What is Logical in First-Order Logic?

Boris Čulina
Department of Mathematics
University of Applied Sciences Velika Gorica
Zagrebačka cesta 5, 10410 Velika Gorica, Croatia
e-mail: boris.culina@vvg.hr

Abstract. In this article, logical concepts are defined using the internal syntactic and semantic structure of language. For a first-order language, it has been shown that its logical constants are connectives and a certain type of quantifiers for which the universal and existential quantifier form a functionally complete set of quantifiers. Neither equality nor cardinal quantifiers belong to the logical constants of a first-order language.

Keywords. logical constants; logical quantifiers; a functionally complete set of logical quantifiers

Starting with Tarski's definition of the concept of logical consequence from 1936 [Tar83] and the definition of the concept of logical constant from 1966 [TC86], the examination of logical concepts is dominated by an approach in which logical concepts are concepts that are in some way invariant to language interpretations.[1] There are differences in how the concept of invariance is formulated. See, e.g., [Bon14] for an overview and defence of such an approach in the case of the concept of logical constant. Of course, there are other approaches,[2] but this approach is dominant and here I will call it *the received view*. The received view, no matter how it was formulated, is based on the ontology of all possible interpretations, an insufficiently clear metaphysical concept. In mathematical logic, interpretations are usually realized in the world of sets - sets, relations, and functions are the values of the

[1]The basics of this approach can already be found in Bolzano from 1837 [RG14].
[2]See, e.g., [Mac17] for an overview in the case of the concept of logical constant.

interpretations. However, the set theory based on ZFC axioms does not have a clear intuitive basis that would tell us what is and what is not.[3] Therefore, interpretations in set theory are also insufficiently clear. Ultimately, these interpretations are translations of the language which interpretations we examine into the language of sets. In this translation, for example, the logical constants of the translated language are described by the same logical constants of the language of sets, which does not contribute to their better understanding. Thus, in my opinion, the received view, due to the use of vague metaphysical assumptions, is unacceptable.

This article is based on an understanding of logical concepts that is opposite to the received view. If we were to call the understanding of logical concepts in the received view an external understanding then this would be an internal understanding of logical concepts. The analysis of logical concepts that will be conducted here is based on the assumption that logic is always the logic of a language, how we apply the language, and how its parts are syntactically and semantically connected. The logic of a language is just the inner organization of the language together with external assumptions of its use. In this article, the analysis will be done for first-order languages. The exterior assumptions of an interpreted first-order language are: (i) the language has its own domain of interpretation – a collection of objects that the language speaks of, (ii) every constant denotes an object, and every variable in a given valuation denotes an object, (iii) every function symbol symbolizes a function which applied to objects gives an object, (iv) every predicate symbol symbolizes a predicate which applied to objects gives a truth value, $True$ or $False$. The only important thing for the logic of the language is that these assumptions are part of the specification of the language, not whether they are fulfilled. Thus, external assumption of the language use have no ontological weight here. The inner organization of a first-order language is determined by the rules of the construction of more complex language forms from simpler ones, starting with names, variables and function symbols for building terms, and with atomic sentences for building sentences. In these constructions we use special symbols which identify the type of the construction. With each construction, and thus the symbol of the construction, a semantic rule is associated that determines the semantic value of the constructed whole using the semantic values of the parts of the construction.[4]

[3] See, e.g., [Čul13].

[4] In a given interpretation and a given valuation of variables, the semantic value of a

The symbol of a language construction will be termed *logical symbol* or *logical constant* if the associated semantic rule is an internal language rule: the rule does not refer to the reality the language speaks of, except possibly referring to external assumptions of the language use. It may be objected that this description of the concept of logical symbol is not clear enough. But ambiguity is also present in the received view where the description of logical constants encompasses all interpretations of a language. However, while in the received view further refinement of the term "all interpretations" necessarily involves metaphysical assumptions, it will be shown below that the description proposed here is self-sufficient and clear enough to give us the answer in a concrete situation whether the symbol of a construction is a logical symbol or not. Since this is an internal language approach in determining logical symbols – an approach that, in addition to external assumptions about the language use, does not include the reality of which the language speaks – it is automatically topic-neutral. Everything defined in it as a logical term will be a logical term in the received view approach. Thus the received view gives only the necessary conditions for logical symbols.

Description of logical concepts in terms of language is present in the literature in various forms. However, as far as I know, it has neither been given sufficient importance nor has it been pronounced precisely enough. For example, in the review article [Sha06] on logical consequence, giving various criteria for the notion of logical consequence, Shapiro also mentions the following criterion:

> Φ is a logical consequence of Γ if the truth of the members of Γ guarantees the truth of Φ in virtue of the meanings of the logical terminology.

However, what "the meaning of logical terminology" means is not specified. Likewise, Quine in [Qui86], page 48, among other criteria, gives the following criterion for logical consequence:

> One closed sentence logically implies another when, on the assumption that the one is true, the structures of the two sentences

term is the object described by the term and the semantic value of a sentence is its truth value.

> assure that the other is true. The crucial restriction here is that no supporting supplementary assumption or information be invoked as to the truth of additional sentences. Logical implication rests wholly on how the truth functions, quantifiers, and variables stack up. It rests wholly on what we may call, in a word, the logical structure of the two sentences.

Although Quine is known for precision, this description is also not precise enough nor is it further specified. That Quine's approach is different from the approach in this article can also be seen below from the consequences derived. In contrast to the above and other similar descriptions known to me, in this paper the semantic extensional rules of the first-order language constructions are precisely specified. They give a clear criterion whether a construction symbol is a logical symbol or not, whether a sentence is a logical truth and whether a sentence follows logically from a set of sentences. An analysis of logical symbols of a first-order language follows.

Connectives of a first-order language give us one way to combine simpler sentences into more complex ones. Every connective, regardless of whether it is abstracted from a corresponding natural connective or not, is determined by a Boolean function $f : \{True, False\}^n \longrightarrow \{True, False\}$, where n is non-negative integer. This function describes how the truth value of the sentence composed by this connective depends on the truth values of sentences from which it is composed. Since Boolean functions are internal semantic functions of the language, functions independent of the reality the language speaks of, these connectives are logical symbols of the language. Of course, the well-known results on functionally complete sets of logical connectives show that in a first-order language we should not have connectives other than standard ones, for example, $\wedge$, $\vee$, $\neg$, $\rightarrow$ and $\leftrightarrow$. All other connectives can be defined using these.

Qualitatively different way of combining sentences to more complex sentences is by combining with quantifiers "for all" and "exists", which are symbolized by symbols $\forall$ and $\exists$. How to characterize this type of combination? Are there other quantifiers of this type? Can we express all of them by quantifiers $\forall$ and $\exists$ which are abstracted from natural language? The approach conducted here is inspired by the description of the universal quantifier in [GB93], page 40.

Let's take, for example, the quantifier $\forall$. From a sentence $S(v)$, where

v is a variable, not necessarily free in the sentence, by the symbol $\forall$ the more complex sentence $\forall v\ S(v)$ is built. In a given interpretation of the language and in a given valuation of all variables except v, the truth value of the sentence $S(v)$ depends on the valuation of the variable v. To determine the truth value of $\forall v\ S(v)$, we must determine the truth values of $S(v)$ for all valuations of v. We can get three sets of truth values: $\{True\}$ (for all valuations of x the sentence $S(v)$ is true), $\{False\}$ (for all valuations of v the sentence $S(v)$ is false) and $\{True, False\}$ (for some valuations of v the sentence $S(v)$ is true and for some valuations it is false). In the first case the sentence $\forall v\ S(v)$ is true, in other cases it is false. So, the quantifier $\forall$ is determined by the function that maps non-empty sets of truth values to truth values. The quantifier $\exists$ is of the same type: it is determined by the function that maps all sets which contain $True$ to $True$, and $\{False\}$ maps to $False$. Every such function that maps non-empty set of truth values to truth values determines an extensional construction. These functions will be termed *quantifier functions*. The corresponding syntactical symbol of the construction will be termed *logical quantifier*. Just as logical connectives are logical symbols because they are determined by internal functions that map truth values to truth values, so logical quantifiers are logical symbols because they are determined by internal functions that map sets of truth values to truth values. It will be shown below that these are the only logical quantifiers of a first-order language, which will justify the name given to them: “logical quantifier”.[5]

Since there are $2^3 = 8$ functions from the set of non-empty sets of truth values to the set of truth values, there are 8 logical quantifiers. However, they do not need to be introduced by special constructs in a first-order language because all the others can be defined using $\forall$ and $\exists$. Definitions are given in the following table ($\top$ is the label for *True*, $\perp$ is the label for *False*):

[5] In this paper, only type <1> quantifiers will be analysed (see, e.g., [Wes19]).

{⊥,⊤}	{⊥}	{⊤}	Logically equivalent form
⊥	⊥	⊥	$\exists v(S(v) \wedge \neg S(v))$
⊥	⊥	⊤	$\forall v\, S(v)$
⊥	⊤	⊥	$\forall v\, \neg S(v)$
⊥	⊤	⊤	$\forall v\, S(v) \vee \forall v\, \neg S(v))$
⊤	⊥	⊥	$\exists v\, S(v) \wedge \exists v\, \neg S(v)$
⊤	⊥	⊤	$\exists v\, S(v)$
⊤	⊤	⊥	$\exists v\, \neg S(v)$
⊤	⊤	⊤	$\forall v(S(v) \vee \neg S(v))$

It means that the set $\{\forall, \exists\}$ is a functionally complete set of logical quantifiers of a first-order language. Because $\forall v\; S(v)$ is logically equivalent to $\neg\exists v \neg\; S(v)$ and $\exists v\;\; S(v)$ is logically equivalent to $\neg\forall v\neg\;\; S(v)$, sets $\{\forall\}$ and $\{\exists\}$ are also functionally complete sets of logical quantifiers of a first-order language. Analogous to the results for connectives, this result shows that in a first-order language we should not have other logical quantifiers besides the standard ones, $\forall$ and $\exists$.

Is equality (the symbol $=$) a logical symbol of a first-order language? Let t_1 and t_2 be terms. Using the symbol $=$ the atomic sentence $t_1 = t_2$ is built. The corresponding semantic construction is as follows: in a given interpretation and a given valuation of variables, the sentence $t_1 = t_2$ is true iff t_1 and t_2 denote the same object. This is a construction that maps the semantic values of these two terms – denoted objects – to the truth value of the corresponding atomic sentence. However, this construction delves deeper into reality than the external assumption of the language which only says that each term denotes an object. To determine whether terms denote the *same* object, we must look at the reality that language speaks of. For example, the key to the Superman story is the claim that Superman = Clark Kent. To determine whether this is true or not, logic does not help us but we have to look into the comics reality. Or Frege's example: "The morning star = The evening star". We know which objects are denoted by these terms, but logic is not enough to determine that it is the same object – we need astronomical observations. Determination of the truth value of the sentence $t_1 = t_2$ using the semantic values of the terms t_1 and t_2 involves reality beyond external assumptions about of the language use. Hence, equality is not a logical symbol. The reason why the logicalness of equality is the subject of dispute[6] may lie in the fact that, unlike, for example, the comparison of numbers, we can state certain logical truths about equality. This is because the description of the symbol of equality includes language in one part – it mentions the denotations of terms. For example, the external assumption of of the language use is that in a given valuation, each variable denotes a specific object. So x and x will denote the same object. Therefore $\forall x\ x = x$ is a logical truth.

Since equality is not a logical symbol, quantifiers described by equality are also not logical symbols. Such is, for example, the quantifier "there is one and only one" (the symbol $\exists!$):

$$\exists! v\ S(v) \leftrightarrow \exists x S(x) \wedge \forall y \forall z (S(y) \wedge S(z) \rightarrow y = z)$$

We can also see in a direct way that such a quantifier is not a logical symbol – by examining the semantic rule of the associated language construction $S(v) \mapsto \exists! v\ S(v)$. In a given interpretation and valuation of all variables except v, the sentence $\exists! v\ S(v)$ is true iff the sentence $S(v)$ is true in exactly one valuation of v. We can describe this construction by a function that maps multisets composed of truth values to truth values.[7] If the multiplicity of $True$ is equal to 1, the function gives the value $True$, otherwise it gives the value $False$. This function, like the Boolean and quantifier functions, is an internal semantic function, a function that connects semantic values independently of the reality the language speaks of. However, the overall semantic rule of this construction includes the reality because the argument of the function, a multiset, cannot be formed without distinguishing objects from the reality. How many times a truth value has occurred cannot be determined without distinguishing valuations of v, that is, without determining when a valuation yields the same object and when it does not. And this requires, as with equality, knowledge of the reality the language speaks of, knowledge which goes beyond the external assumptions of the language use. This argument is easy to generalize. All cardinal quantifiers are not logical quantifiers, because the semantics of the language construction determined

[6] Quine's doubts and pro et contra arguments about the logical status of equality can be seen in [Qui86], pages 61–64.

[7] A multiset composed of truth values is a function from the set $\{True, False\}$ into the set of non-negative integers. The value of a multiset on a given truth value is called its multiplicity.

by such a quantifier is described by the same type of function as for $\exists!$ – by a function that maps multisets composed of truth values to truth values. With all these quantifiers, in a given interpretation and valuation of variables, the identification of the multiset on which the function acts includes reality in the same way as with $\exists!$. So these quantifiers are not logical symbols.

According to the standard recieved view that is clearly presented in [She12], logical constants are constants that are invariant to bijections between domains. They include all cardinal quantifiers, including infinite cardinals, which state how many objects satisfy a formula. In response to such a broad concept of logical constant, in which the distinction between mathematics and logic is lost, the articles [Fef99] and [Bon08] have emerged that widen the conditions of invariance, thus narrowing the concept of logical constant. In the criterion of invariance, Feferman replaces the notion of bijection with the notion of surjection, and this substitution leads precisely to the logical constants of first-order languages established here.

Like the concept of logical symbol, so the concepts of logical truth and logical consequence have a basis in the internal structure of a language. *Logical truth* is a sentence of a language that is true not in terms of what reality is but in terms of what kind of language we use to describe reality – it is truth determined by the internal semantic structure of the language. Eg. $\neg A \vee A$ is a logical truth, because its truth is determined by the internal structure of the language, in this case the semantics of the connectives $\neg$ and $\vee$. Also, that from a set of sentences $\{A_1, A_2, \ldots\}$ *logically follows* a sentence B, means that starting from the truth of the sentences $A_1, A_2, \ldots$ the internal semantic structure of language, not the reality the language speaks of, determines the truth of B. Thus, for example, the semantics of the conjunction $\wedge$ determines that B logically follows from $A \wedge B$. Of course, these are simple examples, and this internal language description of logical truth and logical consequence itself is not entirely accurate. But, as with the concept of logical symbol, in concrete and simpler situations it clearly determines whether a sentence is a logical truth, that is, whether a sentence logically follows from a set of sentences. This is, however, a good enough basis to develop a formal calculus of logical truths and logical consequence. For a first-order language, it is easy to show by a modification of Quine's argument [Qui86], Chapter 4, that this language concept of logical consequence can be described by a formal first-order logic calculus that is complete in terms of the received view. However, this proof includes Tarski's concept of logical consequence that refers to all interpretations of the language. But a proof can be carried out without it. Namely, from the language concept of logical consequence, examining the rules of which a standard complete formal calculus is composed, it is easy to get that from the formal derivability $A_1, A_2, \ldots \vdash B$ follows $A_1, A_2, \ldots \models B$, in the sense of the language concept of logical consequence. The reversal can be shown by contraposition: it should be shown that from $A_1, A_2, \ldots \nvdash B$ follows $A_1, A_2, \ldots \not\models B$. For a standard complete formal calculus, the condition $A_1, A_2, \ldots \nvdash B$ is equivalent to the condition of formal consistency of the set $\{A_1, A_2, \ldots, \neg B\}$. Likewise, for such systems, for example for the system of classical natural deduction, Henkin-type proof of completeness [Hen49] is valid. That proof gives an interpretation which is composed of the language symbols, an interpretation that has no ontological weight, and in which the statements $A_1, A_2, \ldots$ are true and B is false. According to the language concept of logical consequence, this means that $A_1, A_2, \ldots \not\models B$ is valid. This completes the proof. From this result, that in the case of first-order logic, the language concept of logical consequence can be described by some standard formal calculus, it follows that it coincides extensionally with the received view concept of logical consequence. Since logical truth can be described by logical consequence, it also follows from this result that logical truth can be described by some standard formal system of first-order logic.

References

[Bon08] D. Bonnay. Logicality and invariance. *Bulletin of Symbolic Logic*, 14(1):29 – 68, 2008.

[Bon14] D. Bonnay. Logical constants, or how to use invariance in order to complete the explication of logical consequence. *Philosophy Compass*, 9(1):54–65, 2014.

[Čul13] B. Čulina. Logic of paradoxes in classical set theories. *Synthese*, 190(3):525–547, 2013.

[Fef99] S. Feferman. Logic, Logics, and Logicism. *Notre Dame Journal of Formal Logic*, 40(1):31 – 54, 1999.

[GB93] A. Gupta and N. Belnap. *The Revision Theory of Truth*. MIT, 1993.

[Hen49] L. Henkin. The completeness of the first-order functional calculus. *Journal of Symbolic Logic*, 14(3):159–166, 1949.

[Mac17] J. MacFarlane. Logical Constants. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, 2017.

[Qui86] W. V. Quine. *Philosophy of Logic: Second Edition*. Harvard University Press, 1986.

[RG14] P. Rusnock and R. George. *Bernard Bolzano: Theory of Science*. Oxford University Press, 2014.

[Sha06] S. Shapiro. *Necessity, Meaning, and Rationality: The Notion of Logical Consequence*, chapter 14, pages 225–240. John Wiley & Sons, Ltd, 2006.

[She12] G. Sher. Logical quantifiers. In D. Graff Fara and G. Russell, editors, *Routledge Companion to Philosophy of Language*, pages 579–595. Routledge, 2012.

[Tar83] A. Tarski. On the concept of logical consequence. In J. Corcoran, editor, *Logic, Semantics, Metamathematics*, pages 402–420. Hackett, 1983.

[TC86] A. Tarski and J. Corcoran. What are logical notions? *History and Philosophy of Logic*, 7(2):143–154, 1986.

[Wes19] D. Westerståhl. Generalized Quantifiers. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, 2019.