	many-sorted logic
Defines	sort
Defines	sort type

A many-sorted language is a variation of the classical first-order language. Whereas structures of a first-order language consist of a single universe, structures of a many-sorted language may contain many "universes", where each universe is "named" by a symbol called "sort", hence the name "many-sorted".

To formalize the notion of a many-sorted language, we start with a non-empty set S whose elements we call *sorts*. Let S^+ be the set of all (finite) non-empty words over S. Elements of S^+ are called *sort types* and are written as tuples. So instead of writing $s_1 s_2 \cdots s_n \in S^+$, it is written $(s_1, s_2, \cdots, s_n) \in S^+$.

The next item to be defined is the underlying signature of a many-sorted language. A signature Σ consists of

- a non-empty set S of sorts,
- a set F of function symbols, and
- a set R of (non-logical) relation symbols,

such that each element in $F \cup R$ corresponds to a sort type. In other words, there is a function $t : F \cup R \to S^+$, and for every $r \in F \cup R$, $t(r)$ is its sort type. An element $c \in F$ such that $t(c) \in S$ is called a constant symbol (of sort $t(c)$).

In addition to the signature Σ, we introduce additional symbols:

- the set V of variables, such that each sort $s \in S$ corresponds to a countably infinite subset $V_s \subseteq V$ of variables. In other words, there is a function $v : V \to S$, such that for each $s \in S$, $v^{-1}(s)$ is countably infinite. For each variable $x \in V$, its sort is defined to be $v(x)$.
- logical predicates: $\vee, \neg, \forall$
- the equality relation symbol: $=$, and
- the left and right parentheses: $($, $)$

Using Σ and the additional symbols above, we can build terms inductively as follows:

- each variable $x \in V$ is a term of sort $v(x)$

- if $f \in F$ is a function symbol of sort type $(s_1, \ldots, s_n)$, and for each $i = 1, \ldots, n-1$, t_i is a term of sort s_i, then $f(t_1, \ldots, t_{n-1})$ is a term of sort s_n.
- all the terms are built this way.

Finally, from the terms, we inductively define formulas:

- if t_1 and t_2 are terms of the same sort, then $(t_1 = t_2)$ is a formula
- if $r \in R$ is a relation symbol of sort type $(s_1, \ldots, s_n)$, and for each $i = 1, \ldots, n$, t_i is a term of sort s_i, then $r(t_1, \ldots, t_n)$ is a formula
- if α is a formula, then so is $(\neg\alpha)$
- if α, β are formulas, then so is $(\alpha \vee \beta)$
- if α is a formula and $x \in V$ is a variable, then so is $(\forall x(\alpha))$
- all the formulas are formed this way.

The signature Σ, additional symbols, and terms and formulas subsequently defined constitute what is known as the *many-sorted language* $L = L(\Sigma)$ on Σ.

As in first order language, the outer most parentheses may be eliminated without causing much harm, so that $(\neg\alpha)$ becomes $\neg\alpha$. In addition, we may introduce other familiar logical symbols $\wedge, \rightarrow, \leftrightarrow$, and $\exists$ in terms of $\vee, \neg$, and $\forall$. The specifics of how this is done can be found in the entry on first order language.

From this definition, we see at once that the classical first order language is a one-sorted language ($S = 1$). Sort types are identified with their lengths. Thus, the sort type of a function or a relation symbol is its arity.

Remark. It is not hard to show that a many-sorted language is not much different from a first-order language. Provided that V is countably infinite, a many-sorted language L can be "converted" into a first-ordered language L_1. Basically, all the function and relation symbols in L are in L', as well as the additional symbols such as variables and logical symbols. For each sort $s \in S$, we introduce a new unary relation symbol P_s in L_1. For any formula that contains a subformula of the form $\forall x\phi(x)$, we replace each occurrence of such a subformula by a formula of the form $\forall x(P_s(x) \rightarrow \phi(x))$, where x is a variable of sort s and ϕ is some formula in which x occurs as a free variable. The result is that L_1 becomes a one-sorted language.