Routledge Studies in Contemporary Philosophy

PHILOSOPHY OF LOGICAL SYSTEMS

Jaroslav Peregrin

ROUTLEDGE

Philosophy of Logical Systems

"In a tour-de-force survey of logical systems, Peregrin addresses the fundamental question of how logical systems relate to natural langue argumentation. The book develops the view that logical systems are models and challenges widespread assumptions about the nature of logical semantics."

– Georg Brun, University of Bern

"For the general student of philosophy or linguistics it is hard to understand why to do formal logics at all. By using a rule-theoretical access, Peregrin's book shows step by step how ideal models are necessary tools for understanding the form of valid reasoning, but no first results in a meliorating project of replacing natural by formal languages."

– Pirmin Stekeler-Weithofer, University of Leipzig

This book addresses the hasty development of modern logic, especially its introducing and embracing various kinds of artificial languages and moving from the study of natural languages to that of artificial ones. This shift seemed extremely helpful and managed to elevate logic to a new level of rigor and clarity. However, the change that logic underwent in this way was in no way insignificant, and it is also far from an insignificant matter to determine to what extent the "new logic" only engaged new and more powerful instruments to answer the questions posed by the "old" one, and to what extent it replaced these questions with new ones.

Hence, this movement has generated brand new kinds of philosophical problems that have still not been dealt with systematically. *Philosophy of Logical Systems* addresses these new kinds of philosophical problems that are intertwined with the development of modern logic. Jaroslav Peregrin analyzes the *rationale* behind the introduction of the artificial languages of logic; classifies the various tools which were adopted to build such languages; gives an overview of the various kinds of languages introduced in the course of modern logic and the motifs of their employment; discusses what can actually be achieved by relocating the problems of logic from natural language into them; and reaches certain conclusions with respect to the possibilities and limitations of this "formal turn" of logic.

This book is both an important scholarly contribution to the philosophy of logic and a systematic survey of the standard (and not so standard) logical systems that were established during the short history of modern logic.

Jaroslav Peregrin is a research professor at the Department of Logic of the Institute of Philosophy of the Czech Academy of Sciences and a professor at the Faculty of Philosophy of the University of Hradec Králové, Czechia. He is the author of *Doing Worlds with Words* (Kluwer, 1995), *Meaning and Structure* (Ashgate, 2001), *Inferentialism* (Palgrave, 2014), and *Reflective Equilibrium and the Principles of Logical Analysis* (together with V. Svoboda; Routledge, 2017). His current research focuses on logical and philosophical aspects of *inferentialism* and on more general questions of normativity.

Routledge Studies in Contemporary Philosophy

Spatial Senses
Philosophy of Perception in an Age of Science
Edited by Tony Cheng, Ophelia Deroy, and Charles Spence

Transhumanism and Nature
A Critique of Technoscience
Robert Frodeman

Freedom to Care
Liberalism, Dependency Care, and Culture
Asha Bhandary

Moved by Machines
Performance Metaphors and Philosophy of Technology
Mark Coeckelbergh

Responses to Naturalism
Critical Perspectives from Idealism and Pragmatism
Edited by Paul Giladi

Digital Hermeneutics
Philosophical Investigations in New Media and Technologies
Alberto Romele

Naturalism, Human Flourishing, and Asian Philosophy
Owen Flanagan and Beyond
Edited by Bongrae Seok

Philosophy of Logical Systems
Jaroslav Peregrin

For more information about this series, please visit: www.routledge.com/Routledge-Studies-in-Contemporary-Philosophy/book-series/SE0720

Philosophy of Logical Systems

Jaroslav Peregrin

Routledge
Taylor & Francis Group
NEW YORK AND LONDON

First published 2020
by Routledge
52 Vanderbilt Avenue, New York, NY 10017

and by Routledge
2 Park Square, Milton Park, Abingdon, Oxon OX14 4RN

Routledge is an imprint of the Taylor & Francis Group, an informa business

Library of Congress Cataloging-in-Publication Data
A catalog record for this book has been requested

ISBN: 978-0-367-40563-2 (hbk)
ISBN: 978-0-367-80863-1 (ebk)

Typeset in Sabon
by Apex CoVantage, LLC

Contents

Preface

I have been wrestling with the problems discussed in this book for a very long time. It was in the late eighties when I, at the outset of my lifelong engagement with the nature of linguistic meaning, became familiar with formal semantics, which was then in its heyday. I was delighted by the wealth of useful tools for the analysis and explication of meanings that it offered; but I was also dissatisfied with what I perceived as the naive ways in which the tools were often put to use. My book *Doing Worlds with Words* (with the subtitle *Formal Semantics without Formal Metaphysics*), which appeared in 1995, was my attempt to give an intelligible shape to this dissatisfaction (it was also the first book I wrote in English).

In the preface to the book I wrote the following confession:

> Pirmin Stekeler-Weithofer's book *Grundprobleme der Logik* 'awoke me from my dogmatic slumber' and caused me to contemplate my previous activities in the field of formal semantics and thereby see that they were simply the result of being held captive by a certain picture, namely by the picture drawn into our minds by modern formal logic. Before encountering Stekeler-Weithofer's book, I had seen philosophy and, in particular, logic as a way towards the firmest of truths, thereafter I realized that logic is a mere tool that can be used both to elucidate and to blur, and that philosophy cannot be made sense of in any other way than as the Wittgensteinian struggle with the bewitchment of our reason by our language.

I think this is still largely valid, and I am still grateful to Pirmin for helping me to direct my effort in this way. The present book, in a sense, can therefore be seen as a follow up to the old one. Though I am no longer very satisfied with how I put things 25 years ago, the basic driving force behind it continues to drive me further. What has changed is that while in the older book I targeted the application of certain logical tools to the analysis of the semantics of natural language, my target is now, more generally, logic itself and especially modern formal logic.

Again, I am excited by the fact that modern logic has gained access to a broad spectrum of new and useful (especially mathematical) tools, which have moved it a good deal forward, while I am frustrated by what I see as frequent misunderstandings of the tools, which in my eyes lead many logicians to misuse them and to thereby lead logic, if not directly into blind alleys, then into alleys where it has no business. And again, I attempt to give an intelligible shape to this – now more general – dissatisfaction.

This book was written within the parameters of the project supported by research grant No. 17–15645S of the Czech Science Foundation, which I carried out within the Institute of Philosophy of the Czech Academy of Sciences. I am grateful to all those who looked at the manuscript and gave me invaluable advice in its regard, namely to Pavel Arazim, Georg Brun, Vít Punčochář, Pirmin Stekeler-Weithfer, and Vladimír Svoboda.

1 The Transmutation of Logic

1.1 The Evacuation of Natural Language

Nobody would likely deny that logic underwent an essential rebirth during the twentieth century (some might even be inclined to say that this was the *birth* of a *real* logic[1]).

We can say, perhaps with a certain amount of oversimplification, that logic, since its very beginning, has been primarily concerned with the correctness of arguments – not in the sense that its *sole* business would be to characterize correct arguments, but that everything that is its business evolved out of its concern with their correctness. And as arguments are articulated in the medium of a language that is normally a natural language, logic was always concerned with natural languages and with the fact that some sentences in these languages follow from other sentences.

What happened in the twentieth century was that logic introduced and embraced – sometimes rather hastily – various kinds of *artificial* languages. The employment of the languages looked extremely helpful and it looked to be able to elevate logic to a brand new level of rigor and clarity. Therefore, most logicians simply vacated the old, squat shack of natural language and moved to one of the shiny new lofts of the artificial ones, almost completely forgetting about their previous dwelling-place.

Logic flourished in its new dwelling. For the first time, it could be developed with a mathematical rigor (as the artificial languages were basically mathematical structures) rather than as a collection of quasi-empirical comments on either the way human reasoning proceeds (or should proceed), or on actual human languages.[2] The liberation of logic from the yoke of psychology and its fruitful alliance with mathematics provided for the explosion of results that we know from the twentieth century.

1. Coffa (1991, p. 113), for example, commenting on the state of the art around 1900, when Russell tabled the idea of reducing logic to mathematics, writes that at this stage "mathematics was a reality and logic a project."
2. Which we can find in all logic treatises since Aristotle via Port-royal logic (Arnauld & Nicole, 1662) up until nineteenth century logical textbooks (like Lotze, 1874).

However, the change logic underwent in this way was in no way trivial, and it is also far from trivial to determine the extent to which the "new logic" only engaged new and more powerful instruments to answer the questions posed by the "old" one, and the extent to which it replaced them with new ones. And insofar as a replacement took place, it is not easy to see whether it was because the old questions were found to be ill-conceived or obsolete, or whether what was going on was that the shift in question was not really noticed and logic has moved somewhere without its protagonists duly reflecting it.

An optimistic picture is that nothing significant happened, that logic only sharpened its tools and delimited its subject matter with greater precision. This view is often underpinned by the opinion that the ultimate subject matter of logic is neither linguistic structures nor mental entities, but rather some ideal entities that are only imperfectly captured/expressed/represented by our parochially human means. From this viewpoint, logic, by its nature, *is* akin to mathematics – dealing with the very kind of abstract or ideal reality that is studied by algebraists. In this way, the engagement of artificial languages and mathematical tools only brings the nature of the subject matter of logic out into the open.

The point of departure of this book consists in the insight that such optimism is unwarranted, if not naive. The move from natural languages to their artificial replacement is a significant one that must be considered with due care. And to take logic as a type of mathematics is not viable. Consider the claim of Béziau (1994, p. 73):

> Universal Logic is a general study of logic in the same way as Universal Algebra is a general study of algebra. It is based on the fact that there is no One Logic or Absolute Laws of Logic, but rather a type of logical structures who are [*sic*] fundamental mother structures in the sense of Bourbaki. Logic is then an autonomous field of mathematics, with its own intuitions and concepts and which can survive and be developed without importing specific notions from other fields of mathematics.

It is certainly plausible that logic deals with certain kinds of structures and that we can have a very general, abstract theory of such structures that reveals some very general facts holding across different fields of logic.[3] However, does it follow that logic is a part of mathematics? A great deal of work in physics nowadays consists of solving systems of certain differential equations – but does that make physics into a part of mathematics? Mathematics is certainly an indispensable *tool* of physics, but this does not make physics its *part* – and so it is with logic.

3. Such generalizations have started to appear long ago. See, for example, Font and Jansana (1996) or Dunn and Hardegree (2001).

Doing mathematics, we may study all kinds of structures – it is only when it comes to applications that we find certain kinds of structures more pertinent than others. Logic, we can say, is basically a study of overt *reasoning* (and of arguments as its most basic manifestations) – a real-world activity in which we humans engage, and can do better or worse. Thus, any study of mathematical structures done under the heading of logic should be assessed according to its usefulness in helping us do it in a better, rather than in a worse, way.

1.2 Artificial Languages of Logic as Models

The problem, as I see it, is that the artificial languages that became the new dwelling-places of logic are artificial in the sense that they were built exclusively by means of our definitions.[4] Therefore, everything we can find out about them cannot be other than a consequence of the definitions. But how then can the investigation of the consequences of our – deliberate – definitions help us decide which arguments – namely, which steps in our reasoning – are correct?

The answer may not be too complicated: studying abstract structures constituted by definitions is what is done by pure mathematics, and mathematics can certainly help us with all kinds of problems in the real world. Of course, in order for mathematics to be able to do so it must be *applied* – a structure it studies must be fashioned as a model of a phenomenon in the real world adequate enough that the study of the structure can tell us something interesting about the phenomenon.

This invokes the idea that the artificial languages of logic can be seen as models of natural language (and of the argumentative practices of which the natural languages are vehicles), similar to natural scientists' models of natural phenomena. The models are much more exactly delimited and hence much more transparent than the languages we normally use to talk and to reason; but they are not, on the other hand, languages themselves in the fully-fledged sense. Thus, they are not substitutes for the real languages in the sense that we could shift over to them and forget about the natural languages. They can be useful only insofar as they are arguably adequate to what they are models of and insofar as the results of their analyses can help us deal with it.[5]

This asymmetric situation poses, it seems to me, brand new kinds of philosophical problems that are not always properly identified. The artificial languages are always *simulacra* of the natural vehicles of reasoning

4. Or, seen from a more Platonist viewpoint, that they are *identified* in terms of the definitions.
5. This is what I take to be the most basic message of Stekeler-Weithofer's (1986) book, which deeply influenced my view of the matter (see Preface).

and hence are – more or less – parasitic upon them. This is not to say that a lot of interesting logical work cannot be done within such artificial languages, but it is to say that this must be done with an eye on their status as models. Proving that something in an artificial language is inferable from some other thing will only tell us something – beyond the fact that this is merely the result of our definitions of the language – if we can be sure that the language, as a model, is adequate to what it models, that talking about the former can be seen as a proxy for talking about the latter.

Why are artificial languages bound to remain *simulacra* and not to ascend to the level on which they would be on a par with natural languages? (This is a question quite crucial for this book, so it deserves to be answered explicitly.) The point is that perhaps they could ascend as a matter of principle but not as a matter of fact. Natural languages' endowment with meanings is something that develops over millennia, during which time the languages become inseparably and integrally entwined with our "meaningful" practices; the artificial languages cannot make up for this in the time frame of a few years or decades. Therefore, artificial languages, insofar as they are to be treated as meaningful, are tied to natural ones by a semantic "umbilical cord."

And the adequacy of an artificial language to a natural one, which, to be sure, is crucial here, is not a simple concept. Of course, a model is not supposed to be an identical copy of what it models, it is supposed to be in some respects simpler, more transparent, and less vague. Thus, a certain level of *dis*similarity is assumed. For example, what should we make of a situation in which an artificial language (together with some superstructure like a calculus) that acts as the model does not have a property that we would like it to have (or, the other way around, has one we would like it not to have)? If a calculus, for example, is not decidable – what does it mean? Does this fact concern the criteria of correctness of arguments in our natural language or instead only in the artificial one – or does it have to do with the way we produce the calculus as a model of the real language?

1.3 The Reflective Equilibrium of Logical Laws

At the dawn of modern logic, some of its protagonists seem to have developed logical languages in the hope that these languages would be able to replace natural language, at least in the context of mathematical, or perhaps more generally, scientific, reasoning. This, I think, has only materialized to a very limited extent. When we engage in mathematical reasoning, for instance, it often suffices for us that we know, or believe we know, that it would be possible, as a matter of principle, to articulate the reasoning in a logical calculus (though this might be quite laborious and tedious, and it is not really something that we normally do).

In the previous section, we stressed the similarity of the artificial languages of logic and the mathematical models employed by the sciences.

Here we have an important *dis*similarity: there is supposed to be a feedback (though in reality perhaps only a very meager one) from the artificial languages on our actual reasoning. Hence, while natural phenomena do not care about our constructing their models, reasoning, as a social phenomenon, can be influenced by our establishment of its models, for the models can, in this case, act as *norms*.

Thus, though the picture of the artificial languages of logic as models of our actual reasoning is, as I put forward, very illuminating, we must keep in mind that, when it comes to its details, we need to be careful not to overstretch the analogy. The point is that, unlike in the case of the natural sciences, there may be a feedback loop: the model we build may influence what it models, and since, of course, this might prompt an update of the model, it could, in principle, initiate a kind of self-propelling spiral.

The fact that the model may act as a norm means that it can be used to amend the phenomenon that it models (in our case reasoning); that is, to do away with the discrepancies between it and the phenomenon modeled by means of tampering with the latter. On the other hand, it can be taken as a (useful) model only insofar as it is adequate to the phenomenon, which is to say if there are no relevant discrepancies or if the discrepancies are negligible. Thus, we have these two different requirements which appear to pull in opposite directions (if they are not directly inconsistent): we must fashion the model to reflect the actual process of reasoning, while at the same time we are supposed to adjust the process of reasoning so as to comply with the model.

In Peregrin and Svoboda (2017), we argue that this gives rise to a "dialectical" process aiming at what has come to be called the *reflective equilibrium*[6] – an equilibrium between empirical facts concerning reasoning and our theoretical articulation of what we perceive as its rules that results from the back-and-forth movement between considering the arguments which are ("intuitively") taken for correct and the tentative explicit articulation of the corresponding rules. Thus, *logical laws* worthy of the name are born out of this very process – out of the process of a theoretical articulation of rules implicit to our reasoning practices that aim at the reflective equilibrium.

Let me add that this does not cancel the asymmetry between natural languages and the artificial languages of logic that I stressed in the previous section (though it, in a sense, alleviates it). Though there is "two-way traffic" between the two kinds of languages, the artificial ones maintain themselves as *simulacra* and not as languages that could take over the role of the natural ones as direct vehicles of reasoning. Instead, they function as models to which we can outsource a lot of theoretical problems

6. The term was coined by Rawls (1971) in the context of ethics, though the same considerations for logic were already presented earlier, though not under this name, by Goodman (1983). See Brun (2014, forthcoming) for an overview.

connected with reasoning, but can do this only to the extent they are seen as adequate models of the natural languages.

1.4 The Nature of Logical Laws

Languages of formal logic, as I am going to present them in detail in the upcoming chapters, are first and foremost expedients of our effort to articulate the criteria of correctness of arguments and consequently of the explanation of the fact that some sentences follow from other sentences. However, we do not explain *why* it is the case that something follows from something else, we explain it only in the sense that we systematize the cases of following-from and in this sense we present every individual case as a piece in a larger mosaic.

To avoid misunderstanding, such a systematization makes room, of course, for explanations of cases of following-from in terms of other, simpler cases. We can explain that a conclusion follows from some premises in that we can decompose the step from the premises to the conclusion into a chain of simpler and more obvious steps. (This, needless to say, is what we do when we prove theorems in mathematics.) But we accordingly only reduce the more complex cases to simpler ones; we do not explain why the simpler ones hold.

This, again, is similar to natural laws. The law of gravity explains why a stone, released from my hand, falls to the earth. It explains it not in the sense that it would disclose some hidden essence of the stone that would make it fall down; it does so in that it subordinates it to a general law, where the law is just a generalization. In effect, it tells us that this particular stone falls to earth because it is what things always – at least insofar as we have been able to find out – do.

We cannot say *why* the law of gravity holds; we can only state *that* it holds and given this, we can use it to explain individual cases. And we consider rules of logic as articulated by the calculi in a similar way: we articulate the rules as generalizations from the individual, correct arguments. Thus, as we find out that plus minus all the arguments of the form

$$\frac{A \qquad \text{If } A \text{ then } B}{B}$$

in English are correct, we accept

$$\frac{A \qquad A \to B}{B}$$

as a logical law. Are, then, the laws of logic just empirical generalizations?

We have already seen that logical rules differ from ordinary empirical generalizations in that they can be used as *norms* of reasoning. In fact, there are *two* ways in which the rules can be considered normative. First, they are normative in the sense that what they generalize are already *rules*. Their instances are *correct* arguments – they are correct not because we logicians proclaim them as correct, they are correct before any logical theory is in play; they are such because human linguistic practices are essentially rule-governed. Logic is thus supposed to pick up rules that already govern our reasoning, though in their natural form they are implicit, and it is only logical theory that makes them explicit.

Moreover, the laws of logic are normative also in the second sense, in that they can be used as standards of assessment and criticism of actual argumentation. This is because what we do when we articulate them is not only to bring their implicit predecessors into the open but rather also to "finalize," streamline, and extrapolate them to the regions where they were not really operative. This is what happens during the process of moving toward a reflective equilibrium.

Given this, we cannot really say that logic addresses a reality behind the surfaces of natural languages. It builds on some very general rules (proto-rules?) that are constitutive of our linguistic practices but employs its huge apparatus to transform these implicit rules into something clear, unambiguous, and potentially binding. The calculi and other tools of modern logic we are going to discuss in the upcoming chapters are expedients of this enterprise, not a means of naming a hidden reality excavated from the depths of language.

In this sense, I think that the status of the laws of logic is similar to that of natural laws (despite all the other dissimilarities): they do not bring to light a hidden reality behind appearances but rather bring a system to the appearances (to "save" them, as the Ancients would put it[7]). Of course, the system cannot be chosen deliberately, it must *fit* the appearances – but there is no reason to think that it was there, as such, all the time already before we established it.

1.5 Philosophy of Logical Systems

Traditional philosophy of logic targeted the "logical part" of the vocabulary of natural language, which seemed to be responsible for the correctness of a crucial part of the arguments. There were problems that logic shared with philosophy of language (such as: how do the expressions of natural language, especially the logical ones, manage to mean what they do?), and there were problems of the conceptual apparatus engaged by

7. See Lloyd (1978).

logic (for example, the nature of truth). And, of course, there were the philosophical questions surrounding very central problems of logic, such as the criterion of correctness of arguments and thereby also the laws of logic. Later there came discussions about the possibilities of rejecting some of the traditional laws of logic and about alternative logical systems that might ensue from their rejection.

However, after the center of gravity of logic shifted from natural language to the artificial ones, the problems multiplied. Now, for example, the question of which arguments are correct in natural language decomposes into a question of which ones are correct in an artificial language (which is usually a matter of a calculus or a system of formal semantics) and which natural language arguments are adequately represented by the artificial language ones. It is necessary to clarify the status of the artificial languages, their relationship to our actual reasoning and to the natural languages, and the criteria that would let us say that the former is an adequate model of the latter.

One set of problems concerns the newly established artificial languages. Artificial languages, or theories in the languages, may have various properties (such as consistency, completeness, decidability, etc.) and whether a particular language has such a property can be more or less easily determined by mathematical methods. In contrast to this, in the case of natural language the presence or absence of such a property is often not only not so easily found out but also sometimes it is even not clear how to make sense of it. Take, for example, such a property as consistency. Tarski (1944) was convinced that natural language is not consistent because it contains all the ingredients needed to put together liar sentences. In contrast to this, consistency was usually supposed to be a condition *sine qua non* for an artificial language to be useful for logic; and in any case its presence or absence is an unambiguous matter, usually it is subject to a mathematical proof.

The question is what such a proof of consistency of a formal language can tell us about natural language. One possibility is that it tells us nothing, because the artificial language was built precisely to improve on natural language so that it is provably consistent. Another possibility is that its consistency also indicates that the natural language is consistent, at least if cleansed from some problematic cases, some such ballast as, for example, the liar sentences, which are not essential to it. Or there is a possibility that it tells us that we should dispose of the amorphous and potentially vicious natural language in favor of one that is exactly delimited and provably unproblematic.

Another set of problems concerns the very nature of the relationship between natural and artificial languages. The original idea was that the relationship be quite straightforward – logical expressions of the former being replaced by their more rigorous variants and the extralogical ones being abstracted away. But we will see that, especially on the level of predicate logic, this gave way to the artificial languages acquiring structures largely independent of those of natural languages. As a result, "logical analysis," as a process of finding a formula appropriate for a natural language sentence, became a nontrivial kind of "art."

In this way, the artificial languages of logic acquired lives of their own and lost their obvious dependence on natural language. Again, one possibility would be to see this as a process of emancipation of the artificial languages, which were on their way toward superseding the natural ones. Another possibility is to insist that artificial languages cannot but be mere *simulacra* and that their "independent life" cannot but remain highly limited. In this case, there still may be a choice of attitude: their study may be seen either as merely an indirect way of studying natural language, or they may be seen as being in some sense autonomous.

Anyway, it seems that the emergence of artificial languages, and the logicians' relocation of most of their problems into the context of these languages, creates new kinds of philosophical questions that are both unprecedented and yet worth being taken seriously.

1.6 Genealogy of the Artificial Languages of Logic

The key to the philosophical problems of modern logic thus appear to be an understanding of the artificial languages logic employs and of what appears to be their "independent life." The idea pursued in this book is that they were formed out of the rib of natural language in a process during which logical analysis or logical formalization became more than a mere replacement of words and expressions of natural language by their more rigorous regimentations.

The language of propositional logic, as the simplest artificial language, originated mostly from such replacements, and even during its employment for the purposes of logical formalization its operators can often be considered as mere proxies for their natural language counterparts. (Though in many cases it can also be employed in creative manners where its operators' roles become much more than such proxies.) But especially along the way between this language and the language of predicate logic there came about an important change: the latter have come to rest on some syntactic rules (concerning quantifiers), which have no direct counterparts in natural language. Hence, finding a formula regimenting a given natural language sentence necessitated correlating two nontrivially different syntactic structures, opening up a space for various alternative solutions.

In this way, the language of predicate logic ceased to be directly tied to natural language; hence it acquired a certain independence. And as the structure of such a language resulted from the effort to put together an efficient calculus, to understand why the structure came out as it did means to understand the calculi and the problems to which their constitution responded (as well as the new problems which the constitution brought about). This is the task to be undertaken in the following chapters: to anatomize the most basic logical calculi to show how the structure of the language that underlies them can come into being.

A further step in the emancipation of the artificial languages of logic from natural ones came with the establishment of formal semantics in the writings of Tarski, Carnap, and others. This introduction is sometimes depicted in such a way that it was only *via* it that the formal languages truly became *fully-fledged* languages, with not only a "syntax" but also a "semantics." I think that to endow a language with a "real" semantics is not at all so simple as assigning its expressions some set-theoretic constructs. I think then that we should see the establishment of formal semantics as an elaboration and refinement of the technical apparatus we put together to create (useful) models of natural languages as vehicles of our matter-of-fact reasoning rather than as promoting our artificial languages to the level of "genuine" languages.

Hence, the leading idea of this book is that the formal languages of contemporary logic, which now largely determine the direction of logical investigations, evolved out of the effort of logicians to make useful models of those structures of natural language that are important from the viewpoint of argumentation. I believe we can understand them better if we reconstruct this evolution. This is not to say that those who developed the formal languages usually saw themselves as quasi-natural scientists building increasingly self-standing models of the activity of reasoning. Undoubtedly, many of them saw themselves as penetrating to deeper structures of thought or reality by discovering hidden "logical forms"; but here I do not want to reconstruct their intentions or their own understanding of what they were doing but rather what I believe was really happening.

This means that in the upcoming chapters I will survey the story of the constitution of the common artificial languages of logic that, on the one hand, is well-known but which I must, on the other hand, retell in a way that is not so familiar. Thus, I beg the reader for understanding my rationale for repeating things that are all too familiar to them; that rationale being that I want to rearrange the mosaic that these familiar things constitute so as to open a new vista on the nature of modern, formal logic and on the new philosophical problems it opens up.

2 Correctness of Arguments

2.1 Arguments

Logic came into existence as a tool that was to help us discriminate between correct and incorrect arguments; and though it has undergone various kinds of transmutations since the time of its origin, we can still say, perhaps with a certain kind of oversimplification, that this discrimination is its most basic *raison d'être*.

An *argument*, as we will conceive of it in this book, has a list of *premises* (which, however, may also be empty) in which the premises are declarative sentences. It also has a *conclusion*, which, again, is a declarative sentence. Of course, various, more or less different, construals of the notion of argument can be put forward. First, someone might argue that it is not sentences that make up an argument but rather their meanings, propositions.[1] There is a sense in which this is obviously correct (meaningless sentences do not form an argument), but we cannot get hold of a proposition other than by means of a sentence, so even in this case we cannot but present the premises and the conclusion by means of sentences.[2] Second, it may be objected that what makes up an argument is something like beliefs that need not even be accurately expressible by sentences – but if there were anything like this, it could hardly be the subject matter of an explicit theory such as those produced by logic.

Third, somebody may object, under the influence of the Gentzenian sequent calculus, that there may be multiple "alternative" conclusions to an argument.[3] We can certainly think of such kinds of arguments too, but the single-conclusion ones appear to be more basic (indeed an attitude of holding a belief in force of holding other ones appears to be much more primitive than that of holding at least one of a list of beliefs). Fourth, an

1. We use the term *proposition* in a very shallow sense: a proposition is just the meaning of a declarative sentence (whatever it is).
2. Personally, I think that propositions are secondary to sentences usable in the assertive mode; but this is not the place to argue for it.
3. See, e.g., Restall (2013) for an elaborated defense of such a stance.

objection might be made that the structure of an argument may be more complicated than just premises and a conclusion.[4] Again, there is much to this, but we will assume that in the basic cases the simple structure is an acceptable oversimplification. Fifth, we might imagine arguments containing not only *declarative* sentences but also imperative or interrogative ones. I assume here that these are marginal cases, which we can afford to neglect.

So, given that an argument leads from some premises to a conclusion, we can, as it often happens, directly identify it with an ordered pair consisting of a sequence or a set of premises and a conclusion. The task now is to draw a dividing line between those arguments that are correct and those that are not. We will present arguments in the form of the sequence of premises followed by a conclusion separated from them by a horizontal line. The premises will be written below each other

$$\begin{array}{l} A_1 \\ \ldots \\ \underline{A_n} \\ A \end{array}$$

or alternatively besides each other

$$\frac{A_1 \ldots A_n}{A}$$

Sometimes we will also present arguments on a single line, separating the conclusion from the premises by the following symbol

$$A_1, \ldots, A_n \vdash A,$$

or by the word *hence*

$$A_1, \ldots, A_n;\ \textit{hence}\ A.^5$$

We will also consider "metaarguments": "arguments" the premises and conclusions of which are arguments that allow us to get new arguments from already accepted ones. We will write such a metaargument so that we list the arguments which constitute its premises separated by

4. Thus, for example, Toulmin (2003) holds that arguments have a more complex structure.
5. We will mostly use the former in the context of artificial languages, whereas we will use the latter in the context of natural ones.

semicolons ($A^1{}_1, \ldots, A^1{}_{n_1} \vdash A^1; \ldots; A^m{}_1, \ldots, A^m{}_{n_m} \vdash A^m$) followed by a double slash and the argument which constitutes its conclusion

$$A_1^1, \ldots, A_{n_1}^1 \vdash A^1; \ldots; A_1^m, \ldots, A_{n_m}^m \vdash A^m \,//\, A_1, \ldots, A_n \vdash A.$$

2.2 Which Arguments Are Correct?

We *presuppose* that some arguments in natural language are correct, while others are incorrect. That is, we assume a pre-theoretical fact, a fact that we want to "account for" by our logical theories and without which logic would lose its point. (More precisely, what the logical theories are after, first and foremost, is the *characterization* of the correct arguments, a *criterion* of the correctness of arguments.) If an argument is correct, we will also say that its conclusion *follows from* its premises. Thus, the term "follows from," unlike such terms as "is inferable from" or "is a consequence of," which will be used in more or less technical senses, will refer to the pre-theoretical facts. We will often write "following-from" as a name referring to this pre-theoretical relationship.

What does it mean that something follows from something else in a language such as English? And how can we find out that it is so? With a huge oversimplification we can say that to follow is to be taken to follow by a great majority of the speakers of the language in question. (This means not only that they would agree that it follows, but especially that they are ready to use it in argumentation and not to protest when somebody else uses it.) As a consequence, the most reliable way of finding out if something follows from something else would be a public poll (or an empirical research of what speakers of the language, as a matter of fact, take to follow from what); though practically speaking, it is often enough to rely on one's knowledge of the language and on the fact that the speakers of the language know what is generally taken to follow.

Of course, this "democratic" construal of the correctness of arguments requires some crucial qualifications. The first one is that the notion of the correctness of arguments will be heavily context-dependent. What may pass as a correct argument in one context need not pass as such in another context. It is probable that for each of the following arguments there is a context in which it may count for correct

Germans won the world championship in football several times
<u>The football team of Monaco never made it to the championship</u>
Monaco cannot beat Germany

<u>Every fish caught in this fishpond up to now was a carp</u>
Every fish in this fishpond is a carp

The door was unlocked
Somebody used a key

However, logic counts on the (admittedly, slightly idealized) assumption that there is a core of arguments that are "impeccable" in the sense that they count as being correct *in every* conceivable context. And logic – in a narrow sense – concentrates on this core.

A second qualification is that in the case of some kinds of arguments, social factors such as the "division of linguistic labor" famously tabled by Putnam (1975) will be effective – the correctness of some arguments may be unclear to the majority of speakers and they may delegate the verdict to some experts.

However, the most important qualification, without which the "democratic" account would tend to be dangerously misleading, is that such "intuitions" of speakers of a natural language may sometimes be incompatible. Thus, for example, the majority may hold that an argument is correct, that its correctness necessarily brings about the correctness of another argument, and yet that the other argument is not correct. (In reality, "intuitions" may often be much less obviously incompatible.) In such a case, one of the "intuitions" will have to be rejected as wrong.

Here it is also important to note that some arguments are taken as more "basic" than others by the speakers. The ignorance with respect to the (in)correctness of the most primitive arguments will compromise one's status as a competent speaker of the language in question. (These are the arguments that can be called "analytic" in this pragmatic sense of the word.) The ignorance with respect to some other arguments will be more tolerable – their (in)correctness will be taken as a more or less "derivative" matter that can be subjected to some demonstration. And the "intuitions" which might be diagnosed as wrong are those of the latter kind.

Take, for example, the argument

When it rains, the streets are wet
It rains
The streets are wet

This is the kind of "basic" argument such that if someone were to doubt that it is correct, it would be reasonable to suspect that she does not understand the connection *if-then*. In contrast to this, the "intuitions" regarding an argument such as

When the streets are wet, it rains or it snows
When it does not rain, it snows or the streets are not wet

would probably not serve as a touchstone of understanding the sentences constituting it – if somebody were to doubt its correctness, the most probable moral would be that she needs some logical schooling. This is also how logic, despite the "democratic" construal of correctness, can teach us that we reason wrongly – that our holding of an argument for (in)correct is incompatible with our holding of some other arguments for (in)correct.

But should logic not also correct our errors with respect to "primitive" arguments? Should it not tell us that, for example, holding cases of *modus ponens*, namely arguments of the form A, $A \rightarrow B \models B$, for correct is right, whereas holding cases of *affirming the consequent*, that is arguments of the form B, $A \rightarrow B \models A$, for correct is wrong? Well, the problem is that a natural language, like English, does not contain $\rightarrow$, for which this would be provably appropriate, but instead of it something like *if-then*. (Moreover, logic should *not* tell us that B, $A \rightarrow B \models A$ unless it is guaranteed that $\rightarrow$ is, say, the classical implication.) And does *if-then*, from the viewpoint of correctness of arguments, behave enough like $\rightarrow$? In fact, it most probably does, but this is the case only insofar as it does comply with *modus ponens* and not with *affirming the consequent*. Hence, in case the great majority of English speakers were to use *if-then* as complying not with *modus ponens* but rather with *affirming the consequent*, there would be no reason to diagnose this as wrong – for in such a case there would be no reason to think that *if-then* means something close to "$\rightarrow$."[6]

Here are some examples of arguments that I think would be readily accepted by English speakers:

(A1) <u>Fido is a dog</u>
Fido is an animal

(A2) Fido is a dog
<u>All dogs are animals</u>
Fido is an animal

(A3) Fido is a dog
<u>All dogs are dangerous</u>
Fido is dangerous

(A4) <u>Fido is a dog and he is sick</u>
Fido is a dog

(A5) Fido is sick
<u>If Fido is sick, he needs medicine</u>
Fido needs medicine

(A6) Fido is either sick or just sad
<u>Fido is not sad</u>
Fido is sick

6. See Peregrin (2010b).

(A7) Fido is neither sick nor sad
Fido is not sick

(A8) Fido is a dog
Fido does not live on Saturn

A correct argument is sometimes taken to be an argument that is *truth-preserving*; that is, it guarantees the truth of the conclusion in case the premises are true. Nothing that we say here is incompatible with this, but we do not want to commit ourselves to deriving the concept of the correctness of arguments from that of truth or truth-preservation. The minor reason is that arguments that are generally taken for correct probably would not quite coincide with the truth-preserving ones;[7] the major one is that truth is such an enigmatic concept that putting it into the very foundation of our conceptual edifice could make it too wobbly.

2.3 Accounting for the Correctness of Arguments

Given that there is an unlimited number of sentences in natural language, there is an unlimited number of arguments. (This is not to say that we would literally need to count with an infinite number, as it is clear that nobody would ever use an argument with a million premises or a sentence consisting of a million words. However, there is no principled limit to the length of something that still counts as a sentence or an argument. More importantly, the number of arguments is such that they do not form a collection that could be easily surveyed.) This means that it is simply not possible to give a *list* of all correct arguments.

An obvious solution to this problem is a classification: dividing arguments into some relatively small number of types such that some of them would comprise only correct arguments, while the others would comprise only incorrect ones. Note that if such a classification were to really help us distinguish between correct and incorrect arguments, the *type* of an argument would have to be something readily determinable – it would be of no help to know which types comprise correct arguments and which comprise incorrect ones if we were not able to determine which is the type of a given argument.

For this reason, we usually classify arguments according to their *forms*, where a form amounts to the argument's "surface" – to what we can see when the argument is presented to us. As a first approximation, we can say

7. There are certainly many truth-preserving arguments that would be considered as at least awkward (such as that there is no connection between what their premises talk about and that which is talked about by their conclusion); and there are probably also arguments that are taken for correct despite not being guaranteed to be impeccably truth-preserving (such as (A8)).

that a form of an argument may be represented by the argument with some of its parts left out, replaced by meaningless symbols which we will call *parameters*. Thus, we get a form of the argument (A3) (one of its forms, that is) by replacing the word "dangerous" with the meaningless symbol *P*:

(AF3) Fido is a dog
All dogs are *P*
Fido is *P*.

In fact, we usually produce forms of arguments on a background of some kind of grammatical theory telling us which parts of sentences "go together" (so that we see it as being okay to replace *a dog* or *is a dog* by a single letter, but not *Fido is*); which different parts count as merely different forms of the same part (so that *a dog* in the first sentence can be replaced by the same parameter as *dogs* in the second); which parts are of different categories (so that *Fido* should be replaced by a different kind of letter than *is a dog*), etc.

A distinguished kind of form is what is usually called *logical* form, which is based on the division of the vocabulary of the language we consider into a logical and an extralogical part (plus, perhaps, some expressions that are neither, for they serve merely auxiliary, syntactical purposes). A logical form of an argument, in the simplest case, is then the form in which all extralogical expressions are replaced by parameters. There is no canonical way of drawing the dividing line, though some words are standardly taken as logical (*and*, *if-then*, *everything*, etc.), whereas many others are standardly taken as extralogical (*table*, *dancing*, *awkwardly*, etc.).[8]

A logical form of (A3) might be

(ALF3) a is P
All P are Q
a is Q.

(The same logical form, for that matter, pertains also to (A2).) Let us call a form *valid* iff all arguments of the form are correct; and let us call it *counter-valid* iff all arguments of the form are incorrect. Can we have a set of forms such that

- every argument is of one and only one of the forms;
- every form is either valid or counter-valid?

8. There are certain (not quite decisive) characteristics of the logical words of a natural language: they should play a nontrivial role in arguments and they should be topic-neutral, namely occur in arguments independently of their subject matter. Words like *always*, *the*, *probably* do not seem to be on any one side of the divide, at least not obviously.

In particular, can logical forms play this role?

What might seem to get in the way of this project is that which, since de Saussure (1931), has been called "the arbitrariness of the linguistic sign": there is generally no connection between how a sign looks and what it signifies. Therefore, it seems there is no way to tell whether an argument is valid on the basis of its surface form. But this is not a grave obstacle: for we do not want to pick up types of arguments that would be (counter-)valid in any conceivable language (true, such a task would make no sense!), but rather ones that happen to be (counter-)valid in a fixed language – that is, the current variety of English. And in this case we can certainly find kinds of arguments that are valid, such as the logical form of (A4):

(ALF4) $\underline{A \text{ and } B}$
$A.$

This, to be sure, depends on the fact that *and* in English means what it does, and is not guaranteed that it remains valid should the meaning of *and* shift; but this does not compromise the fact that in contemporary English arguments of this form at least appear to be of this kind.

So we can find some valid forms and thus delimit some sets of correct arguments, and we can also find some counter-valid forms, such as

$\underline{\text{Neither } A, \text{ nor } B}$
$A.$

Can we specify a set of forms such that each of them would be valid or counter-valid and every argument would be of this form? Take the argument

(A9) The sun is shining or it is raining
The sun is shining.

This, obviously, is not a correct argument. Its logical form is

(ALF9) $\underline{A \text{ or } B}$
$A,$

which is certainly not valid – but is it counter-valid? It would seem not, for the scheme would have some correct instances, for example

(A10) The sun is shining or the sun is shining
The sun is shining.

Of course, this argument has, aside from the previous logical form, also another logical form that is valid, namely

(ALF10) $\dfrac{A \text{ or } A}{A}$,

which can lead us to the conclusion that an argument is not correct if it has *no* valid logical form. Can we say that an argument is correct iff it has *a* valid logical form; or are there arguments that are correct despite having no valid logical form, that is, arguments that are correct, but not *logically* correct?

2.4 Correct, but Not Logically Correct, Arguments?

Consider the argument (A1). Is it correct? I think that it would be seen as correct by English speakers (if one were to argue from its premise to its conclusion, it could be hardly challenged). Moreover, it seems to be an instance of the valid schema

(AF1) $\dfrac{a \text{ is a dog}}{a \text{ is an animal}}$

This schema, however, is not a *logical form* of the argument. A logical form would be

(ALF1) $\dfrac{a \text{ is } P}{a \text{ is } Q}$

and it is clearly *not* valid. Does the argument have another, valid logical form, so that we can keep our conjecture that to be correct is to have a valid logical form?

At first sight, it would seem that there is no alternative logical form. However, some logicians are likely to suggest that the valid logical form it has is

(ALF1′) a is P
$\dfrac{\text{Every } P \text{ is } Q}{a \text{ is } Q.}$

The reason, they would insist, is that the argument (AF1), as it stands, is *not* correct. We, they would insist, take it to be correct because we take for granted that it contains one more covert premise, namely

Every dog is an animal.

However, this argument is shaky. Why should we assume that there is this covert premise? The argument seems to hold by itself – it is enough to assume that the words *dog* and *animal* mean what they do in English. And to assume that sentences constituting an argument have fixed meanings appears to be not only unproblematic but also in fact inevitable.

The only reason for stipulating the covert premise appears to be precisely the need for safeguarding the conjecture that an argument is correct if and only if it has a valid logical form. But is it reasonable to save it at the cost of making logical forms of arguments so removed from their surface forms?

Suppose that we do not accept this. Then we will distinguish between arguments with valid logical forms, which we will call *logically correct*, and arguments that are correct but do not have valid logical forms. This seems to be much more natural than trying to squeeze every correct argument into a valid logical form in the Procrustean way.

If we thus divide arguments into those that are correct logically and those that are correct for extralogical reasons, we can restrict the attention of logic to the former kind. However, we can still take into account the previous considerations as indicating that there is a way of "transforming" extralogically correct arguments into logically correct ones by encapsulating their "extralogical content" into additional premises. (By this we do not admit that extralogically correct arguments are "in fact" logically correct, we claim that in the vicinity of an extralogically correct argument there is always a logically correct one that can be sometimes used as its proxy.)

In this sense, we can reduce the correctness of some arguments (the "logically correct ones") to validity of logical forms; and we can reduce the correctness of other arguments to such validity in an indirect way. What about incorrectness? Can we say that any incorrect argument has a counter-valid form or can be reduced to one that does?

This, unfortunately, does not seem to be the case. The logical form of the incorrect argument (A9) appears to be (ALF9), which is certainly not valid, but it is also not counter-valid (and there does not appear to be another one that would be able to do better). Thus, its logical form (which is such that it has both correct and incorrect instances) does not tell us anything about its (in)correctness.

Thus, there appears to be an asymmetry: while correctness of arguments can be accounted for in terms of logical forms (at least as a matter of principle), for incorrectness this is not possible.[9] Of course, even accounting for correctness of arguments yields us, indirectly, an account of incorrectness. If we manage to characterize, exhaustively, correct arguments, we thereby characterize the incorrect ones as those that defy this

9. Cf. Svoboda and Peregrin (2016).

characterization. In other words, if we find the criterion of correctness of arguments, we have *eo ipso* a criterion of incorrectness.

But even if we restrict the task of logic only to account for *correct* arguments (directly for the logically correct and indirectly for other correct ones), there appears to be another problem. Though we can account for logical correctness in terms of valid logical forms, the trouble is that valid logical forms are still too numerous to be simply listed. A solution is something like an "axiomatic system," listing some basic valid forms and giving us ways to derive further valid forms from already given ones. But before we look at this in greater detail, let us pay some attention to the means that we use to articulate logical forms.

3 Languages of Logic

3.1 Simple Logical Symbols: Parameters and Constants

We saw that to articulate kinds of arguments, or their forms, we need some symbols to substitute for those parts of arguments that are of no interest, symbols like the *A*, *B*, *a*, *P*, or *Q* that we used in the previous chapter. It is important that these symbols are "empty," that they are not connected with any specific meanings. Such symbols are sometimes called *variables*, but as we want to reserve this term for a more specific kind of symbol we have employed another term. We use the word *parameter*.

The use of parameters is as old as logic. Take, for instance, the following passage, from Aristotle's *Prior Analytics*:

> First then take a universal negative with the terms *A* and *B*. If no *B* is *A*, neither can any *A* be *B*. For if some *A* (say *C*) were *B*, it would not be true that no *B* is *A*; for *C* is a *B*. But if every *B* is *A* then some *A* is *B*. For if no *A* were *B*, then no *B* could be *A*. But we assumed that every *B* is *A*. Similarly too, if the premise is particular. For if some *B* is *A*, then some of the *A*s must be *B*. For if none were, then no *B* would be *A*. But if some *B* is not *A*, there is no necessity that some of the *A*s should not be *B*; e.g. let *B* stand for animal and *A* for man. Not every animal is a man; but every man is an animal.

We can see that the letters *A*, *B*, and *C* (in the English translation) are employed in a way similar to the one in which we employed them earlier: they allow us to say not merely that if every horse is a mammal then it follows that some particular horse is a mammal, but, more generally, that from every premise of the form *Every B is A* there follows the conclusion *Some B is A*.[1]

Parameters are thus the principal means of *abstraction*. By abstracting away certain parts of a sentence we reach a form (schema, template, pattern, etc.), which, needless to say, is not itself a sentence and cannot be considered, by itself, as either true or false. A schema can be *valid*,

1. Not that Aristotle would put it this way. The explicitly metalinguistic way of putting things is mostly a modern innovation.

which means that all its instances are true, or it can be *counter-valid*, which means that all its instances are false. The same holds for argument schemata (resulting from arguments by replacing one or more of their sentences by schemata): such a schema is neither correct nor incorrect; it can be valid (meaning that all its instances are correct) or counter-valid (meaning that all its instances are incorrect).

Note that as we presuppose that there are pre-theoretically correct arguments (namely those the conclusion of which *follows from* their premises), we also presuppose that the forms we create become pre-theoretically valid or not valid. Note also that from this viewpoint it is not the case that arguments are correct *because* of their forms – a form is something that is extracted from the arguments, not the other way around. Of course, we can conjecture a (metaphysical or psychological) theory, according to which the forms exist independently of the arguments (in some Platonist heaven or in our minds) and prior to them; but such a theory would require an underpinning over and above the facts we are considering here, the facts concerning how our natural languages are employed by their speakers.

Parameters, of course, are not the only kind of symbols employed by logicians. Another crucial kind is *constants*. Their role is very different from that of parameters: while parameters are employed to express generality, constants are employed to implement rigor. Words and expressions in natural language are often used in haphazard, fuzzy, and open-ended ways; constants are employed to act as their rigorous surrogates. They are to mimic the important and clear-cut parts of the functioning of their natural language counterparts, while their functioning can be regimented deliberately where the functioning of the natural language counterpart is not clear-cut or does not interest us.

Consider the introduction of implication of Frege (1879, §5):

> If *A* and *B* stand for contents that can become judgments . . . there are the following four possibilities:
> (1) *A* is affirmed and *B* is affirmed;
> (2) *A* is affirmed and *B* is denied;
> (3) *A* is denied and *B* is affirmed;
> (4) *A* is denied and *B* is denied.
> Now
>
> 
>
>
> stands for the judgment that the third of these possibilities does not take place, but one of the three others does. Accordingly, if
>
> 
>
>
> is denied, this means that the third possibility takes place, hence that *A* is denied and *B* affirmed.

The only important issue for Frege was that implication must not lead us from truth to falsity – so he takes his constant to map truth and falsity on falsity. In most other cases, he might have thought, the implication is true, but he was not interested in particularizing this, so he made the constant map *any* other combination of truth values on truth. The result does not quite match the functioning of the natural language *if-then*, but it matches it to the extent to which Frege found it necessary.

Constants are not, by far, as old as parameters, their systematic employment came only in the second half of the nineteenth century.[2]

The relationship between the logical expressions of natural language and artificial logical constants is thus reminiscent of the relation of *explication* as introduced by Carnap (1947, pp. 7–8):[3]

> The task of making more exact a vague or not quite exact concept used in everyday life or in an earlier stage of scientific or logical development or rather of replacing it by a newly constructed, more exact concept, belongs among the most important tasks of logical analysis and logical construction. We call this the task of explicating, or of giving an explication for, the earlier concept; this earlier concept, or sometimes the term used for it, is called the explicandum; and the new concept, or its term, is called an explicatum of the old one.

Individual logical constants, as well as the whole formal language, can be seen precisely as an explication of their natural prototypes (though the fact that there is a feedback from them to the prototypes indicate that the process in play is something more than merely the explication).[4]

It is crucial to fully appreciate the deep difference between the role of parameters and that of constants. We will call the replacement of a natural language expression by a constant, appropriating the term of Quine, *regimentation*. Regimentation thus leads us from a natural expression with its natural meaning to an artificial expression with an artificial – and hence artificially gerrymandered – meaning. We have already started to call the replacement of a natural language expression by a parameter

2. The standard logical constants in the shape in which we know them today, i.e., sentential operators, quantifiers, etc., were not introduced before the end of the nineteenth century, when they were proposed by Frege, Peano, Russell, and others (see Grattan-Guinness, 2000). Some earlier precedents, though, can perhaps be found: the commentators of Aristotle, for example, came to introduce the well-known shortcuts for the syllogistic connections of the terms: *a*, *e*, *i*, *o*. We can see those as short-hands for *Every . . . is . . .* , *Some . . . is . . .* , *Not every . . . is . . .* , *No . . . is . . .* .
3. See also Beaney (2004).
4. See Brun (2016) for an exposition of some intricacies of Carnapian explication.

abstraction; an abstraction thus leads us from a natural expression with its meaning to a symbol which has no independent meaning at all.[5]

It is important to distinguish when a symbol (typically a letter) is used as a parameter and when it is used as a constant. Note, however, that the difference between a constant and a parameter can be understood on two levels. In the first sense, the difference between them is just a matter of the intention of those who introduce the symbols, in particular of whether they want to see them as firmly associated with specified elements of natural language. Thus, the symbol *P*, which we use to replace *dog* when we regiment *Fido is a dog* as *a is P*, or as *P*(*a*), may be an (extralogical) constant if it is intended to be tied specifically to *dog* (and hence then not to be used for the regimentation of *mouse* in *Mickey is a mouse*), while if it is not intended to be tied to any specific expression and it is used as a mere placeholder for any kind of common noun or predicate (and hence can be used for *mouse* as well as for *dog*), it will be a parameter. To emphasize the difference, we will usually use italic letters as parameters and letters printed in bold italics as constants.

The other level of difference between a parameter and a constant is based on the fact that the intention to use something as a constant normally leads to specifying its behavior within arguments in some explicit and exact way. Indeed, this is *the point* of regimentation. Hence, from this viewpoint something is a constant iff there is this kind of specification. On this level, then, the difference between a constant and a parameter is the difference between a term the behavior of which in arguments is "canonized" and that for which this is not the case.

Fixing some arguments, or types or arguments, in an artificial language as correct is to create a theoretical relation that consists of the correct arguments and that is to act as the theoretical counterpart of the pre-theoretical relation of *following-from* – we will call it the relation of *inference*. Hence, the role of an expression within this relation can be dubbed its *inferential role* – thus, the inferential role of an expression amounts to the "contribution" of the expression to the correctness of the arguments in which it occurs.

3.2 Artificial Languages of Logic

While both parameters and constants are well-known from the history of logic (though, as we have noted, the history of the latter is much shorter than that of the former), the idea of using them to construct a self-contained artificial language (or perhaps "language") appeared no earlier

5. This terminology was introduced by Svoboda and Peregrin (2009); and then subsequently developed by the same authors (Peregrin & Svoboda, 2016, 2017).

than in the second half of the nineteenth century. Frege especially realized that instead of local explications of various locutions of natural language, we can have one global explication of the whole language – an artificial language that could serve as a "logical proxy" of the natural one.

Logicians of Frege's ilk started to feel that the two kinds of symbols used to articulate forms of correct arguments could be developed and integrated with each other so that they could form wholly self-contained artificial languages. Frege's (1879) *Begriffsschrift* was probably the first truly self-conscious attempt, but there were others; Boole (1848) before him, Peano (1897) shortly after him, and, of course, Whitehead and Russell (1910) still later.[6] Since then, the attention of logicians has shifted almost completely to the new, artificial languages, as though they were the true vehicle of our reasoning, which constitutes the subject matter of logic. This was backed up by the Russellian doctrine of the logical forms of expressions as being something real, something that is concealed by the surface forms of the expressions but that is the ultimate subject matter of logic. Hence, artificial languages started to look like the means for an entirely transparent representation of the logical forms that are represented – not very transparently – by the means of natural languages.

Viewed thus, an artificial language of logic, which, unlike natural language, represents logical forms in a transparent way, appears to lead us to a "surpassing" of natural languages. And indeed, some logicians (like, for example, Carnap)[7] seemed to assume that the languages of logic provide us with better language forms, and hence potentially better languages than the natural ones. The view of the ur-father of modern logic, Frege, was much more modest. In his *Begriffsschrift*, he compared his artificial language to the microscope (p. 5):

> Because of the range of its possible uses and the versatility with which it can adapt to the most diverse circumstances, the eye is far superior to the microscope. Considered as an optical instrument, to be sure, it exhibits many imperfections, which ordinarily remain unnoticed only on account of its intimate connection with our mental life. But, as soon as scientific goals demand great sharpness of resolution, the eye proves to be insufficient. The microscope, on the other hand, is perfectly suited to precisely such goals, but that is just why it is useless for all others. This ideography [*Begriffsschrift*], likewise, is a device invented for certain scientific purposes, and one must not

6. See Grattan-Guinness (2000) and Haaparanta (2009).
7. The so-called principle of tolerance, put forward by Carnap (1934, pp. 44–45), implies, according to the author, that "everyone is at liberty to build his own logic, i.e. his own form of language, as he wishes", hence natural language has no precedence over anything better that we can engineer.

condemn it because it is not suited to others. If it answers to these purposes in some degree, one should not mind the fact that there are no new truths in my work.

We have already noted that even this modest claim has not quite materialized; not even scientific treatises (nor mathematical ones) have come to be written in a language of logic. Despite this, the theories of modern logic have influenced, in various ways, the languages of science and to some, modest extent perhaps also ordinary argumentation and natural languages.

The relationship between natural language with its pre-theoretical *following-from* and an artificial language of logic, as it has been sketched so far, is depicted in Figure 3.1. The bottom rectangle represents the

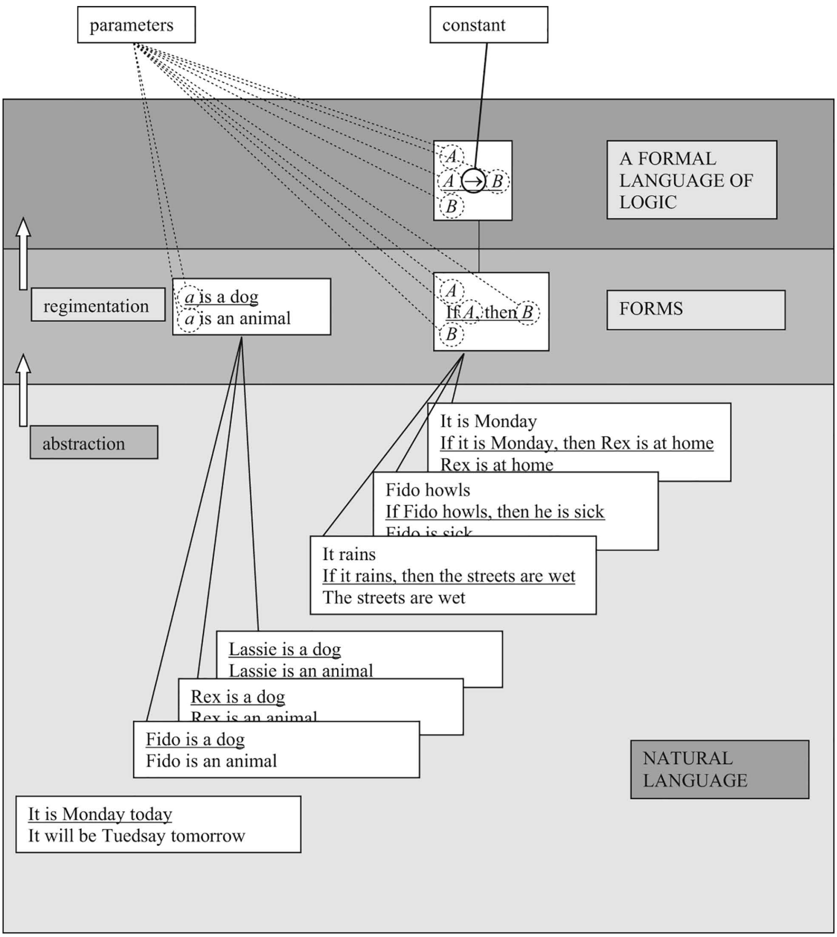

Figure 3.1

natural language with various kinds of correct arguments (instances of *following-from*). If we "abstract away" extralogical expressions (replacing them by parameters), we move to the level of *forms* (represented by the middle rectangle), some of which are *logical* forms and some of which are *valid*. If we further "regiment" logical expressions (namely, replace them by stipulatively produced constants), we reach the upper rectangle representing a formal language of logic.

3.3 Formal and Formalized Languages

The most straightforward way of combining constants and parameters to produce an artificial language useful for logic is replacing logical expressions of natural language by constants and the extralogical ones by parameters. (This, to be sure, presupposes that we have the dividing line between logical and extralogical vocabulary of natural language.)[8] This leads us to a language which we will call *formal* (see Figure 3.1). It is important to realize that the "sentences" of a formal language are schemata, and hence not direct counterparts of the sentences of natural language. A formal language, strictly speaking, is not a language but a *language form*.

Alternatively, we can replace all expressions of natural language by constants. This presupposes that we can also produce constants adequate to extralogical expressions – *extralogical constants*. This is to say that we can define constants whose inferential roles ("meanings")[9] approximate those of natural language extralogical expressions (or those of their part that we are interested in). Especially in the case of empirical expressions, this is not quite so easy. However, it is generally taken for granted that there are ways to do it – within the framework of intensional logic, for example.[10]

We will call a language that results from such a general regimentation, following Tarski (1935), a *formalized language*. Note that the sentences of a formalized language, unlike those of a formal language, *are* on the same level as those of natural language – they *can* be (considered) true or false. Thus, there is a basic difference between formal languages (which are, strictly speaking, not languages but rather language forms) and formalized languages (which are "fully interpreted" languages). See Figure 3.2.[11]

8. See note 8 in Chapter 2.
9. The inferential role is that part of the meaning of an expression that is the direct subject matter of logic. I leave aside the question as to whether it could be seen as *exhausting* the meaning of a constant, which is the claim of so-called *inferentialism* (Peregrin, 2014) and perhaps also of *proof-theoretic semantics* (Francez, 2015).
10. See, especially, Montague (1974) and his followers. See also Peregrin (2006a).
11. Note a difference between Figure 3.1 and Figure 3.2. A formal language results from a regimentation and an abstraction; but the two steps can be carried out in any order. So while Figure 3.1 depicts regimentation as following abstraction, Figure 3.2 has it the other way around.

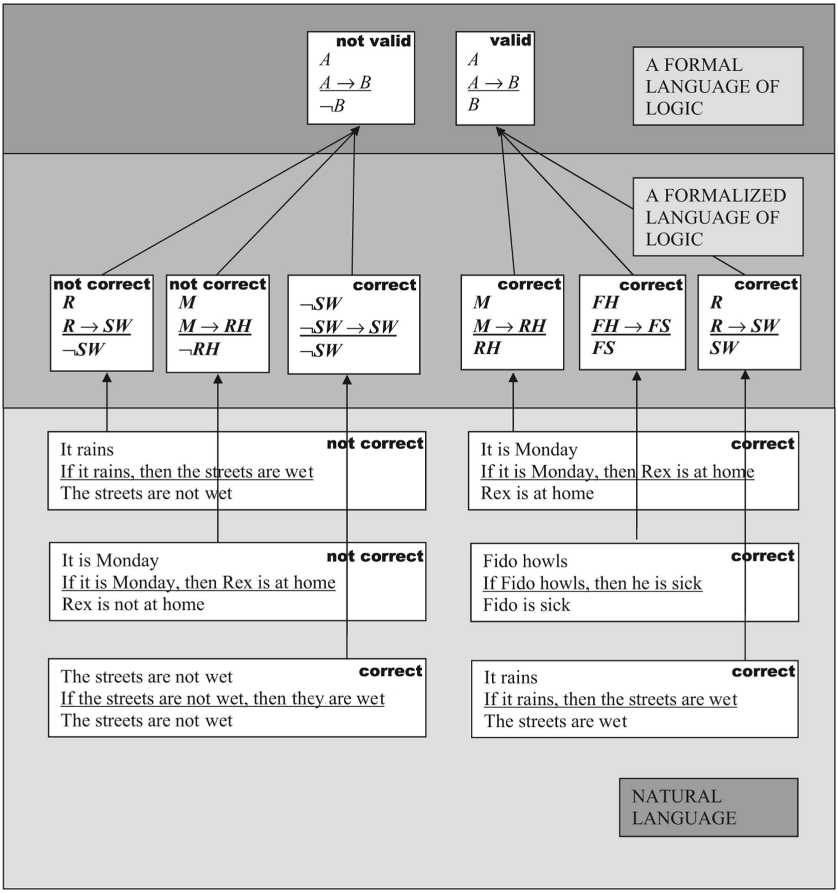

Figure 3.2

It is important to keep in mind that the terms *formal* or *formalized* are often also used in a quite different sense, namely for a language (or something else) which is produced exclusively by our definitions. It would be perhaps possible to call such languages *ideal*; but we will mostly simply speak about *artificial* languages. Both our formal and formalized languages are artificial in this sense; the difference is that a formalized language is an artificial simulacrum of a *language*, while a formal language represents a *language form*.

Thus the distinction between a formal language and a formalized one is quite important; but it is often obscured. It is clear that we can turn a formalized language into a formal one by dropping the "meanings" of extralogical constants; and it seems that we can do something very similar if we do not really *drop* the "meanings" but only "disregard" them. In this sense, it seems that we do not need any formal languages, only formalized ones in which we sometimes change the way we consider extralogical constants. However, there *is* a difference between a symbol with no meaning and a symbol that does have a meaning that we are, however, not interested in; and this difference may be, sometimes, crucial.[12]

Note also that, as we saw, what makes an expression a parameter or a constant, in one sense, is a matter of its intended relationship to something outside the artificial language to which it belongs, and, in another sense, a "superstructure" of the language: a specification of the inferential roles of constants, but not of parameters. It follows that if we deal alone with the language, disregarding its relationship to a natural language, and if we disregard any "superstructure," the distinction between a constant and a parameter becomes void; in such a case we will use the neutral term *word*.

We can imagine a language as a vocabulary, divided into syntactic categories plus syntactic rules, each of which builds an expression of a specified syntactic category out of expressions of specified syntactic categories. (Thus a usual rule of the language of propositional logic builds a sentence out of two sentences and a binary sentential operator.) If $<W_i>_{i\in I}$ is the collection of syntactical categories (where a category is identified with the set of its words) and $<R_j>_{j\in J}$ is the collection of syntactic rules, then we can identify a language with the ordered pair

$$<<W_i>_{i\in I}, <R_j>_{j\in J}>.$$

In fact, of course, language consists not only of words but also of complex expressions coming out of the words by means of rules. If we extend every category by all the complex expressions of the category, we will have a structure which is called a *many-sorted algebra* and which we will denote as

$$<<C_i>_{i\in I}, <R_j>_{j\in J}>.$$

To indicate that the algebra is generated by the words (and the rules), we will also write

$$<[<W_i>_{i\in I}], <R_j>_{j\in J}>.$$

Language in this sense, without any specification of inferential roles (not to mention meanings), will be called a *bare language*.

12. We saw that if I formalize *Fido is a dog* as $P(a)$ so that P and a are constants corresponding to *is a dog* and *Fido*, respectively, I cannot use the same constants to formalize, say, *Mickey is a mouse*. But once I "disregard" the meanings of P and a, it becomes possible to formalize *Mickey is a mouse* as $P(a)$ (of course, if the two formalizations are independent, i.e., if they are not, e.g., parts of a formalization of a single argument). This introduces a treacherous ambiguity of P and a, which, obviously, would be better to avoid by using different kinds of symbols in both cases.

A formal language arises from a bare language if we specify the inferential roles of those of its expressions that we want to treat as constants. A formal language understood in this way may be also called a *logical system*. (Of course, this label makes nontrivial sense only in the case of a formal language that is useful for logical purposes.) We will sometimes speak simply about a *logic* rather than about a *logical system*, as is usual; but as this term is obviously hugely ambiguous, it must be done with a certain care.

We may see the relationship between formal and formalized languages as between kinds and their instances. We have seen that a formal language is a language form, articulating a logical skeleton that may be shared by many different concrete languages; and a formalized language may be an instance of this form, its embodiment resulting, as it were, from adding meat to this skeleton. To get from a formal language to a formalized one we must replace the parameters of the formal language by extralogical constants, that is, to supply (some of) them with inferential roles (and drop the rest). We can therefore get, for example, from the formal language of the first-order predicate logic to the formalized language of Peano arithmetic: while the former contains, aside from logical constants, unlimited sets of parameters, the latter contains, aside from the logical constants, only a few extralogical constants, such as **0** or **+**.

Formalized languages that embody, in this way, the structure spelled out by a formal logical language are sometimes called *theories* within the logic that is articulated by the formal language. Thus, for example, formalized languages that embody the structure of first-order predicate logic are called *theories within first-order logic* or just *first-order theories*.[13] Hence, when we embed Peano arithmetic into the mold of first-order predicate logic, we speak about a first-order theory.[14]

BARE LANGUAGE = vocabulary + syntactic rules

FORMAL LANGUAGE (LOGICAL SYSTEM) = bare language + specification of inferential roles of logical constants

FORMALIZED LANGUAGE = formal language + specification of inferential roles of extralogical constants

Figure 3.3

13. See, e.g., Mendelson (1964).
14. Let us note that the term "theory" is often used in a more abstract sense, for a set of sentences of a language. Theories in the sense of formalized languages (such as Peano arithmetic) can be seen as special cases: they too can be identified with sets of sentences, for the apparatus that is needed to make all the expressions of the language in question into constants also delimits a set of sentences that constitute the theory.

Remember that for a symbol to be a constant it is sometimes enough for it to be intended as a constant, that is, to regiment a specific element of natural language. What makes it into a constant in the *nontrivial* sense is that it has an explicitly fixed inferential role. Thus, while in the case of a formal language we have an apparatus for fixing the inferential roles of a certain part of the vocabulary only (namely of the logical constants), in the case of formalized languages we need the specification of the inferential roles of all expressions of the language. If we specify the roles in terms of a calculus, then moving from a logic to a theory within the logic is extending the calculus to not only account for the inferential roles of logical constants of a formal language but also the extralogical ones of a formalized one.

Let us, to conclude, sketch the standard definition of the simplest commonly used formal language of logic, the language of propositional logic. The vocabulary of this language consists of an unlimited stock of sentential words (A, B, C, A_1, . . . , A_n, . . .), the unary sentential operator $\neg$, and the binary sentential operators $\wedge$, $\vee$, and $\rightarrow$ (or some subset thereof). Then every sentential word is a sentence, and if ϕ and ψ are sentences, then also $\neg\phi$, $(\phi \wedge \psi)$, $(\phi \vee \psi)$, and $(\phi \rightarrow \psi)$ are sentences. (The outermost brackets of a sentence will usually be omitted.) It is this language that underlies the most basic logical calculi. See the summarization in Figure 3.4.

3.4 Structure of the Artificial Languages of Logic*

An artificial language, as understood in this book, is something we create by means of definitions. We assume that an artificial language consists

THE (BARE) LANGUAGE OF STANDARD PROPOSITIONAL LOGIC

Vocabulary:

Sentential words: A, B, C, A_1, …, A_n, …

Unary sentential operator: $\neg$

Binary sentential operators: $\wedge$, $\vee$, $\rightarrow$

Syntactic rules:

A sentential word is a sentence

If O is a unary sentential operator and ϕ a sentence, then $O\phi$ is a sentence

If O is a binary sentential operator and ϕ and ψ are sentences, then $(\phi O \psi)$ is a sentence

Figure 3.4

of a *vocabulary* (its elements, *words*, being usually divided into *syntactic categories*), a set of *syntactic rules*, which determine how to compose complex expressions of various categories out of simpler expressions. A distinctive syntactic category is that of *sentences*; its members may be composed into arguments. The syntactic rules thus determine the set of (well-formed) *expressions* of the language, the vocabulary being its subset. In a typical case, a language of logic is supplemented by a calculus and/or a formal semantics (which we will discuss in detail later). We have already dubbed the mere delimitation of expressions of a language a *bare language*.

I find it useful to articulate this, in a more rigorous way, in algebraic terms. However, I do not want to discourage readers for whom algebra is not a preferred prism and have therefore marked this section (and some more sections in the rest of the book) by an asterisk, indicating that it can be skipped without losing any relevant context.

From an algebraic viewpoint, a language can be seen as what is called a *many-sorted algebra* (Goguen et al., 1977; Janssen, 1986), that is, an ordered pair

$$<<C_i>_{i \in I}, <R_j>_{j \in J}>,$$

where I and J are finite sets of indices, each C_i is a set (the elements of which are called *expressions* of the category C_i; each R_j is a function with an assigned signature $<C_{i_1}, \ldots, C_{i_n}, C_{i_{n+1}}>$ for some $\{i_1, \ldots, i_n, i_{n+1}\} \subseteq I$, which means that R_j is a function from $C_{i_1} \times \ldots \times C_{i_n}$ to $C_{i_{n+1}}$.

Moreover, we assume that there is a collection $<W_i>_{i \in I}$ such that each W_i is finite and $<C_i>_{i \in I}$ is the smallest collection of sets such that it contains $<W_i>_{i \in I}$ (in the sense that C_i contains W_i for every $i \in I$) and is closed w.r.t. all of R_j's. The elements of W_i are then called *words* of the category C_i.[15] We will also denote this algebra as

$$<[<W_i>_{i \in I}], <R_j>_{j \in J}>.$$

Moreover, what is characteristic of language is that one of its sorts or categories is distinguished – the category of *sentences*. Hence, we can say that a language is a finitely-generated many-sorted algebra with a

15. For obvious reasons, we assume that a language must be based on a finite vocabulary. This may seem to conflict with the fact that some formal languages of logic, such as predicate calculus, are usually presented as based on an infinite set of categories, each of which, moreover, contains an infinite number of words. However, such generous definitions are usually meant not as delimiting languages in the strict sense but rather as "reservoirs" out of which any really usable language of the corresponding structure can draw its restricted resources.

distinguished sort.[16] We will, however, often look at a language simply as the set of its sentences.

Expressions of a language are categorized into sorts, but there is also a finer categorization having to do with the way the expressions are formed by means of rules – categorization in terms of the *forms* of the expressions. In particular, any element of the algebra can be assigned what Janssen (1986) calls a "term" and what is the explicit record of the way in which the element results from the generator of the algebra. (If, for example $a = R_1(a_1, \ldots, a_n)$, and $a_n = R_2(b_1, \ldots, b_m)$, where $a_1, \ldots, a_{n-1}$, $b_1, \ldots, b_m$ are among the generators, then a "term" associated with a is "$R_1(a_1, \ldots, a_{n-1}, R_2(b_1, \ldots, b_m))$.") If X is a subset of the generators of the algebra, then an X-form of an element is the "term" associated with it in which all words with the exception of the elements of X are replaced by meaningless symbols (of the kind of our *parameters*). (Continuing our previous example, if $a_1, \ldots, a_{n-1}$ belong to X, but $b_1, \ldots, b_m$ do not, then an X-form of a may be "$R_1(a_1, \ldots, a_{n-1}, R_2(x_1, \ldots, x_m))$.") In what follows, we will usually treat forms in much more abstract ways.

One possibility of how to finalize a bare language into something that can be used by logic is adding a calculus. A calculus stipulates which sentences of the language are inferable from other sentences. It equips certain words of the language, namely constants, with inferential roles. As we assume that some arguments of natural language are correct and others are incorrect, words of natural language also have their inferential roles (though much more fuzzy and indeterminate), and this makes it possible to take the constants with their stipulated inferential roles as the explications of certain words of natural language with their natural inferential roles.

Another possibility is to enrich the bare language with a *formal semantics* – to assign the constants some explications of the meanings of the words to which they correspond. In the next chapter, we will start to investigate the nature of logical calculi; later, we will turn our attention to the systems of formal semantics.

16. This definition could be further narrowed down: we might, for example, require that every expression is part of a sentence (where we can define the relation *is part of* in the obvious way on the basis of the syntactic rules), and/or we may require that the algebra be free, etc.

4 Basic Sentential Operators

4.1 Conjunction

Let us review the most straightforward ways in which we can introduce logical constants. Consider the English connective *and*. It is obvious that if *It rains* and *The sky is cloudy*, it follows that *It rains and the sky is cloudy*, and, conversely, if *It rains and the sky is cloudy*, it follows both that *It rains* and *The sky is cloudy*:

It rains, The sky is cloudy; *hence* It rains and the sky is cloudy
It rains and the sky is cloudy; *hence* It rains
It rains and the sky is cloudy; *hence* The sky is cloudy

Moreover, this seems to be a matter of the connective *and* alone; hence, if we abstract away the concrete sentence, we get the pattern

A, *B*; *hence A* and *B*
A and *B*; *hence A*,
A and *B*; *hence B*,

which is valid in the sense that whichever sentences we substitute for the parameters *A* and *B*, we get conclusions that follow from the premises.

But perhaps this is not without exceptions. Note that, given this pattern, *B and A* follows from *A and B*. (Indeed, *B and A* follows from *B* and *A*, while both *B* and *A* follow from *A and B*.) However, English sentences of the shape *A and B* and *B and A* are not always taken to follow from each other, for *and* is often taken to indicate the time succession, so that *Tom fell down and lost his consciousness* is not taken to follow from *Tom lost his consciousness and fell down* (nor, for that matter, vice versa).

This may lead us to introduce a regimented version of *and*, $\wedge$, which is governed *exclusively* by the pattern

$$A, B \vdash A\wedge B$$
$$A\wedge B \vdash A$$
$$A\wedge B \vdash B$$

and the inferential role that is *exhausted* by this pattern. $\wedge$ is thus a simplified and exactly delimited version of *and*, which in many cases can be used in place of it.

Let us add a few remarks. Although it is commonly accepted that in this way we make $\wedge$ into a fair regimentation of *and*, we saw this does not mean that it captures its behavior in arguments in all details. The fact that $\wedge$ does not capture this behavior of *and* perfectly does not disqualify it as its regimentation. On the contrary, $\wedge$ is part of a *model* of the natural language, and it is in the nature of a model that it simplifies, idealizes, and streamlines. As long as we keep in mind that such a simplification is in play, the simplified regimentation can serve us quite well.

Note also that this very simplification is not inevitable or somehow inherent to the process of regimentation. We may develop a logical constant that would be more faithful to the English *and*. (And indeed such versions of conjunction were proposed, for example, within the framework of dynamic logic.[1]) Working with models is generally a matter of a trade-off between the adequacy with which a model captures what it models and its simplicity. In this particular case, we could also consider whether such a deviation from the standard analysis does not take us beyond the boundaries of logic: perhaps *and* only partly acts as a *logical* word and the aspect of its behavior which concerns time expresses its non-logical component.

Note that out of the three argument forms (rules) that we accepted to specify the inferential role of $\wedge$, one has the constant in the conclusion only, while the other two have it in the premise only. The rules of the first kind are usually called introduction rules ("I"), while those of the second kind are called elimination rules ("E").[2] Hence, we can assign mnemonic labels to the individual rules

($\wedge$I) $A, B \vdash A \wedge B$

($\wedge$E1) $A \wedge B \vdash A$

($\wedge$E2) $A \wedge B \vdash B$.

4.2 The Nature of *Following-From*

When we talked about the pattern governing *and*, we took for granted that *following-from* in English has certain properties. We reasoned that as

A, *B*; *hence A* and *B*.
A and *B*; *hence A*,
A and *B*; *hence B*,

1. See Peregrin (2000b).
2. The terminology derives from Gentzen (1935a).

it is the case that

A and *B*; *hence B* and *A*.

The reason was that when *C* follows from *A* and *B* and both *A* and *B* follow from *D*, then *C* follows from *D*. This can be decomposed into two simple properties of following-from:

If [*X*, *A*, *Y*; *hence B*] and [*Z*; *hence A*], then [*X*, *Z*, *Y*; *hence B*]

and

If [*X*, *A*, *A*, *Y*; *hence B*], then [*X*, *A*, *Y*; *hence B*],

where *X*, *Y*, and *Z* stand for arbitrary sequences of sentences.

Both of these properties seem to be obvious. Are there any others? It seems that another property is that the order of premises is irrelevant, that is,

If [*X*, *A*, *B*, *Y*; *hence C*], then [*X*, *B*, *A*, *Y*; *hence C*].

The last two properties of following-from only spell out the fact that while officially we take the premises as forming a sequence, in fact we treat them as merely forming a set.

Then there is the fact that by adding a surplus premise we cannot distort a case of following-from, that is,

If [*X*, *Y*; *hence A*], then [*X*, *B*, *Y*; *hence A*].

This is a property perhaps slightly more controversial than the previous ones, for one may feel that a genuine premise of an argument is to do a nontrivial job, hence that there is a sense in which, for example, *A and B* does *not* follow from *A*, *B*, and *C* – for it follows from *A* and *B* alone.

The last trivial property of *following-from* is that any sentence follows from itself, hence

A; *hence A*.

Note that, again, it is not so that *following-from* in a language such as English would clearly and unambiguously have these properties; in some cases, the properties may be disputable. This is to say that if we reconstruct following-from using these principles, we gain its reasonable approximation – an approximation that presents a perhaps simplified version of following-from, but a version that can be handled in an exact way.

Let us consider, for the sake of illustration, a very simple formal language, consisting of the sentential words *A*, *B*, *C*, and the single logical constant $\wedge$. This formal language is intended to capture the arguments that are correct thanks to the *and* as regimented by the constant $\wedge$. We stipulate the following pattern

$$A \wedge B \vdash A,$$
$$A \wedge B \vdash B,$$
$$A, B \vdash A \wedge B,$$

and assume that the relation $\vdash$ has the previous properties. It follows that, for example, the following is the case

$$B, A \vdash A \wedge B,$$
$$A, B, C \vdash A \wedge B,$$
$$A, B, C \vdash A \wedge (B \wedge C),$$
etc.

Of course, as A, B, C, . . . are parameters, any one of them can serve to articulate a pattern as well as any other, hence if

$$A, B \vdash A \wedge B,$$

then also

$$A, C \vdash A \wedge C,$$
$$C, B \vdash C \wedge B,$$
etc.

This may be treated as another property of $\vdash$ within our *formal* language (namely, the toy language that contains – aside from the logical constant $\wedge$ – only meaningless parameters). By a systematic replacement of parameters in an instance of $\vdash$, we always get another instance of $\vdash$. (This kind of property, needless to say, does not concern *following-from* in a natural language as this does not contain any meaningless parameters but only meaningful expressions.)

Note that, given these properties of $\vdash$, we can show that we can use the single sentence $A \wedge B$ among the premises of an argument just in case we can use the sequence A, B:

$$X, A, B, Y \vdash C$$

if and only if

$$X, A \wedge B, Y \vdash C.$$

Indeed, if the first of them is the case, then we can use ($\wedge$E1) and ($\wedge$E2) to get

$$X, A\wedge B, A\wedge B, Y \vdash C,$$

which reduces to the second. Conversely, if the second is the case, the first one is yielded directly by ($\wedge$I).

This provides for the possibility of "amalgamating" the premises of arguments: any argument can be brought to a form where it has not more than one premise.

4.3 Implication

After we have touched upon conjunction, let us look at another standard logical operator, implication, which we will denote, as usual, as $\rightarrow$ and which we introduce as a regimentation of the English *if-then*. The most basic characteristic feature of implication is that it obeys the *modus ponens* rule:

$$\text{(MP)}\quad A, A\rightarrow B \vdash B\,.$$

In case the premise $A\rightarrow B$ is itself inferable from the empty sequence of premises, we can leave it out, hence (remember that the sign "//" separates the arguments, which form the premises of a metaargument, from its conclusion)

$$\text{(Ded–)} \vdash A\rightarrow B\ //\ A \vdash B.$$

If we also had the converse of this, namely

$$\text{(Ded+)}\ A \vdash B\ //\ \vdash A\rightarrow B,$$

it would be the case that B would be inferable from A just in case $A\rightarrow B$ were inferable from the empty sequence of premises. Sentences inferable from the empty sequence will be called *theorems*.

This would make it possible to express each correct argument with a single premise as a theorem of the form of implication. Let us call the sentence $A\rightarrow B$ the *internalizer* of the argument $A \vdash B$. If we then have implication governed by (MP) and (Ded+), an argument with a single premise is correct (its conclusion is inferable from its premise) if and only if its internalizer is a theorem.

Let us note that the ability to "internalize arguments" seems to be one of the important capabilities of connectives of the kind of the English *if-then*: it seems to be important that *It rains* being inferable from *It rains* *and it is dark* is tantamount to *If It rains and it is dark then it rains* being (necessarily) true.[3]

Let us also note that the characterization of $\rightarrow$ in terms of (MP) and (Ded+), which may also be rechristened as ($\rightarrow$E) and ($\rightarrow$I), is different from the characterization of $\wedge$ in terms of ($\wedge$E1), ($\wedge$E2), and ($\wedge$I). While all of the latter are argument forms, ($\rightarrow$I) is a transition from an argument form to another one. This is unavoidable insofar as implication is to act as the tool of the internalizing of arguments. This indicates that the picture in which we would characterize logical constants in terms of arguments is oversimplified.

What about the internalization of arguments with more premises,

$$A_1, \ldots, A_n \vdash A?$$

Can we internalize them? If we also have a conjunction and hence can amalgamate premises, then an internalization might be

$$(A_1 \wedge \ldots \wedge A_n) \rightarrow A.$$

Even without conjunction, an internalization would be possible if instead of (Ded+) we had the more general:

$$\text{(Ded+*)}\ X, A \vdash B \,//\, X \vdash A \rightarrow B.$$

In this case, obviously, the general argument would be internalized also by

$$(A_1 \rightarrow \ldots (A_n \rightarrow A) \ldots).$$

Hence, let us call a sentence of this form the *internalizer* of the argument of the corresponding form $A_1, \ldots, A_n \vdash A$. (The internalizer of the argument $\vdash A$ with no premises will be A.)

Aside from letting us internalize arguments, implication also lets us internalize certain metaarguments in the sense that it provides a sentence which is a theorem if and only if the metaargument holds. Consider the metaargument

$$X \vdash A_1; \ldots; X \vdash A_n \,//\, X \vdash A.$$

3. According to the doctrine of *logical expressivism* (Brandom, 2000; Peregrin, 2008; Arazim, 2017), something like this is what constitutes *the point* of having implication in a language.

It is easy to see that it holds if and only if the following argument holds:

$$A_1, \ldots, A_n \vdash A;$$

and hence that its internalizer, namely

$$(A_1 \rightarrow \ldots (A_n \rightarrow A) \ldots),$$

is also an internalizer of the metaargument.

For suppose that the argument holds and instantiate X as $A_1, \ldots, A_n$. We have

$$A_1, \ldots, A_n \vdash A_1; \ldots; A_1, \ldots, A_n \vdash A_n \;//\; A_1, \ldots, A_n \vdash A.$$

Then, as all the premises hold, so does the conclusion.

Suppose, conversely, that $A_1, \ldots, A_n \vdash A$ holds and that $X \vdash A_1, \ldots, X \vdash A_n$. Then, obviously, $X \vdash A$ holds thanks to the properties of inference stated previously; in particular $A_1, \ldots, A_n \vdash A$ and $X \vdash A_1, \ldots, X \vdash A_n$ entail $X, \ldots, X \vdash A$, which in turn entails $X \vdash A$.

In this way, we can express the whole of the relation of inference between the sentences of the language underlying the standard calculi of propositional logic in terms of the sentences of the calculus. This is an important possibility and almost every logical calculus has an implication, which serves as a tool of internalization.

4.4 Negation

We have seen that the rules governing conjunction are quite simple and transparent; and that those governing implication, though they are a bit more complicated, are also relatively transparent. But the situation is more complicated with a further operator that appears to be a *sine qua non* of logic, namely negation.

The trouble with negation, we can say, is the following. While conjunction can be said to directly "extend" following-from (a conjunction of two sentences follows from given premises if and only if both the premises follow from them) and the same can be said, though perhaps in a less direct sense, about implication (an implication follows from given premises if its consequent follows from the premises together with the antecedent), negation seems to be related to cases of *non-following*. It would seem that the negation of a sentence follows from some premises only if the sentence itself does not (unless, of course, the premises are such that everything follows from them).

One way of accounting for this would be to accept that aside from the natural relation of *following-from* that holds among sentences of natural language, there is another relation, something like *counter-following-from*, which we can reconstruct as $\dashv$, such that

$$A_1, \ldots, A_n \dashv A$$

reads as

if $A_1, \ldots, A_n$ are the case, A is not the case.

Then we can characterize the constant $\neg$, which is to regiment negation, so that

$$A_1, \ldots, A_n \vdash \neg A \quad \text{iff} \quad A_1, \ldots, A_n \dashv A.$$

This is a route that some logicians have taken (though most of them relatively recently), and it is usually called *bilateralism*.[4] The trouble with it is that while it is quite plausible that in natural language some sentences (are taken to) follow from other sentences, to say that, aside from this, some sentences "counter-follow" from other sentences (or whichever term we want to use for this relationship) does not seem to be so plausible.[5] (The situation is quite different if we do not merely look at the practices of normal speakers of natural language, but concentrate on the practices within science, especially mathematics – there refutation can be perhaps taken as an enterprise of its own, relatively independent of that of proving.)

Another possible way would be to accept that aside from the relation of *following-from*, there is a pre-theoretical property of *inconsistency* pertaining to sets of sentences. That there is such a pre-theoretical property is perhaps less counterintuitive than the relation of counter-following; it would seem that normal speakers of a natural language should be willing to ascertain that not only do some sentences follow from other sentences, but that also some sentences do not "fit together" – thus a normal speaker of English is likely to accept that *Fido is a dog* and *Fido is a cat*, or *Adam is bigger than Bill*, *Bill is bigger than Charlie*, and *Charlie is bigger that Adam*, are inconsistent. Given this property, it is easy to define the relation of counter-following in its terms. If we reconstruct the former as $\perp$ (so that $\perp A_1, \ldots, A_n$ says that the sequence $A_1, \ldots, A_n$ is inconsistent), then

$$A_1, \ldots, A_n \dashv A \qquad \text{iff} \qquad \perp A_1, \ldots, A_n, A$$

and hence

$$A_1, \ldots, A_n \vdash \neg A \qquad \text{iff} \qquad \perp A_1, \ldots, A_n, A.$$

4. See Restall (2013) or Ripley (2017).

5. The practices of argumentation, which constitute a relatively common "language game," usually proceed via "giving and asking for reasons" (a phrase coming from Brandom, 1994) for a conclusion.

Then there is also a third way, rejecting the recognition of any other pre-theoretical relation or property aside from following-from and trying to further reduce incompatibility to following-from, so that

$$A_1, \ldots, A_n \vdash \neg A \qquad \text{iff} \qquad A_1, \ldots, A_n, A \vdash B \text{ for every } B.$$

This is somewhat reminiscent of the rhetorical figure when, instead of claiming *this cannot be the case*, I claim: *if this is the case, then anything is*. (It is a form of the (in)famous *ex falso quodlibet* argument that assumes that anything follows from false premises, which we will discuss in greater detail later.)

The last option is to consider inconsistency as the only truly primitive concept and try to define following-from in its terms:[6]

$A_1, \ldots, A_n \vdash A$ iff for every $B_1, \ldots, B_m$, if $\perp A, B_1, \ldots, B_m$, then $\perp A_1, \ldots, A_n, B_1, \ldots, B_m$.

Let us, then, admit that there is a relation of inconsistency, be it primitive or reducible to following-from. We can say that negation is basically characterized by the fact that X, A is inconsistent if and only if $X \vdash \neg A$. Thus, we have

($\neg$I*) if X, A is inconsistent, then $X \vdash \neg A$

($\neg$E*) if $X \vdash \neg A$, then X, A is inconsistent.

Now, if we assume that X is inconsistent iff $X \vdash A$ for every A, ($\neg$I*) and ($\neg$E*) entail

($\neg$I) $X, A \vdash B; X, A \vdash \neg B \;//\; X \vdash \neg A$

($\neg$E) $A, \neg A \vdash B.$

To see that ($\neg$E) holds, take the special case of ($\neg$E*) where X is $\neg A$. As $\neg A \vdash \neg A$, we have that A, $\neg A$ is inconsistent, which means that everything is inferable from it.

Seeing that ($\neg$I) holds is slightly more complicated. So assume that X, $A \vdash B$ and X, $A \vdash \neg B$. This yields us, via ($\neg$E), X, $A \vdash C$ for every C, and hence the inconsistency of X, A. Given ($\neg$I*), this yields us $X \vdash \neg A$.

It is also the case that, conversely, ($\neg$E) and ($\neg$I) imply ($\neg$I*) and ($\neg$E*). ($\neg$I*) is obviously a direct consequence of ($\neg$I), whereas ($\neg$E*) follows from ($\neg$E).

4.5 Disjunction

The last common sentential operator is disjunction, $\vee$, regimenting the connectives of the kind of the English *or*. One possibility to introduce it

6. See Peregrin (2015).

is to compose it of conjunction and negation, or implication and negation so that the sentence

$$A \vee B$$

is treated as the shorthand for

$$\neg(\neg A \wedge \neg B)$$

or for

$$\neg A \rightarrow B.$$

An alternative possibility is to define it directly in terms of the following inferential rules:

($\vee$I1) $A \vdash A \vee B$

($\vee$I2) $B \vdash A \vee B$

($\vee$E) $X, A \vdash C; X, B \vdash C \mathbin{//} X, A \vee B \vdash C.$

While the introduction rules ($\vee$I1) and ($\vee$I2) are obvious, ($\vee$E) is a "metarule" and it is perhaps not so obvious. It stipulates that whenever a conclusion follows from each of the disjuncts, it also follows from the disjunction.

Conjunction, implication, negation, and disjunction are the most common sentential operators; we will call them the *core operators of propositional logic*. Putting together all the introduction and elimination rules for the operators we have assembled so far, we have:

($\wedge$I) $A \wedge B \vdash A,$

($\wedge$E1) $A \wedge B \vdash B,$

($\wedge$E2) $A, B \vdash A \wedge B.$

($\vee$I1) $A \vdash A \vee B$

($\vee$I2) $B \vdash A \vee B$

($\vee$E) $X, A \vdash C; X, B \vdash C \mathbin{//} X, A \vee B \vdash C.$

($\rightarrow$I) $X, A \vdash B \mathbin{//} X \vdash A \rightarrow B$

($\rightarrow$E) $A \rightarrow B, A \vdash B$

($\neg$I) $X, A \vdash B; X, A \vdash \neg B \mathbin{//} X \vdash \neg A$

($\neg$E) $A, \neg A \vdash B.$

(Needless to say, instead of the particular parameters we have used, A, B, C, . . . , we could have used any other.)

This assembly will lead us to what we will call a Gentzenian calculus for intuitionistic logic. But before we can move to its rigorous formulation, we must say more about formal languages and calculi.

5 Gentzenian Logical Calculi

5.1 A Gentzenian Calculus

It is now time to take more seriously the fact that in trying to regiment the most basic logical elements of natural language (such as *and* or *if-then*), we create a new, artificial language. So far we have considered the regimented versions of the elements, constants like $\wedge$ or $\rightarrow$, as just our way of investigating the "inferential behavior" of the English words (or, for that matter, analogous words in other natural languages). But it has turned out to be helpful to look at the languages that arise as the by-products of such investigations as kinds of languages in their own right.

The language that we assembled in the course of the previous chapter was already characterized at the end of §3.3: its vocabulary consists of an unlimited stock of sentential parameters (A, B, C, $A_1, \ldots, A_n, \ldots$), plus the logical constants $\neg$, $\wedge$, $\vee$, and $\rightarrow$ (or some subset thereof); and a sentence is either a sentential parameter or an expression of one of the forms $\neg\phi$, $(\phi\wedge\psi)$, $(\phi\vee\psi)$, and $(\phi\rightarrow\psi)$, where ϕ and ψ are sentences. (We usually leave out the outermost brackets of a sentence.)[1] This language "regiments" certain very general structures of English and serves as a basis for the definitions of calculi producing relations of inference, which "regiment" certain parts of the relation of following-from that holds among English sentences.

We have already stated that in modern logic the fact that correctness of arguments is accounted for in terms of calculi that are, however, formulated not directly for the natural language to which the arguments belong, but rather for some artificial language, is ubiquitous. The artificial languages are meant to capture the logical forms of sentences of the natural language – hence serve as *models* of natural language in the sense discussed in §1.2. We try to indicate that the most basic regimentation, carried out with the aim of constituting such calculi, does not force us to

1. As is well-known, the language could also be formulated entirely without the brackets in the so-called Polish notation, due to Łukasiewicz – so these symbols are to be seen as dispensable auxiliaries. See Mendelson (1964, p. 21).

deviate from the natural language in a too non-transparent manner. We have already seen that the regimentations that lead to the common sentential operators are mostly a matter of what can be called "direct regimentation"; that is, replacing a word or a phrase of a natural language by means of an explicitly and rigorously defined constant. Given this, the relationship between the natural language and the emerging artificial language is still rather transparent.

What is crucial is that the model has a certain independence from the phenomenon it models, and, dealing with it, we can forget about some of the perplexities of the real phenomenon (for they are just not present within the model) and can concentrate fully on the investigation of the simpler model. (It is only after we reach some interesting results concerning the model that we return to the perspective from which the model is just our way of studying the phenomenon and try to project the results back on the original phenomenon.)

Hence, if we return to our example of a toy formal language, consisting of the sentential words A, B, C and a single logical constant $\wedge$, we will look at it as not just a means of capturing something from English, but as a kind of language in its own right. We forget that A, B, C were introduced as parameters, which were to "abstract away" extralogical expressions; we take them just as basic building blocks of the new language. We have seen that the basic arguments we take for correct are

$$A, B \vdash A \wedge B,$$
$$A \wedge B \vdash A,$$
$$A \wedge B \vdash B.$$

Also we have seen that the pattern holds not only for A and B, but rather for any pair of sentences of the language. To account for this, we can employ some new kinds of letters, for example, ϕ and ψ (which are not elements of our language, but rather means we will use to talk about it), and write

$$(\wedge \mathrm{I}) \quad \phi, \psi \vdash \phi \wedge \psi,$$
$$(\wedge \mathrm{E1}) \quad \phi \wedge \psi \vdash \phi,$$
$$(\wedge \mathrm{E2}) \quad \phi \wedge \psi \vdash \psi.$$

(If we were not to forget that the symbols A, B, C are already parameters, then ϕ and ψ would appear to us as "*meta*parameters." This is slightly strange, for in this case already the language we are considering is a language of forms that is to capture kinds of arguments in a natural language, and now we need to capture kinds of arguments in this language, that is, "forms of forms." But the reason is that we have (temporarily) assumed a perspective from which we disregard any links between the artificial language and the natural one, and consequently we do *not* see

A, B, C as parameters but rather as merely the basic building blocks of the language, however it is they may be seen from different perspectives.)

We have already indicated that a way to characterize correct arguments would be to make a potential list of their kinds. (We cannot make an actual list, for there are too many of them, but we can present some basic correct arguments and some ways to produce new correct arguments from already given ones). We would try to make the potential list in such a way as to cover as many correct arguments as possible.

Needless to say, such a delimitation of correct arguments yields a kind of "axiomatic system." We have the basic arguments as the "axioms," we have some "rules of derivation" of arguments from arguments, and we have the "theorems," which are all the arguments derivable from the axioms. The only difference between this system and prototypical axiomatic systems (such as the system of Euclid's *Foundations* or the modern axiomatizations of geometry or arithmetic by Hilbert or Peano[2]) is that in the prototypical case the axioms are sentences and the rules take us from sentences to sentences; while in our case the axioms are arguments and the rules take us from arguments to arguments. To avoid misunderstandings that might stem from this, we will talk about *axiomatic arguments* and *theorematic arguments*. The conclusion of a theorematic argument with no premises will be called a *theorem*.

Note that we now have a certain ambiguity of the term *axiomatic argument* (and, for that matter, *theorematic argument* and the terms *axiom* and *theorem* to be discussed later). ($\wedge$E1), for example, is an argument *form*, a form with many instances in the language of propositional logic, instances like

$A \wedge B \vdash A,$

$C \wedge B \vdash C,$

$(A \wedge B) \wedge (B \wedge C) \vdash (A \wedge B),$

etc.

What we can call the axiomatic argument is either the scheme itself or its instances. Either of the options is in some respect problematic. In the first case, axiomatic arguments do not really belong to the language with which we deal (for example, the language of propositional logic to which we are aiming now), and in fact are not really arguments (but rather forms). In the second case, we have an infinity of axiomatic arguments, which seems to contradict the very idea of an axiomatic system that appears to be based on the assumption that the number of axioms is relatively small. Out of these, the less misleading seems to be the second option and so, strictly speaking, axiomatic arguments will be the *instances* of schemata like ($\wedge$E1); the

2. See Mueller (1969).

schemata themselves will be called *axiomatic argument schemata* (and similarly *theorematic argument schemata* and later *axiom schemata* and *theorem schemata*). But we will not always speak so strictly – where no misunderstanding is likely, we will let the term *axiomatic argument* (and the like) fluctuate in between the schema and its instances.

There is also a third possibility, which is to have as axiomatic arguments neither the schemata, nor their instances, but rather one, distinguished, instance of each schema. Hence the axiom, for example, is neither the schema

$$\phi \wedge \psi \vdash \phi,$$

nor all its instances

$$A \wedge B \vdash A,\ A \wedge C \vdash A,\ B \wedge A \vdash B, \ldots,$$

but rather only a single instance, say

$$A \wedge B \vdash A.$$

Then, however, we need to add the rule of substitution, which produces, out of this axiom, all the other instances of the schema as theorems. This also is not a plausible solution. One thing is that the stipulation that $A \wedge B \vdash A$ is an axiom, while $B \wedge A \vdash B$ is not, seems to be all too arbitrary. Another thing is that the very rule of substitution seems to be strangely different from other rules of derivation: while the other rules appear to approximate straightforward and transparent patterns of steps leading us from instances of following-from to other such instances in natural language, this rule appears to amount to something much more involved.

Anyway, if we have some axiomatic arguments, what we need in order to complete a Gentzenian calculus are some rules of derivation, rules inferring arguments from arguments ("metaarguments") – given them, we will be able to produce the theorematic arguments. The theorematic arguments determine what is correctly inferable from what (a conclusion, according to the calculus, is correctly inferable from a premise iff the premises and the conclusion constitute a theorematic argument). Hence, we can say that the axioms plus the rules of derivation also generate a relation of inference: instances of the relation are precisely the theorematic arguments.

5.2 Structural Rules

Which rules of derivation of arguments from arguments should we accept? We have already seen that the correctness of some arguments "brings about" the correctness of others. Thus, given that both

$$A, B \vdash A \wedge B$$

and

$$A\wedge B, C \vdash (A\wedge B)\wedge C,$$

we can compose them into the argument

$$A, B, C \vdash (A\wedge B)\wedge C.$$

The corresponding general rule of composition would then be:

(Cut) $\Gamma \vdash \phi;\ \Delta, \phi, \Sigma \vdash \psi\ //\ \Delta, \Gamma, \Sigma \vdash \psi,$

where the capital Greek letters stand for unspecified (possibly empty) sequences of sentences.

In a similar fashion, we can turn other properties of the relation of following-from that we diagnosed in §4.2 into metaargument forms, that is, rules of derivation of arguments from arguments:

(Ref) $//\ \phi \vdash \phi.$
(Ext) $\Gamma, \Delta \vdash \phi\ //\ \Gamma, \psi, \Delta \vdash \phi$
(Perm) $\Gamma, \phi, \psi, \Delta \vdash \xi\ //\ \Gamma, \psi, \phi, \Delta \vdash \xi$
(Con) $\Gamma, \phi, \phi, \Delta \vdash \psi\ //\ \Gamma, \phi, \Delta \vdash \psi$

The set of (Ref), (Ext), (Perm), (Con), and (Cut) constitute the so-called *structural rules*, discussed by Gentzen (1935a); they are summarized in Figure 5.1.

A calculus which incorporates all of these rules is called *structural*. (It is not necessary that the calculus is *defined* in terms of the rules, it is enough

STRUCTURAL RULES

(Ref) $//\ \phi \vdash \phi$
(Ext) $\Gamma, \Delta \vdash \phi\ //\ \Gamma, \psi, \Delta \vdash \phi$
(Perm) $\Gamma, \phi, \psi, \Delta \vdash \xi\ //\ \Gamma, \psi, \phi, \Delta \vdash \xi$
(Con) $\Gamma, \phi, \phi, \Delta \vdash \psi\ //\ \Gamma, \phi, \Delta \vdash \psi$
(Cut) $\Gamma \vdash \phi;\ \Delta, \phi, \Sigma \vdash \psi\ //\ \Delta, \Gamma, \Sigma \vdash \psi$

Figure 5.1

that it is "closed" to them in the sense that whenever premises of one of the rules are theorematic, the conclusion is also theorematic – in this case, we say that it *admits* the rules.) The structural calculi explicate an intuitive notion of following-from that we saw is quite plausible: composability of arguments, reusability of premises, irrelevance of their order and of "idle" premises – these are all properties that would (almost) seem to go without saying when we think about arguments in the prototypical sense of the word, or if we consider, for example, the common praxis of proving in mathematics.

There may also be, however, alternative explications of following-from, such that, for example, something is reasonably called a premise of a correct argument only if it is *not* "idle," that is, if the argument without it would not go through. Or we can dispose of the rules (Perm) and (Con)

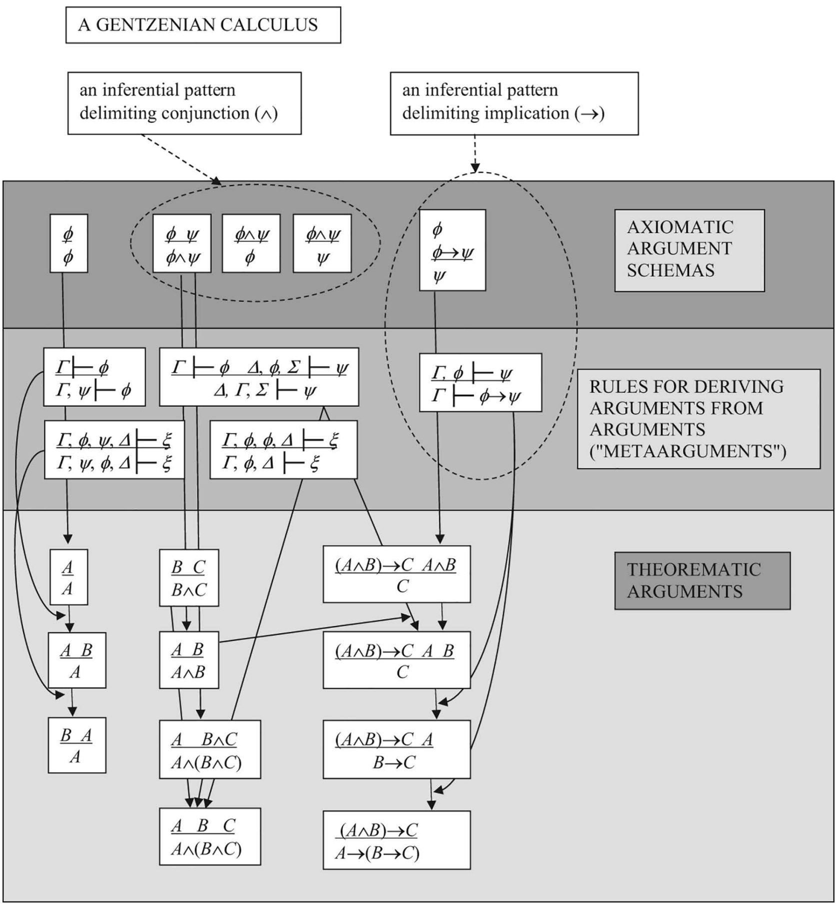

Figure 5.2

if we consider the premises as forming not a sequence but rather a set. Thus, if we were to define an argument not as an ordered pair of a finite *sequence* of sentences and a sentence but of a finite *set* of sentences and a sentence, we would need only three structural rules.

In what follows we will assume that the structural rules hold unless it is stated otherwise.

What has been concluded in the previous chapter regarding the introductory and elimination rules for core logical operators now leads us to the following Gentzenian calculus for intuitionistic propositional logic:

(∧I) $\phi,\psi \vdash \phi\wedge\psi$

(∧E1) $\phi\wedge\psi \vdash \phi$

(∧E2) $\phi\wedge\psi \vdash \psi$

(∨I1) $\phi \vdash \phi\vee\psi$

(∨I2) $\psi \vdash \phi\vee\psi$

(∨E) $\Gamma, \phi \vdash \xi; \Gamma, \psi \vdash \xi \,//\, \Gamma, \phi\vee\psi \vdash \xi$

(→ I) $\Gamma, \phi \vdash \psi \,//\, \Gamma \vdash \phi\rightarrow\psi$

(→ E) $\phi\rightarrow\psi, \phi \vdash \psi$

(¬I) $\Gamma, \phi \vdash \psi; \Gamma, \phi \vdash \neg\psi \,//\, \Gamma \vdash \neg\phi$

(¬E) $\phi,\neg\phi \vdash \psi$

5.3 Single-Conclusion vs. Multiple-Conclusion Arguments

The Gentzenian calculi I have presented up to now are not exactly the kind introduced by Gentzen himself. They can be seen as an intermediary step between what he himself called *calculi of natural deduction* and that which he dubbed *sequent calculi*. The calculi of natural deduction were devised to reconstruct the ways that our reasoning with logical constants actually works[3] and did not really incorporate "metaarguments" as rules for moving from arguments to other arguments. Instead, they contained what looks like moves from arguments to sentences. Thus, the introduction rule for the implication looks like the step from the premise $\phi \vdash \psi$ to the conclusion $\phi\rightarrow\psi$.

This, to be sure, is not the way in which the calculi of natural deduction were usually presented. Instead of being articulated in terms of arguments-to-sentences rules, they were usually presented as calculi for reasoning that is hypothetical in the sense that it proceeds from premises accepted only "for the sake of argument" (which is even much more

3. Gentzen (1935a, p. 176), writes: "First I wished to construct a formalism that comes as close as possible to actual reasoning. Thus arose a 'calculus of natural deduction.' "

obvious when they are presented in the form proposed by Fitch, 1952 than in the original form of Gentzen). The idea is that as we build up an argument by combining elementary arguments we use premises, some of which do not end up as the premises of the resulting argument but get "discarded" – they are rendered as premises of the "subarguments" from which we move to sentences. Thus, for example, we introduce ϕ as a premise and if we manage to reach the conclusion ψ we may "discard" ϕ and move to the conclusion $\phi\to\psi$.

However, what we conclude in such a case is not merely $\phi\to\psi$ but rather $\vdash \phi\to\psi$. Similarly, if the whole argument were to use some premises Γ, the conclusion would be $\Gamma \vdash \phi\to\psi$. Thus, it seems that what is going on is indeed using rules for moving from arguments to arguments.[4]

Gentzen (1935a, b) made this quite explicit within the framework of his *sequent calculus*, which evolved out of his system of natural deduction. In this framework, we move from arguments to arguments quite explicitly; however, there is one more modification, namely that the arguments we work with have multiple conclusions. Thus, we no longer consider arguments of the form

$$\phi_1, \ldots, \phi_n \vdash \phi,$$

but rather generalized arguments, called *sequents*, of the general form

$$\phi_1, \ldots, \phi_n \vdash \psi_1, \ldots, \psi_m.$$

An informal elucidation of this kind of sequent is

> whenever all of $\phi_1, \ldots, \phi_n$ are the case, then at least one of $\psi_1, \ldots, \psi_m$ is the case

or

> if ϕ_1 and . . . and ϕ_n, then ψ_1 or . . . or ψ_m.

Multiple-conclusion sequents bring about certain technical simplifications; but we must extend the structural rules to also cover them. For example, while the (Cut) for the single-conclusion calculus is

$$\Gamma \vdash \phi; \Delta, \phi, \Sigma \vdash \psi \;//\; \Delta, \Gamma, \Sigma \vdash \psi,$$

4. This is also close to how the metarules are construed in the authoritative textbook of Prawitz (1965, p. 23): "A deduction rule can be thought of as a rule that allows us to infer a formula from a whole deduction given as a 'premiss' and to determine what set of formulas the inferred formula depends on."

now we will have

$$\Delta, \phi, \Sigma \vdash \Gamma; \Gamma * \vdash \Delta *, \phi, \Sigma * \,//\, \Delta, \Gamma *, \Sigma \vdash \Delta *, \Gamma, \Sigma *,$$

and similarly for the other rules.

In this context, for example, the rule ($\vee$E) yields us

($\vee$E*) $\phi \vee \psi \vdash \phi, \psi,$

which secures that disjunction cannot obtain without at least one of its disjuncts obtaining. Indeed, as $\phi \vdash \phi, \psi$ and $\psi \vdash \phi, \psi$, ($\vee$E) yields us ($\vee$E*). As conversely ($\vee$E*) entails ($\vee$E), we can then replace ($\vee$E) by the simpler ($\vee$E*). That ($\vee$E*) entails ($\vee$E) is seen in the following way: If we assume that $\phi \vdash \xi$ and $\psi \vdash \xi$, then using (Cut)[5] we get $\phi \vee \psi \vdash \xi, \xi$, and using (Con) (again generalized to the right-hand sides) we have $\phi \vee \psi \vdash \xi$.

It is interesting to note that the situation where disjunction cannot obtain without at least one of its disjuncts obtaining can be brought about, even within the single-conclusion sequents context, by accepting the assumption that the introductory rules for disjunction represent *the only way* in which we can reach the disjunction. Indeed, if this is the case then we know that if a disjunction is inferable from some premises, then it can only be because at least one of the disjuncts is inferable from them, which, in effect, amounts to ($\vee$E*). I think that this provides a way to throw some light on the cryptic and often cited remark of Gentzen (1935a, p. 189), that "the introductions represent, as it were, the 'definitions' of the symbols concerned, and the eliminations are no more, in the final analysis, than the consequences of these definitions."

5.4 The Algebraic Structure of G-Calculi*

We may see a bare language as consisting of three levels, as in Figure 5.3. We have (i) the vocabulary, divided into words belonging to several categories, (ii) the set of expressions produced from the vocabulary by means of syntactic rules, again divided into categories with sentences as one of the categories; and (iii) arguments as conglomerates of sentences. The calculus then delimits a subset of the set of arguments, the set of *theorematic arguments*. A sentence that is the conclusion of theorematic arguments with no premises is a *theorem*.

What must be added to this picture is the fact that we classify arguments in terms of their *forms*. I suggested in a previous chapter that bare language is best reconstructed as a kind of many-sorted algebra, but here

5. Its special case we need here is $\phi \vdash \xi; \zeta \vdash \phi, \psi \,//\, \zeta \vdash \xi, \psi$.

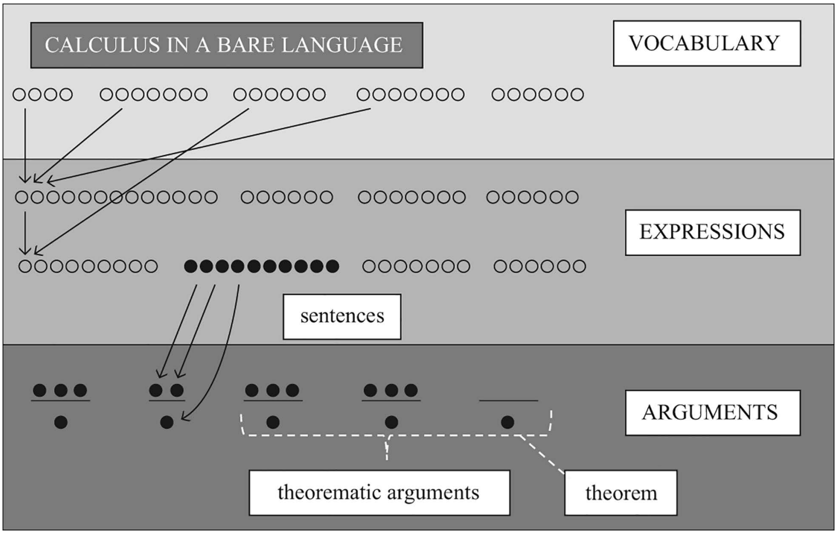

Figure 5.3

we switch to the simpler view. A (*bare*) *language*, from this viewpoint, can be seen simply as a set of sentences each of which has a certain form. A *form* is supposed to be something that yields a sentence via a kind of *substitution*. Thus, we can see a bare language as an ordered triple $<S,F,U>$, where S is a set (the elements of which are called *sentences*), F is a set (the elements of which are called sentence *forms*), and U is a set of functions (called *substitutions*), each of which maps F on S. If there is an $u \in U$ such that $u(f)=s$, then s is said to be an *u-instance* (or simply an *instance*) of f. We assume that the relation is "decidable" in the sense that we are always able to decide whether s is an instance of f.

Consider, for example the language of classical propositional logic with the sentential words A, B, C, . . . and sentential operators $\neg$, $\wedge$, $\vee$, and $\rightarrow$. Thus, we have the sentences A, B, $A \wedge B$, $B \vee C$, etc. By replacing all sentential words within a sentence by the symbols ϕ, ψ, ξ, . . . , which we call (*sentential*) *parameters*, we get a sentence form. We then have the forms

ϕ,

ψ,

$\phi \wedge \psi$,

$(\xi \vee \xi) \wedge \xi)$, etc.

A substitution, in this special case, is any mapping of all the parameters on sentences, which induces a mapping of all forms on sentences. Thus, if we have the substitution mapping ϕ on A, ψ on $A \wedge B$, etc., we have the mapping of $\phi \wedge \psi$ on $A \wedge (A \wedge B)$, etc. If this mapping is u, then $A \wedge (A \wedge B)$ is the u-instance of $\phi \wedge \psi$.

An argument over a language $L = \langle S,F,U \rangle$ is an ordered pair $\langle\langle s_1, \ldots, s_n\rangle, s_{n+1}\rangle$, where $s_1, \ldots, s_n, s_{n+1} \in S$; $s_1, \ldots, s_n$ are the *premises* of the argument and s_{n+1} is its *conclusion*. An argument form is an ordered pair $\langle\langle f_1, \ldots, f_n\rangle, f_{n+1}\rangle$, where $f_1, \ldots, f_n, f_{n+1} \in F$. (Again, $f_1, \ldots, f_n$ are its premises and f_{n+1} is its conclusion.) An argument $\langle\langle s_1, \ldots, s_n\rangle, s_{n+1}\rangle$ is an u-instance (or simply an instance) of an argument form $\langle\langle f_1, \ldots, f_n\rangle, f_{n+1}\rangle$ iff $u(f_k) = s_k$ for $k = 1, \ldots, n+1$. We will often write $s_1, \ldots, s_n \vdash s_{n+1}$ instead of $\langle\langle s_1, \ldots, s_n\rangle, s_{n+1}\rangle$, and write $f_1, \ldots, f_n \vdash f_{n+1}$ instead of $\langle\langle f_1, \ldots, f_n\rangle, f_{n+1}\rangle$.

Examples of arguments over the language of propositional calculus are

$A \vdash A \wedge B$;

$A \wedge B \vdash B$;

$A, B, A \wedge B \vdash C$, etc.

Examples of argument forms are

$\phi \vdash \phi \wedge \phi$;

$\phi \vdash \phi \wedge \psi$;

$\phi \wedge \psi \vdash \phi$, etc.

If u is the substitution mapping ϕ on A, ψ on $A \wedge B$, etc., we have the mapping of $\phi \vdash \phi \wedge \psi$ on $A \vdash A \wedge (A \wedge B)$, etc.

An *inferential relation* over $L = \langle S,F,U \rangle$ relates sequences of elements of S to elements of S. (As the basis of the relation are arguments and as an argument can only have a finite number of premises, it may seem that it only relates *finite* sequences of elements of S to elements of S, but in view of the fact that adding any number of ("idle") premises to an argument need not spoil its correctness (as the structural rule (Ext) stipulates), we will consider the relation as possibly extending to infinite sequences.) If I is an inferential relation and $\langle\langle s_1, \ldots, s_n\rangle, s_{n+1}\rangle \in I$, then we will also write $s_1, \ldots, s_n \vdash_I s_{n+1}$.

A *metaargument* over L is an ordered pair $\langle\langle a_1, \ldots, a_n\rangle, a_{n+1}\rangle$, where $a_1, \ldots, a_n, a_{n+1}$ are arguments over L. A *metaargument protoform* over L is an ordered pair $\langle\langle f_1, \ldots, f_n\rangle, f_{n+1}\rangle$, where $f_1, \ldots, f_n, f_{n+1}$ are argument forms over L. A metaargument $\langle\langle a_1, \ldots, a_n\rangle, a_{n+1}\rangle$ is an *u-instance* (or simply an *instance*) of the metaargument protoform $\langle\langle f_1, \ldots, f_n\rangle, f_{n+1}\rangle$ iff there is a $u \in U$ so that a_k is an u-instance of f_k for $k = 1, \ldots, n+1$. We will often write $a_1; \ldots; a_n \,//\, a_{n+1}$ instead of $\langle\langle a_1, \ldots, a_n\rangle, a_{n+1}\rangle$. Examples of metaarguments are

$A, B \vdash A \wedge B \,//\, B, A \vdash A \wedge B$

$A \vdash B; A \vdash C \,//\, A \vdash B$.

Examples of metaargument protoforms then are

$$\phi, \psi \vdash \phi \wedge \psi \mathbin{//} \psi, \phi \vdash \phi \wedge \psi$$
$$\phi \vdash \psi ; \psi \vdash \xi \mathbin{//} \phi \vdash \xi.$$

The reason why we talk about "protoforms" (rather than forms) is that we want to construe forms in a still more general way. We want to be able to consider arguments with an unequal number of premises as of the same form. For this purpose, we can use specific parameters, such as $\Gamma, \Delta, \Sigma, \ldots$, to stand for sequences of sentence forms. Thus, we will have forms like

$$\Gamma \vdash \phi ; \Delta, \phi, \Sigma \vdash \psi \mathbin{//} \Delta, \Gamma, \Sigma \vdash \psi,$$

the instances of which are

$$\vdash A; C, A \vdash B \mathbin{//} C \vdash B, \text{ or}$$
$$C \vdash A; D, E, A, F \vdash B \mathbin{//} D, E, C, F \vdash B,$$
etc.

To implement this into our framework, we will take metaargument forms to be unspecified objects (just like sentence forms), the only explicit property of which is that they are instantiated by metaargument protoforms.

We thus introduce an (artificial) concept of a proto-calculus as an ordered triple $<L,G,V>$, such that L is a language, G a set (the elements of which are called *metaargument forms*), and V is a set of functions each of which maps G on metaargument protoforms over L. If $v \in V$ such that $v(f) = b$, then b will be called a *v-instance* (or simply an *instance*) of f; if, moreover, $L = <S,F,U>$ and $u \in U$ such that $u(b) = a$, then the metaargument a is said to be an (v, u)-*instance* (or simply an *instance*) of f.

In the case of propositional calculus, we can express metaargument forms by replacing sub-sequences of parameters by the specific kinds of parameters introduced previously, Γ, Δ, Σ:

$$\Gamma, \phi, \psi, \Delta \vdash \phi \wedge \psi \mathbin{//} \Gamma, \psi, \phi, \Delta \vdash \phi \wedge \psi .$$

In this case, if v maps Γ on ξ, ϕ, and Δ on ξ, its v-instance is

$$\xi, \phi, \phi, \psi, \xi \vdash \phi \wedge \psi \mathbin{//} \xi, \phi, \psi, \phi, \xi \vdash \phi \wedge \psi;$$

and if, moreover, u maps ϕ on A, ψ on B and ξ on C, its (v, u)-instance is

$$C, A, A, B, C \vdash A \wedge B \mathbin{//} C, A, B, A, C \vdash A \wedge B.$$

We can now define a Gentzenian calculus in a straightforward way. A *G-calculus* is simply a proto-calculus plus a finite set of metaargument forms (determining a metainferential relation) over it. The inference relation generated by the calculus $C=\langle\langle L,G,V\rangle,M\rangle$, $\vdash_C$, is the smallest set of arguments closed to all instances of the elements of M. (Note that as the premises of some metainferential rules from M may be empty, the inference relation need not be empty.)

We say that an inferential relation I over L *admits* a metaargument form m iff I is closed to all instances of m. If I is generated by a calculus $\langle P,M\rangle$ and $m\in M$, then, trivially, I admits m. A relation is called *structural* if it admits all the structural rules. It is called *substructural* if it is not structural.

6 Hilbertian Logical Calculi

6.1 Hilbertian Calculi

We have seen that Gentzenian calculi are axiomatic systems of a strange sort – their axioms and theorems are not sentences, but rather arguments. The possibility to internalize arguments, however, opens up the possibility of the transformation of the calculi into axiomatic systems in the traditional sense. (In fact, historically it was the other way around, Gentzen proposed his calculi only after the more traditional ones were already put to use by Hilbert and others.)

We saw that the possibility of turning the axiomatic system of correct arguments into something much more similar to the axiomatic system in the traditional sense is made possible by the presence, in a calculus, of an operator $\rightarrow$ governed by

(Ded–) $\vdash \phi \rightarrow \psi \,//\, \phi \vdash \psi$

and

(Ded+*) $\Gamma, \phi \vdash \psi \,//\, \Gamma \vdash \phi \rightarrow \psi$,

namely, implication. If we can present correct arguments in the form of theorems, we can delimit correct arguments using axioms in the form of sentences and rules deriving sentences from sentences. Take conjunction. It would seem that if we have implication, the axiomatic arguments governing it, that is, ($\wedge$I), ($\wedge$E1), and ($\wedge$E2), can be replaced by their internalizations, namely the axioms

($\wedge$I′) $\phi \rightarrow (\psi \rightarrow (\phi \wedge \psi))$,

($\wedge$E1′) $(\phi \wedge \psi) \rightarrow \phi$,

($\wedge$E1′) $(\phi \wedge \psi) \rightarrow \psi$.

A similar thing can be done with disjunction, which is also defined in terms of a metaargument:

$$(\vee\mathrm{E})\quad \Gamma, \phi \vdash \xi;\ \Gamma, \psi \vdash \xi \ //\ \Gamma, \phi\vee\psi \vdash \xi,$$

which, in the presence of implication can be reduced to

$$\vdash \phi\to\xi;\ \vdash \psi\to\xi \ //\ \vdash (\phi\vee\psi)\to\xi,$$

We have seen that it is possible to internalize metaarguments of this form, the internalization being

$$(\vee\mathrm{E}')\quad (\phi\to\xi)\to((\psi\to\xi)\to((\phi\vee\psi)\to\xi)).$$

Also negation, defined in terms of the metaargument:

$$(\neg\mathrm{I})\quad \Gamma, \phi \vdash \psi;\ \Gamma, \phi \vdash \neg\psi \ //\ \Gamma \vdash \neg\phi$$

which is equivalent to

$$\Gamma \vdash \phi\to\psi;\ \Gamma \vdash \phi\to\neg\psi \ //\ \Gamma \vdash \neg\phi$$

can be internalized as

$$(\neg\mathrm{I}')\ (\phi\to\psi)\to((\phi\to\neg\psi)\to\neg\phi)$$

However, we must keep in mind that this works only insofar as implication works as the means of internalization, which is conditioned by the fact that (Ded-) and (Ded+*) are admissible (that is, whenever their premise is theorematic, the conclusion is bound to be theorematic too). It follows that we cannot replace (MP) by its internalization (for it is only (MP) in its non-internalized form that yields us (Ded-)). Moreover, we must somehow guarantee the admissibility of (Ded+*): for (Ded+*) is not even an argument (but rather a metaargument), so unlike (MP) it cannot simply be adopted as our rule of derivation.

A calculus in which the axioms are sentences and the rules derive sentences from sentences is called *Hilbertian*; we will speak about *H-calculi*.[1] And while G-calculi derived from the intention of making deduction natural in the sense that they tried to articulate the most basic rules of inference governing logical operators, H-calculi are perhaps "natural" in a very different sense. They continue the ("natural") tradition of "axiomatic systems" going back to Antiquity.

1. See, e.g., Mendelson (1964) or Shoenfield (1967).

The calculi are called so because of David Hilbert, who had already worked with them before Gentzen introduced his.[2] These calculi do not directly aim at delimiting correct arguments (as the theorematic ones); they rather aim at delimiting logical (or, more broadly, mathematical, or whatever) truths. They can be seen as a response to the challenge of condensing a whole domain of knowledge (such as logic or arithmetic) into a few principles, from which all other truths of the domain can be recovered by a few rules of derivation. This indicates that while the most basic virtue of G-calculi is supposed to be their transparency (the important point is not how many axioms there are, but how natural and transparent they are), that of the H-calculi is *austerity* (the fewer axioms that we have, and the shorter they are, the better).

How, then, can we transform a given Gentzenian calculus into a Hilbertian one? As soon as we have an implication that works as the means of internalization, we can transform rules for all other operators into axioms. Hence, what we need is the kind of implication that obeys (Ded-) and (Ded+*). (Ded-) is not a problem: it is enough to accept (MP) as a rule of derivation. How can we achieve the admissibility of (Ded+*)?

It turns out that this is also not a big problem. In the case of intuitionistic propositional calculus, if we have the axioms

$$(\rightarrow 1) \quad \phi\rightarrow(\psi\rightarrow\phi)$$
$$(\rightarrow 2) \quad (\phi\rightarrow(\psi\rightarrow\xi))\rightarrow((\phi\rightarrow\psi)\rightarrow(\phi\rightarrow\xi))$$

plus (MP) as the only rule of derivation, then (Ded+*) is admissible. More generally, consider an argument scheme of the form

$$\Theta_1, \ldots, \Theta_n \vdash \Theta.$$

Here, $\Theta_1, \ldots, \Theta_n, \Theta$ are "metaparameters" standing for sentence schemes so that, for example, (MP), taken as an argument scheme, is its instance (for we may take n as 2 and instantiate Θ_1 as ψ, Θ_2 as $\psi\rightarrow\xi$, and Θ as ξ). Now take a parameter which occurs in none of $\Theta_1, \ldots, \Theta_n, \Theta$, say ϕ, and form the axiom scheme

$$(\phi\rightarrow\Theta_1)\rightarrow(\ldots((\phi\rightarrow\Theta_n)\rightarrow(\phi\rightarrow\Theta))\ldots).$$

This sentence scheme will be called a *relativization* of the argument scheme $\Theta_1, \ldots, \Theta_n \vdash \Theta$. (Thus, ($\rightarrow$2) is a relativization of (MP).) Then we can say that if a H-calculus contains relativizations of all its rules of derivation, then (Ded+*) is admissible in it. (See §6.3 for a proof.) Thus,

2. See Hilbert and Ackermann (1928).

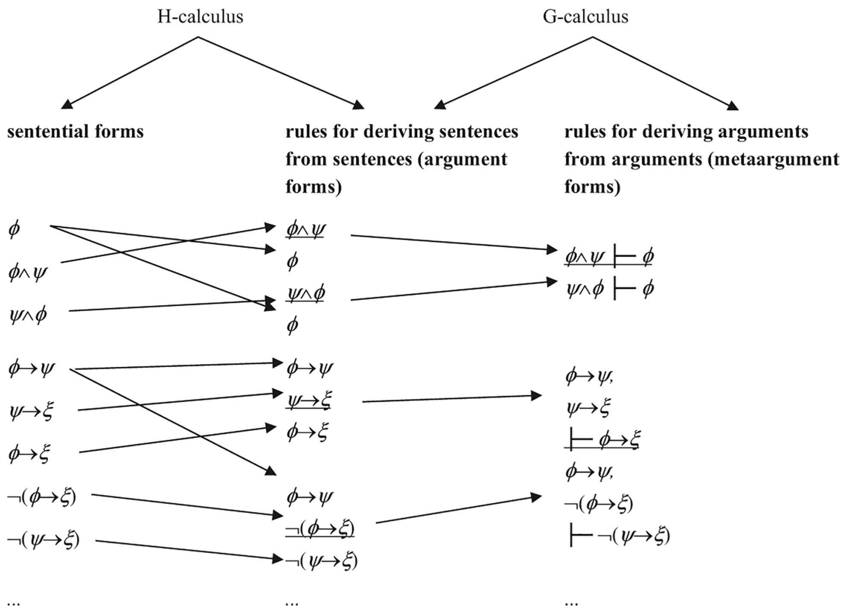

Figure 6.1

given that (MP) is the only rule of derivation, we guarantee this by adding the scheme (→2).

The difference between G-calculi and H-calculi can be depicted by means of the diagram in Figure 6.1. There are three kinds of items in play: sentences, arguments (moves from sentences to sentences), and "metaarguments" (moves from arguments to arguments). Correspondingly, we have three kinds of forms: sentence forms, argument forms (which may be taken as rules for deriving sentences from sentences), and metaargument forms (which may be taken as rules for deriving arguments from arguments). H-calculi trade in the items of the first two levels (and not directly in the third one): their axioms are sentences and their rules of derivations are the rules for inferring sentences from sentences. In contrast to this, G-calculi deal with the items of the second and third level: their axioms are arguments and their rules of derivations are the rules for inferring arguments from arguments.

6.2 Correct Arguments in Hilbertian Calculi

A Hilbertian calculus consists of sentence schemes, the instances of which are axioms and rules of derivations of sentences from sentences (argument schemes). The sentences derivable from the axioms in terms of the rules are the *theorems* of the calculus. But the calculus does not delimit the relation of inference, and hence does not delimit the arguments that are theorematic ("correct") according to it as directly as the Gentzenian one. This must be done via a detour through the concept of *proof*.

A proof of a conclusion (sentence) from a finite list of premises (sentences) is a sequence of sentences which ends with the conclusion and which has the property that every sentence in the sequence is either an axiom, or a premise, or is derivable from some sentences occurring earlier in the sequence by a rule of derivation. Thus the proof shows, as one would expect, that if we accept the premises (and we accept the calculus), we cannot but also accept the conclusion. Hence, it shows that the conclusion, according to the calculus, is inferable from the premises.

What is remarkable is that the arguments, which are delimited in this way by a calculus (that is, the conclusion of which is provable from their premises), are precisely those that are derivable from the axioms and the rules of derivation of the calculus by means of the structural rules. This means that if we take the axioms (as premiseless arguments), plus the rules of derivation (as arguments) as the axioms of a G-calculus, and take the structural rules as its rules of derivation, the theorematic arguments of the calculus will be precisely those for which there is a proof of their conclusion from their premises in the original H-calculus (see the next section for more details). In this sense, the concept of proof in the previous sense "embodies" the structural rules.

We have seen that the Gentzenian introduction and elimination rules for conjunction can be straightforwardly turned into axioms of a Hilbertian system. The same holds for disjunction and for negation:

(∨I1′) $\phi\to(\phi\vee\psi)$

(∨I2′) $\psi\to(\phi\vee\psi)$

(∨E′) $(\phi\to\xi)\to((\psi\to\xi)\to((\phi\vee\psi)\to\xi))$

(¬E′) $\phi\to(\neg\phi\to\psi)$

(¬I′) $(\phi\to\psi)\to((\phi\to\neg\psi)\to\neg\phi)$.

The situation, of course, is slightly more complicated for implication, where we cannot simply internalize the rules (→I) and (→E) (which would both yield us the trivial $(\phi\to\psi)\to(\phi\to\psi)$), for we need implication to keep acting as the internalizer. Therefore, we must keep (MP) (or (→E)) in its original shape:

(MP) $\phi\to\psi, \phi \vdash \psi$.

And we have seen that what we must add are the following two axioms

$(\to 1)$ $\phi \to (\psi \to \phi)$
$(\to 2)$ $(\phi \to (\psi \to \xi)) \to ((\phi \to \psi) \to (\phi \to \xi))$.

This completes a Hilbertian calculus for intuitionistic logic:

$(\wedge \mathrm{I}')$ $\phi \to (\psi \to (\phi \wedge \psi))$
$(\wedge \mathrm{E1}')$ $(\phi \wedge \psi) \to \phi$
$(\wedge \mathrm{E2}')$ $(\phi \wedge \psi) \to \psi$
$(\to 1)$ $\phi \to (\psi \to \phi)$
$(\to 2)$ $(\phi \to (\psi \to \xi)) \to ((\phi \to \psi) \to (\phi \to \xi))$
$(\vee \mathrm{I1}')$ $\phi \to (\phi \vee \psi)$
$(\vee \mathrm{I2}')$ $\psi \to (\phi \vee \psi)$
$(\vee \mathrm{E}')$ $(\phi \to \xi) \to ((\psi \to \xi) \to ((\phi \vee \psi) \to \xi))$
$(\neg \mathrm{E}')$ $\phi \to (\neg \phi \to \psi)$
$(\neg \mathrm{I}')$ $(\phi \to \psi) \to ((\phi \to \neg \psi) \to \neg \phi)$
(MP) $\phi \to \psi, \phi \vdash \psi$.

We can see that a Gentzenian calculus and a Hilbertian one can be two sides of the same coin; in particular, a calculus for intuitionistic propositional logic can be formulated both as a Gentzenian and as a Hilbertian one.

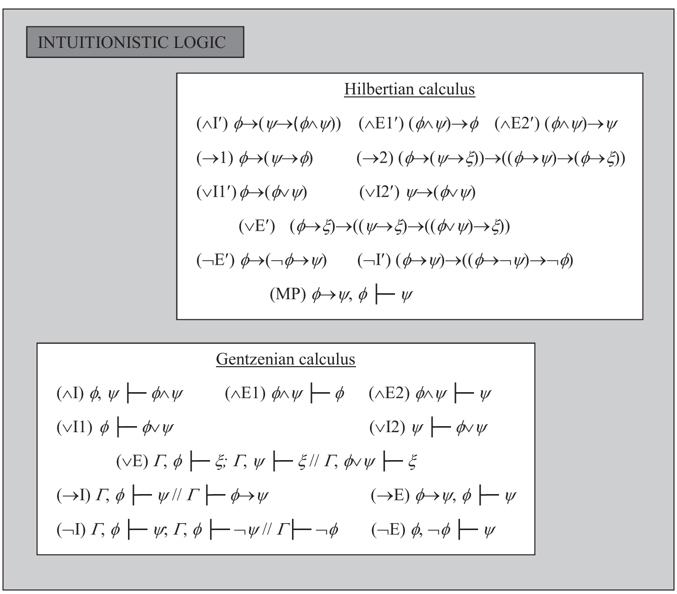

Figure 6.2

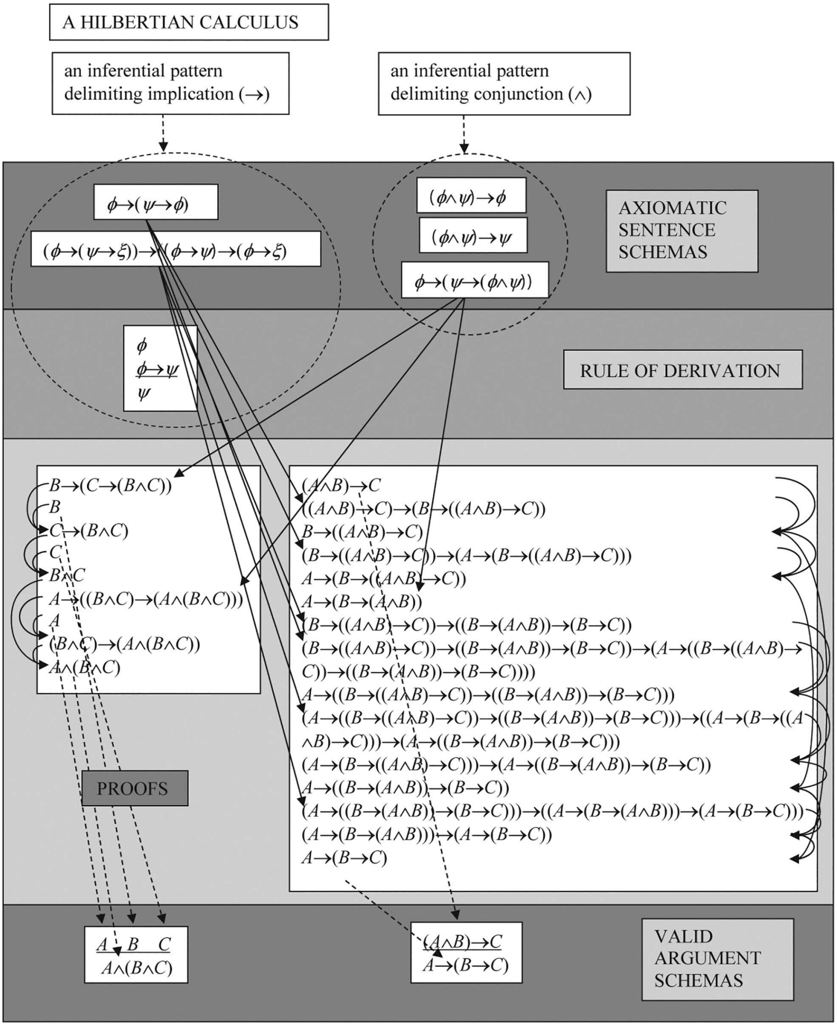

Figure 6.3

6.3 The Algebraic Structure of H-Calculi*

An *H-calculus* consists of a language, a set of sentence forms, and a set of argument forms (that is, rules for deriving sentences from sentences) over it. Let C=<L,S,A> be an H-calculus; and let us define the inference relation that it generates. Hence, let ϕ be a sentence of L and Γ a sequence of sentences of L. A *C-proof of the conclusion ϕ from the premises Γ* is a sequence of sentences such that: (i) every element of the sequence is either an element of Γ, or is an instance of an element of S, or is a conclusion of an instance of an element of A, all the premises of which occur earlier in the sequence, (ii) ϕ is the last element of the list. *ϕ is C-provable from Γ* iff there exists its C-proof from Γ: we will write $\Gamma \Vdash_C \phi$. A sentence ϕ such that $\Vdash_C \phi$ is called a *C-theorem*.

An interesting fact, noted previously, is that the relation of provability coincides with the closure of S and A with respect to the structural rules in the following sense:

Theorem. Let $C=<L,S,A>$ be an H-calculus; let A^* be the smallest set that (1) contains A, (2) contains, for every element of S, an argument form with the empty sequence of premises and with the element of S as the conclusion, (3) is closed under the structural rules. Then $\Gamma \Vdash_C \phi$ iff $\Gamma \vdash_{A^*} \phi$.

Proof: Let first $\Gamma \vdash_{A^*} \phi$. Let us prove, by induction, that $\Gamma \Vdash_C \phi$. If $\Gamma \vdash_A \phi$ (that is, $<\Gamma,\phi>\in A$), then it is trivially the case that $\Gamma \Vdash_C \phi$. Now let us assume that the theorem holds for all premises of a structural rule and let us prove that it holds for its conclusion. This is obvious for (Ext), (Perm), and (Con): obviously, if there is a proof of ϕ from $\phi_1, \ldots, \phi_i, \phi_{i+1}, \ldots, \phi_n$, then there is, *eo ipso*, its proof from $\phi_1, \ldots, \phi_{i+1}, \phi_i, \ldots, \phi_n$; and similarly for the other two cases. Thus, consider (Cut). Suppose there is a proof P of ϕ from $\phi_1, \ldots, \phi_{i-1}, \phi_i, \phi_{i+1}, \ldots, \phi_n$ and a proof P' of ϕ_i from $\psi_1, \ldots, \psi_m$. Then the proof of ϕ from $\phi_1, \ldots, \phi_{i-1}, \psi_1, \ldots, \psi_m, \phi_{i+1}, \ldots, \phi_n$ can be constructed simply by replacing ϕ in P by P'.

Let now $\Gamma \Vdash_C \phi$, that is, let there be a proof P of ϕ from Γ. We will prove that $\Gamma \vdash_{A^*} \phi$ by induction on $|P|$, the length of P. Let $|P|=1$. Then either $\phi\in\Gamma$, and then $\Gamma \vdash_{A^*} \phi$, thanks to (Ref) and (Ext); or ϕ is an axiom, and then $\Gamma \vdash_{A^*} \phi$, thanks to (Ext). Now suppose that the theorem holds for all proofs of the lengths smaller or equal to n, and let us suppose that $|P|=n+1$. If $\phi\in\Gamma$ or ϕ is an axiom, the theorem holds trivially, so we may assume that there is a Δ such that $\Delta \vdash_A \phi$ and $\Delta^* \subseteq P^*$ (where X^* is the set of all the elements constituting the sequence X). Hence, for every ψ from Δ there is a proof of the length not more than n, so it is the case that $\Gamma \vdash_{A^*} \psi$, and $\Gamma \vdash_{A^*} \phi$ follows by (Cut).

The next theorem we are going to prove concerns the admissibility of (Ded+*), which we saw is needed for implication to act as a vehicle of internalization.

Theorem. Let $C=<L,S,A>$ be a Hilbertian calculus. Let A contain (MP) and ($\rightarrow$1) and let it contain a relativization of every element of A (that is, it contains an axiom scheme of the form $(\phi\rightarrow\Theta_1)\rightarrow(\ldots ((\phi\rightarrow\Theta_n)\rightarrow(\phi\rightarrow\Theta)) \ldots)$ for every $\Theta_1, \ldots, \Theta_n \vdash \Theta$ from A; especially let it contain ($\rightarrow$2) or another relativization of (MP)). Then (Ded+*) is admissible, that is, $\Gamma \Vdash_C \phi\rightarrow\psi$ for every Γ, ϕ and ψ such that $\Gamma, \phi \Vdash_C \psi$.

Proof. Assume $\Gamma, \phi \Vdash_C \psi$; then there is a proof P of ψ from Γ, ϕ. We will prove by induction that for all elements ξ of P it holds that $\Gamma \Vdash_C \phi\rightarrow\xi$; and that hence $\Gamma \Vdash_C \phi\rightarrow\psi$. Note that $(\rightarrow 1)$ and (MP) guarantee that if $\Gamma \Vdash_C \xi$, then $\Gamma \Vdash_C \zeta\rightarrow\xi$ for any ζ. Let ξ be the first element of P. If ξ is an axiom or $\xi\in\Gamma$, then $\Gamma \Vdash_C \xi$ and hence $\Gamma \Vdash_C \phi\rightarrow\xi$, according to $(\rightarrow 1)$. If ξ is ϕ, then $\Gamma \Vdash_C \phi\rightarrow\xi$ as $\vdash \phi\rightarrow\phi$ follows from $(\rightarrow 1)$, $(\rightarrow 2)$, and (MP). Now assume that the theorem holds for all elements of P up to nth and that ξ is a $(n+1)$th element. Then if ξ is an axiom, or an element of Γ, or ϕ, the reasoning is as before. Hence, the only possibility left is that A contains a rule $\Theta_1, \ldots, \Theta_n \vdash \xi$ such that all of $\Theta_1, \ldots, \Theta_n$ occur earlier in P. But then, according to the inductive assumption, $\Gamma \Vdash_C \phi\rightarrow\Theta_i$ for $i=1, \ldots, n$ and as A contains $\vdash (\phi\rightarrow\Theta_1)\rightarrow(\ldots ((\phi\rightarrow \Theta_n)\rightarrow(\phi\rightarrow\xi)) \ldots)$, it follows that $\Gamma \Vdash_C \phi\rightarrow\xi$.

Corollary. Let $(\rightarrow 1)$ and $(\rightarrow 2)$ be among the axioms of a Hilbertian calculus $C=\langle L,S,A\rangle$, and let (MP) be its only element of A. Then (Ded+*) is admissible, that is, $\Gamma \Vdash_C \phi\rightarrow\psi$ for every Γ, ϕ and ψ such that $\Gamma, \phi \Vdash_C \psi$.

7 Properties of Calculi

7.1 Material Soundness and Completeness

Let us now return to the problem of the relationship between a calculus in an artificial language and following-from in natural language; in particular to the question as to what the findings concerning the former tell us about the latter.

A calculus of a formal language is intended to capture the logically correct arguments in a natural language of which the formal language is a model; the relation of *inference* generated by the calculus is to reconstruct the pre-theoretical relation of *following-from* or some of its relevant part. (Possibly in *any* natural language, insofar as the language can be considered as a model of *any* language.) This, of course, works only if the model is adequate, at least to a reasonable extent. An absolute adequacy would mean that the theorematic arguments of the calculus pick out all and only logically correct arguments (that is, those that have valid logical forms) – which is to say that all and only arguments that have the form of one of the theorematic argument forms of the calculus are logically correct.

In fact, this would be too strong a requirement. One of the most basic reasons why we study natural language via logical models is that the models draw sharp boundaries where there are none in natural language, smoothing out irregularities and filling in gaps. We can expect that there will be many arguments in natural language about the correctness of which there will be no consensus; in their cases, the model will have to decide about their correctness stipulatively (though, certainly, not quite willfully). Moreover, cases of arguments that look logically correct but do not turn out as instances of theorems – and also perhaps some cases of arguments that are not logically true but do turn out as such instances – are acceptable.

To consider the adequacy in greater detail, we must presuppose that we know which natural language arguments count as instances of argument forms from our formal language. Let us call every assignment of sentences of natural language to sentences of the formal language (which induces an assignment of arguments to argument forms) an *interpretation* of the language. Because later we will consider different concepts of interpretation, we speak about *material* interpretation here and we will write *interpretation*$_M$.

We assume that some of the interpretations$_M$ are *acceptable*. (Thus, for example, if we interpreted A as *London is in England* and B as *Snow is white* then it would be acceptable to interpret $A \wedge B$ as *London is in England and snow is white*, but not as *Every human is mortal*.) A natural language sentence which an acceptable interpretation$_M$ assigns to a sentence of the artificial one will be called its instance$_M$; and the same for arguments. Now we can say that an argument scheme is *valid*$_M$ iff all its instances$_M$ are correct arguments; and we can say that a calculus is *sound*$_M$ if all its theorematic arguments are valid$_M$, whereas it is *complete*$_M$ iff all the valid$_M$ argument schemes are theorematic.

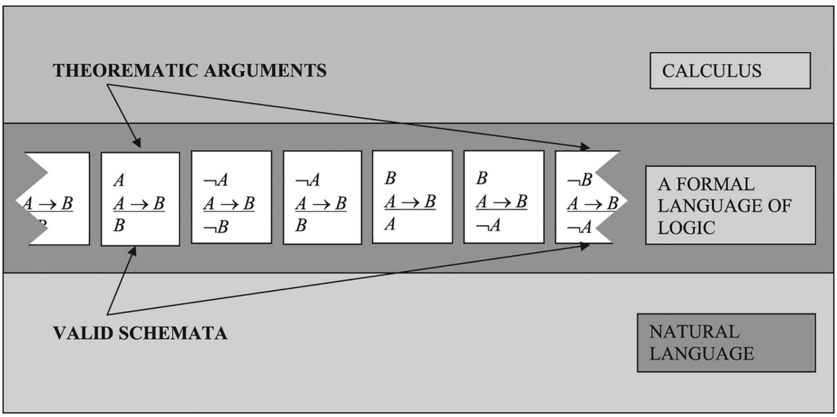

Figure 7.1

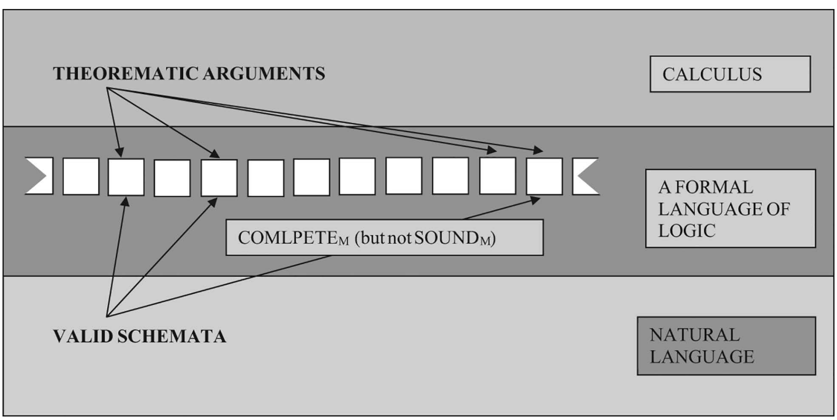

Figure 7.2

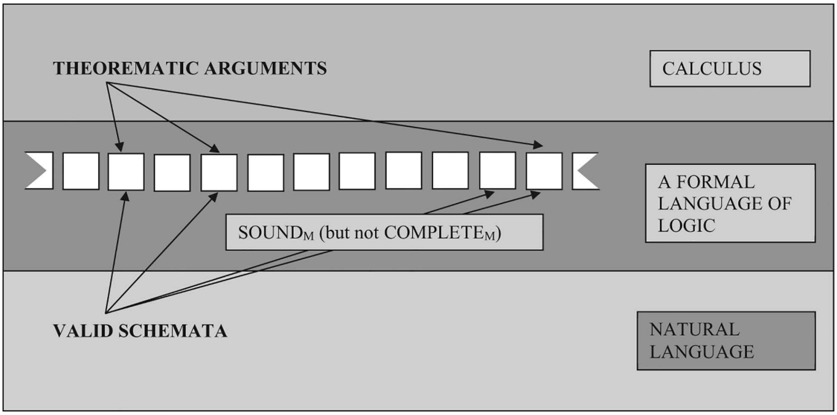

Figure 7.3

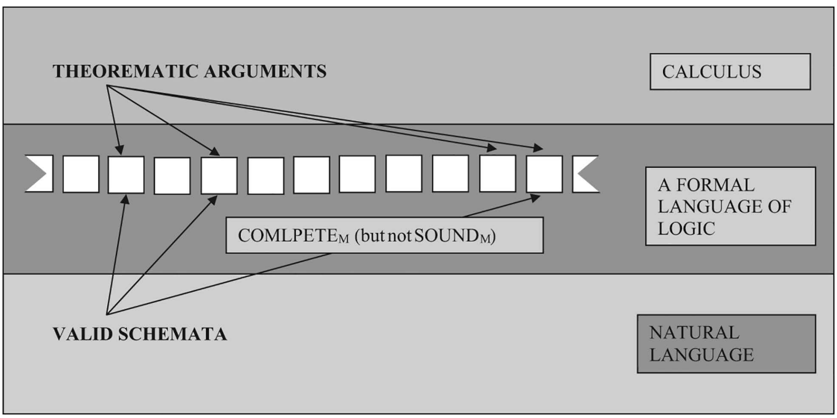

Figure 7.4

Let us now consider a formalized, rather than a formal, language. In this case, there should be only one acceptable interpretation$_M$. (But, of course, this would hold only modulo "saying the same" – it is clear that there will always be multiple, nominally different sentences to interpret$_M$ a given sentence of a formalized language.)[1] Thus, to every argument in

1. A sentence 1+1=2 of the language of Peano arithmetic can be surely interpreted$_M$ as, e.g., *One plus one is two* as well as *Adding one to one gives us two*; but we do not consider this to be two different interpretations$_M$. Let us note that assuming such a sharply delimited relation of synonymy is no minor presupposition and that, hence, sometimes the dividing between a formal and a formalized language may be blurrier than it would seem from our presentation.

the formalized language there corresponds a single argument in the natural one, and the language is sound$_M$ iff all counterparts of theorematic arguments are correct, while it is complete$_M$ iff all correct arguments have theorematic counterparts.

As we already indicated, soundness$_M$ and completness$_M$ are idealized desiderata that are, however, not quite viable. Soundness$_M$ is closer to what can be practically achieved, but we can tolerate a sporadic reclassification of natural language arguments. A limited number of those that are not taken as correct may be reclassified as correct, and especially they can be reclassified as being of a different form than they *prima facie* seem. Completeness$_M$, on the other hand, can be much more easily given up: we may want to capture only some kinds of arguments, disregarding other ones and hence disregarding the existence of some logically correct arguments that are instances of no theorematic forms.

Note also that soundness$_M$ and completeness$_M$ of a calculus depend on the set of acceptable interpretations$_M$, that is, on which natural language sentences count as instances of which sentences of the artificial language. And though we are certainly not free to tamper with the interpretations in any way we choose (remember that the relation between a form and its instances should be "decidable" in the sense that we should be able to "see" whether something is an instance of a form), there are possibilities of tweaking it in various ways.

The most important thing to realize is that soundness$_M$ and completeness$_M$ are not "mathematical" properties of the artificial language; they are the matter of that which interconnects the language, as a mathematical structure, with an empirical phenomenon – that is, a natural language. Therefore, we cannot hope to have a *mathematical proof* of them any more than we can hope that we can have a mathematical proof that a given model in physics is adequate to the reality it models.

7.2 Decidability

A calculus provides for an "exact reconstruction" of the natural (language) relation of *following-from* in an artificial language. (And it should help us decide, via the acceptable interpretations$_M$ of the language of the calculus, what follows from what in natural language.) That the relation of inference is exactly delimited in this way, however, need not mean that if we are to decide whether a given sentence is inferable from some given sentences, we are always capable of doing it. The calculus for which we are able to do it is called *decidable*.

A calculus delimits the relation of inference in that it potentially generates all the theorematic arguments of which it consists. If we are confronted with the task of deciding whether a given sentence of the corresponding artificial language is inferable from given sentences, we may start generating the theorematic arguments, and if we are lucky, we reach the case in point and we know that it is inferable. However, what if we are not so lucky and do not reach it? Then, obviously, there are two possibilities: either the argument is *not* theorematic, or it is, but we have not reached it *yet*. And, in fact, until we do reach it both of these possibilities remain open and we do not know which of them is the actual one.

Note that if we could, besides being able to generate all the theorematic arguments, also generate all those that are not theorematic, a decision procedure would be forthcoming. In such a case, the procedure could be set up so that we would start to generate, in parallel, both the theorematic and the non-theorematic arguments, and the argument in question would be bound to appear, sooner or later, on one of the lists. Hence, a calculus can fail to be decidable only if there is no calculus generating the non-theorematic arguments.

Note, moreover, that even if a decision procedure exists, there is no guarantee that we will always be able to reach the decision. The fact that the decision will be reached "sooner or later" does not suffice. We would need to know that it will be reached in some reasonable time – a procedure that would deliver us the verdict after, say, a million years is obviously not much better than one that would not deliver it at all. Hence, if what we are after is a "humanly feasible" decision procedure, we would need a more fine-grained classification of the procedures according to some upper limits for how long the decision could take. Such classifications are developed within theories of computational complexity.[2]

Anyway, generating the correct arguments does not necessarily lead us to the decision whether a given argument is correct. Hence, the existence of a procedure that would allow us to reach this decision in every case is a property depending on the concrete nature of a calculus. We know that such a procedure exists for classical propositional calculus but not already for the classical predicate one.

7.3 Deductive Completeness

We have introduced the concepts of a valid$_M$ argument form; that of a counter-valid$_M$ form can be introduced accordingly. Now, turning our attention to Hilbertian calculi, we define the concepts of valid$_M$ and counter-valid$_M$ *sentence* forms: a sentence form is valid$_M$ iff all its instances are true, and it is counter-valid$_M$ iff all its instances are false. A *formal* language can be expected to contain sentences that are valid$_M$ and also sentences that are counter-valid$_M$ (negations of the valid$_M$ ones), but it can also be expected to contain sentences that are neither. Thus, we should certainly not require that every sentence is either a theorem or a counter-theorem (a negation of a theorem).

2. See, e.g., Du and Ko (2014).

The situation, however, may be different in the case of *formalized* languages. If we take a language such as that of Peano arithmetic (PA), we may, it would seem, assume that each sentence is either true or false (and that it is such permanently). (Not all kinds of formalized languages must be of this kind: there may be formalized languages containing sentences that are neither true nor false; and there may be such that contain sentences the truth values of which change depending on some empirical circumstances.) In such a case, it makes sense to require what we will call the *deductive completeness* of the calculus: namely, that every sentence is either a theorem or a counter-theorem.

However, Gödel (1931) showed that in the case of some formalized languages, in particular the language of PA, deductive completeness is not attainable. (This disclosed the crucial limitations of logical calculi.) What Gödel showed was that the language of PA contains a sentence, $\boldsymbol{G}$, such that both its probability and its refutability lead to a contradiction. In view of this, $\boldsymbol{G}$ can be neither provable nor refutable, neither $\vdash \boldsymbol{G}$ nor $\vdash \neg\boldsymbol{G}$.

Does this proof tell us something more general than what holds about the specific calculus of arithmetic? Yes, indeed it does. The thing is that, as we will see, in complex logical calculi (beginning with second-order predicate calculus) we can "build" arithmetic out of merely the purely logical parts of the calculus and hence the incompleteness of arithmetic projects into the incompleteness of the logical calculi. We will deal with this in greater detail later.

8 Calculi of Propositional Logic

8.1 Propositional Languages

In previous chapters, we dealt with a basic, formal language of propositional logic; on top of this bare language, we can then, as was discussed in the previous chapters, define the Gentzenian and Hilbertian calculi. The bare language, we saw, can serve as the basis for a calculus of intuitionistic logic, and it can also serve for many other common calculi (especially, as we will see, the calculus of classical logic). Some other calculi of propositional logic require extended languages, which arise from this one by the addition of further logical operators. Thus, for example the calculi of modal logic may require, in addition, the unary sentential operator $\Box$.

Thus, in general, a propositional language is characterized by the fact that aside from the category of sentences it only has categories of sentential operators (sometimes also called *connectives*). Each category of sentential operators is distinguished by an "arity" n, and there is always a syntactic rule mapping an n-ary operator plus n sentences on a sentence.

Thus, algebraically, a propositional language is the many-sorted algebra

$$<[<S, O_{n_1}, \ldots, O_{n_m}>], <R_{n_1}, \ldots, R_{n_m}>>,$$

where $n_1, \ldots, n_m$, are different natural numbers, all sorts are nonempty, and R_{n_i} maps an element of O_{n_i} plus n_i elements of S on S (where S is the category of sentences).

As for the vocabulary of a propositional language, the sort of sentences is supposed to contain an unlimited supply of words. Our basic propositional language, then, contains (logical) constants of only two categories of sentential operators, those of the arities 1 and 2. The former contains the only symbol $\neg$, while the latter contains one to three of the logical operators $\wedge$, $\vee$, and $\rightarrow$. Hence, its general form is

$$<[<\{A, B, C, \ldots\},\ \{\neg\},\ \{\wedge, \vee, \rightarrow\}>], <R_1, R_2>>\,.$$

It is also possible to see this language as a simple, one-sorted algebra. In this case, it is only the sentences that constitute the carrier of the algebra, whereas the operators are "embedded within" the rules. (Thus, the rule $R_\wedge$ takes A and B to the conjunction $A\wedge B$.) The algebra then is

$$<[\{A, B, C, \dots\}], < R_\neg, R_\wedge, R_\vee, R_\to>> .$$

8.2 Intuitionistic vs. Classical Logic

There might seem to be arguments formalized by the language of propositional logic that we feel are correct but are not theorematic in intuitionistic logic; and there may be arguments with a connective that cannot be captured by the sentential operators of intuitionistic logic.

We might, for example, feel that the (premiseless) argument

$$\frac{}{A\vee\neg A}$$

should be theorematic, for such arguments as

$$\frac{}{\text{Fido is sick or Fido is not sick}}$$

are correct; or we may feel that we need some more sentential operators to articulate the forms of, say, arguments like

$$\frac{\text{It is necessary that birds fly}}{\text{It is possible that birds fly}}$$

which we may again take for correct. This may make us think about *alternatives* to, or *modifications* of, intuitionistic logic.

The rules we used to govern the logical constants that led us to intuitionistic logic seemed natural, but they were not the only available options. Which other alternatives are available? There do not seem to be any obvious alternative rules for conjunction or disjunction; and also our rules for implication were quite well motivated, at least insofar as we wanted to have an implication as the "internalizer" of inference. What about negation?

We derived the rules for negation from the idea that negation should be something as a means of reduction of inconsistency to inference. We wanted to reduce the claim that a set of sentences is inconsistent to a claim about inference, which we accomplished by introducing negation so that Γ, ϕ is inconsistent iff it is the case that $\neg\phi$ is inferable from Γ.

Now consider that instead of the stipulation that Γ, ϕ is inconsistent iff $\Gamma \vdash \neg\phi$, we stipulate that $\Gamma, \neg\phi$ is inconsistent iff $\Gamma \vdash \phi$. This might seem to be almost the same, but it is not. If we accept the alternative stipulation, we will still have

(¬E) $\phi, \neg\phi \vdash \psi$,

but instead of

(¬I) $\Gamma, \phi \vdash \psi; \Gamma, \phi \vdash \neg\psi \,//\, \Gamma \vdash \neg\phi$

we will have

(¬I′) $\Gamma, \neg\phi \vdash \psi; \Gamma, \neg\phi \vdash \neg\psi \,//\, \Gamma \vdash \phi$.

If in (¬I′) we instantiate Γ as $\neg\neg\phi$, we get

(¬¬) $\neg\neg\phi \vdash \phi$.

Thus, assuming $\Gamma, \phi \vdash \psi$ and $\Gamma, \phi \vdash \neg\psi$, we have (via (¬¬) and (Cut)) $\Gamma, \neg\neg\phi \vdash \psi$ and $\Gamma, \neg\neg\phi \vdash \neg\psi$, and (¬I′) yields us $\Gamma \vdash \neg\phi$. Thus, (¬I′) entails (¬I).

We do not get (¬¬) from (¬I), which indicates that (¬I′) leads us to a stronger logic than (¬I). And indeed it is no longer intuitionistic logic, but rather *classical* logic. Besides (¬¬), its theorems also include, over and above those included by intuitionistic logic, the *excluded middle*:

(EM) $\phi \vee \neg\phi$.

Is there any reason to favor, as we did, (¬I) over (¬I′) (or, for that matter, the other way around)? Well, (¬I) is closer to an introduction rule than (¬I′), though it is not a perfect introduction rule: it not only contains negation in the conclusion but also in one of the premises. But it at least does contain it, unlike (¬I′), in the conclusion. Hence, if we want to submit to the discipline of the introduction/elimination rules, we head for intuitionistic logic.

Despite this, we can see that the difference between the two logics is delicate; that general considerations can lead us to one of them as well as to the other. Similarly, we can see other kinds of propositional logics as alternatives to the intuitionistic calculus: relevant logics, linear logics, etc., can be seen as calculi working with the same bare language as the classical one but putting forward more or less different relations of inference and sets of theorems.

Thus, classical and intuitionistic logic may be seen as differing in their views on the validity of certain arguments – the intuitionistic logic rejects some of those that the classical one accepts. There are more

CLASSICAL PROPOSITIONAL LOGIC

Hilbertian calculus

($\wedge$I′) $\phi\to(\psi\to(\phi\wedge\psi))$ ($\wedge$E1′) $(\phi\wedge\psi)\to\phi$ ($\wedge$E2′) $(\phi\wedge\psi)\to\psi$

($\vee$I1′) $\phi\to(\phi\vee\psi)$ ($\vee$I2′) $\psi\to(\phi\vee\psi)$

($\vee$E′) $(\phi\to\xi)\to((\psi\to\xi)\to(\phi\vee\psi\to\xi))$

($\to$1) $\phi\to(\psi\to\phi)$ ($\to$2) $(\phi\to(\psi\to\xi))\to((\phi\to\psi)\to(\phi\to\xi))$

($\neg$E′) $\phi\to(\neg\phi\to\psi)$ ($\neg$I′) $(\neg\phi\to\psi)\to((\neg\phi\to\neg\psi)\to\phi)$

(MP) $\phi\to\psi, \phi \vdash \psi$

Gentzenian calculus

($\wedge$I) $\phi, \psi \vdash \phi\wedge\psi$ ($\wedge$E1) $\phi\wedge\psi \vdash \phi$ ($\wedge$E2) $\phi\wedge\psi \vdash \psi$

($\vee$I1) $\phi \vdash \phi\vee\psi$ ($\vee$I2) $\psi \vdash \phi\vee\psi$

($\vee$E) $\Gamma, \phi \vdash \xi; \Gamma, \psi \vdash \xi // \Gamma, \phi\vee\psi \vdash \xi$

($\to$I) $\Gamma, \phi \vdash \psi // \Gamma \vdash \phi\to\psi$ ($\to$E) $\phi\to\psi, \phi \vdash \psi$

($\neg$I) $\Gamma, \neg\phi \vdash \psi; \Gamma, \neg\phi \vdash \neg\psi // \Gamma \vdash \phi$ ($\neg$E) $\phi, \neg\phi \vdash \psi$

Figure 8.1

alternative calculi, which, similar to intuitionistic logic, reject some theorems of classical logic. Thus, relevant logic[1] rejects the *ex falso quodlibet* argument:

(EFQ) $\phi, \neg\phi \vdash \psi$.

The reason is obvious: arguments of this kind allow us to infer a conclusion that has nothing to do with the premises, while relevant logicians maintain that an argument is correct only if the premises are relevant for the conclusion. A side effect of rejecting (EFQ) is that a local contradiction in a calculus (that is, the provability of a sentence together with its negation) does not necessarily entail a global one (namely the provability of everything). The point is that in classical logic, once you can prove a sentence as well as its negation, you can prove all sentences of the language (this is precisely what is facilitated by (EFQ)); in particular, everything is a theorem once you have two

1. See Anderson et al. (1975, 1992).

contradictory theorems. This is no longer the case if (EFQ) is not admissible. Then the derivability of a local contradiction does not directly lead to the global contradiction and thus to the trivialization of the whole calculus.

Like intuitionistic logic, relevant logic can be seen as an attempt at a "correction" of the operators of classical logic, at devising versions of the operators that are in some sense more appropriate. (Note, however, that the appropriateness might be measured using very diverse – and often quite vague – criteria, like closeness to the logical vocabulary of natural language or usefulness for building the foundations of mathematics). In any case, we have various alternative versions of the calculi of propositional logic.

Let us note that rejection of (EFQ) is also one of the points of departure of the so-called *paraconsistent* logic, the proponents of which maintain that contradiction is not something logic should completely shun.[2] As we fallible humans do not – and cannot – always avoid contradictory beliefs, so paraconsistent logicians argue, logic should reflect this and provide us with the calculi that can tolerate local contradictions. In this way, even the traditionally most cherished law of logic, the law of (non-) contradiction (which, in classical logic is expressed as $\neg(\phi\wedge\neg\phi)$[3]), gets rejected by some of the logics of this kind.

RELEVANCE LOGIC R

Hilbertian calculus

$\phi\rightarrow\phi \qquad (\phi\rightarrow\psi)\rightarrow((\psi\rightarrow\xi)\rightarrow(\phi\rightarrow\xi)) \qquad (\phi\rightarrow(\psi\rightarrow\xi))\rightarrow(\psi\rightarrow(\phi\rightarrow\xi))$

$(\phi\rightarrow(\phi\rightarrow\psi))\rightarrow(\phi\rightarrow\psi) \qquad \phi\rightarrow(\phi\vee\psi) \qquad \psi\rightarrow(\phi\vee\psi)$

$(\phi\wedge\psi)\rightarrow\phi \qquad (\phi\wedge\psi)\rightarrow\psi \qquad (\phi\wedge(\psi\vee\xi))\rightarrow((\phi\wedge\psi)\vee(\phi\wedge\xi))$

$((\phi\rightarrow\psi)\wedge(\phi\rightarrow\xi))\rightarrow(\phi\rightarrow(\psi\wedge\xi)) \qquad ((\phi\rightarrow\xi)\wedge(\psi\rightarrow\xi))\rightarrow((\phi\vee\psi)\rightarrow\xi)$

$\neg\neg\phi\rightarrow\phi \qquad (\phi\rightarrow\neg\psi)\rightarrow(\psi\rightarrow\neg\phi)$

$\phi,\ \phi\rightarrow\psi \vdash \psi \qquad \phi,\ \psi \vdash \phi\wedge\psi$

Figure 8.2

2. See Priest (1979) or Béziau and Carnelli (2006).
3. Surprisingly, this, in classical logic, is equivalent to (EM), $\phi\vee\neg\phi$. This is because both laws are *constitutive* of classical logic in the sense that they hold in it automatically. Thus, the formulations of the laws *within* logic say the same in the sense that they both say nothing.

8.3 Alternative or New Operators?

We saw that languages of propositional logic may contain different variants of the core operators (that is, conjunction, implication, negation, and disjunction), and they may also contain more operators. Thus, if we take classical logic as the reference point (as is usual), we may want to distinguish between the *modifications* of its operators and the *additions* of new ones. However, as we will see, the situation is not this clear-cut.

First, however, let us mention the simplest possibility of extending a logic: adding new operators that, however, can be composed out of the existing ones; hence, sentences containing them may be seen as merely notational variants of some sentences without them.

Thus, once we have implication and conjunction, we can define *equivalence* in terms of them

$$\phi \leftrightarrow \psi \equiv_{\text{Def.}} (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi).$$

Similarly, we can produce the operator ↓, known as Peirce's arrow, out of negation and conjunction:

$$\phi \downarrow \psi \equiv_{\text{Def.}} \neg\phi \wedge \neg\psi.$$

We should also note that in some logics even some of the core operators can be composed of other ones. Thus, in classical logic we need only negation and one more operator; the rest can be defined in their terms. Therefore, for example, the disjunction, $\phi \vee \psi$ can be composed out of conjunction and negation, or implication and negation so that the sentence

$$\phi \wedge \psi$$

is treated as shorthand for

$$\neg(\neg\phi \wedge \neg\psi)$$

or for

$$\neg\phi \rightarrow \psi.$$

This does not hold for intuitionistic logic. The different inferential roles of classical and intuitionistic operators are documented by the fact that while, as we already saw, it is the case in classical logic that

$$\vdash \phi \vee \neg\phi,$$

this is not the case in intuitionistic logic. The difference stems from the fact that while in both classical and intuitionistic logic it holds that

$$\phi \vdash \phi\vee\neg\phi \quad \text{and} \quad \neg\phi \vdash \phi\vee\neg\phi,$$

due to the differences between the inferential rules governing negation in the calculi, the following metarule holds only in classical logic

$$\phi \vdash \psi;\neg\phi \vdash \psi \,//\, \vdash \psi.$$

Now let us turn our attention to the operators that either nontrivially modify, or nontrivially extend, the core operators of classical logic.

We have already seen that the operators of intuitionistic logic represent *alternatives* to those of the classical one. Thus, for example, intuitionistic negation differs from the classical one in that it does not obey the double negation rule:

$$\neg\neg\phi \vdash \phi.$$

It seems then that we have a clear example of an operator that is a variant of the classical one, not a new operator. But the situation is not so simple – why could we not see intuitionistic negation as a new operator that has nothing to do with the classical one? And as Quine in fact pointed out, if logical operators are individuated by their inferential roles, then it simply *is* a different operator, though it might seem in some respects *similar* to the classical negation.

This opens up the question as to whether we can see any two logics as rivals at all. Is it not so, as Quine suggested, that once we change a rule for an operator we have a new operator and no reason to see it as an alternative or an upgrade on the original one?[4] The answer is that until we look at the operators as nothing more than vehicles of their respective calculi, the possibilities of seeing two of them as alternatives or as competing are severely limited.

One possibility to do so might seem to be to call two operators as alternatives if they are "similar enough," where "similar" here cannot mean anything other than "governed by similar inferential rules." This, however, would presuppose the development of a measure of similarity of inferential roles; this is something that surely is not inconceivable, but which I have never seen carried out in a satisfactory way.[5]

Another possibility would be to proclaim some specific rule or rules as distinguished and to see all operators governed by the distinguished rule or rules (while differing in other rules that govern them) as "variants." Imagine, for example, that we say that any operator that is governed by

4. This was famously pointed out by Quine's (1986, p. 80) slogan "change of logic, change of subject."

5. But see Jönsson (2017) for an interesting attempt.

MP is an implication. Then we can have a whole lot of implications and we can compare them with respect to their usefulness.

Both of these possibilities, however, have a significant air of arbitrariness. On the basis of what do we draw a sharp boundary between inferential roles that are similar and those that are not? Or why should we say that all operators governed by a rule are variants of the same operator? (Note, for example, that conjunction does obey MP!)

But there is a wholly different view on the matter, which lets us make better sense of the idea of competing variants of logical operators. We must realize that calculi do not originate out of the blue, that they are basically tools of investigating real arguments – namely arguments in natural languages. From this viewpoint, it is important that the constants of the artificial languages that underlie our calculi approximate words or expressions of natural language. And indeed, this is usually the case: conjunction is usually considered to be a more or less straightforward proxy for the English *and*, disjunction for *or*, etc. Now, insofar as there is this relationship of the elements of the artificial calculus to something outside the calculus, there is a natural way of seeing different operators as competing alternatives; they are competing alternatives if they aim at regimenting the same element of natural language.

Take implication. The frequent objection to the implication of classical logic (also called *material* implication) is that English conditionals do not behave in an analogous way. The so-called *strict* implication proposed by C. I. Lewis (1912, 1917), for example, was born out of the author's dissatisfaction with how material implication captures what we would in natural language call *implication*, especially with the paradoxes of implication. From this viewpoint, it makes very good sense to compare material and strict (or, for that matter, intuitionistic or relevant) implication.

8.4 More Operators

We have seen that the distinction between trying to upgrade, say, an implication of a logical calculus and trying to introduce a wholly new one is blurry. However, there are definitely lots of sentential operators that have little to do with the core ones and which were proposed to enhance the scope of logic. The paradigmatic case is represented by the modal ones, which also appeared relatively early in the development of modern logical calculi.

The concepts of necessity and possibility, and their logical behavior, were already investigated by Aristotle as an integral part of the agenda of logic, so it is no wonder that the attempts at their incorporation into the logical calculi appeared as early as the first decades of the twentieth century. Ironically, there is a sense in which they did not result from a conscious attempt at widening the range of logical operators incorporated in the calculi but rather from C. I. Lewis's attempt to introduce his better version of implication mentioned previously. At some point of his musing over the most appropriate axioms for the strict implication, Lewis found out that he cannot make do without the concept of possibility and the corresponding operator $\Diamond$. The five variants of the calculi of strict implication he formulated[6] then gave the impetus for the development of modern modal logic, which then came to be based on the operator $\Box$ of necessity. Possibility and strict implication have then come to be defined in its terms:

$$\Diamond\phi \equiv_{\text{Def.}} \neg\Box\neg\phi$$
$$\phi \mathrel{-\!\!3} \psi \equiv_{\text{Def.}} \Box(\phi\rightarrow\psi).$$

It is remarkable how unclear the axioms for $\Box$ (and other modal operators) were. True, we saw that even in cases of the core operators there were some quarrels among different authors, but here we see that even the proponent, C. I. Lewis, proposed five different calculi, for he was not able to decide which is the "right" one. Since then, calculi for $\Box$ have multiplied to uncountability.

Of course, there are some calculi that appear more natural than others. The simplest of Lewis's five calculi, S5, is produced when we add to the Hilbertian calculus of classical logic the following axioms and rule:

(T) $\Box\phi\rightarrow\phi$;
(S5) $\neg\Box\phi\rightarrow\Box\neg\Box\phi$;
(K) $\Box(\phi\rightarrow\psi)\rightarrow(\Box\phi\rightarrow\Box\psi)$;
(Nec) $\phi \vdash \Box\phi$.

When we strike out the axioms (T) and (S5) we gain a calculus that is in a certain sense "minimalistic" and which nowadays is called K.

The primary reason why there are so many calculi for necessity appears to be that the concept of necessity is itself unclear. There are various

MODAL LOGIC S5

Hilbertian calculus

($\wedge$I′) $\phi\rightarrow(\psi\rightarrow(\phi\wedge\psi))$ ($\wedge$E1′) $(\phi\wedge\psi)\rightarrow\phi$ ($\wedge$E2′) $(\phi\wedge\psi)\rightarrow\psi$

($\rightarrow$1) $\phi\rightarrow(\psi\rightarrow\phi)$ ($\rightarrow$2) $(\phi\rightarrow(\psi\rightarrow\xi))\rightarrow((\phi\rightarrow\psi)\rightarrow(\phi\rightarrow\xi))$

($\neg$E′) $\phi\rightarrow(\neg\phi\rightarrow\psi)$ ($\neg$I′) $(\neg\phi\rightarrow\psi)\rightarrow((\neg\phi\rightarrow\neg\psi)\rightarrow\phi)$

(K) $\Box(\phi\rightarrow\psi)\rightarrow(\Box\phi\rightarrow\Box\psi)$;

(T) $\Box\phi\rightarrow\phi$ (S5) $\neg\Box\phi\rightarrow\Box\neg\Box\phi$;

(MP) $\phi\rightarrow\psi,\ \phi \vdash \psi$ (Nec) $\phi \vdash \Box\phi$

Figure 8.3

6. See the Appendix of Lewis and Langford (1932).

versions of necessity: logical, physical, deontic, and what not. But even if we restrict our attention to one of the versions, it is still not quite clear what exactly holds for it (if something is possible, does it follow that is it possible that it is necessarily possible?). Moreover, once we have a variety of calculi for $\Box$ we can try to derive new ones just for the hell of it (or for the reasons that they may underlie interesting mathematical structures).

The necessity operators are just a sample of a great variety of operators, which were proposed in addition to the core ones. Given the wealth of them, the question, of course, comes up as to how much freedom we have in introducing such new operators – in particular, as to whether such an operator can be constituted by *any* kind of inferential pattern.

The answer is negative in the sense that many such collections of inferential rules do not lead to anything that can be useful as a logical operator. Consider, for example, a hybrid between disjunction and conjunction, $\otimes$, which is defined in terms of the introduction rules of the former and elimination rules of the latter:

(⊗I1) $\phi \vdash \phi \otimes \psi$

(⊗I2) $\psi \vdash \phi \otimes \psi$

(⊗E1) $\phi \otimes \psi \vdash \phi$

(⊗E2) $\phi \otimes \psi \vdash \psi$.

As for any ϕ and ψ we have both $\phi \vdash \phi\otimes\psi$ and $\phi\otimes\psi \vdash \psi$, we get $\phi \vdash \psi$ via (Cut).[7] Therefore, if our calculus admits (Cut), then the presence of $\otimes$, governed by the previous rules, makes any sentence inferable from any other sentence (hence, if the calculus has at least one theorem, then every sentence is a theorem). It does not seem that such an operator would be useful – on the contrary, it seems to be a disaster for any calculus in which it would appear.

This leads to the question of where the limits of "reasonable" logical operators are; in particular, whether it is possible to characterize collections of rules and/or metarules that lead to such "reasonable" operators. One such characterization was proposed by Belnap (1962) and Hacking (1979), who argue that the hallmark of such a "reasonable" logical operator is its *conservativeness*: the fact that it does not introduce any new inferential links among sentences which do not contain it. (Note that $\otimes$ violates this condition; in particular, it introduces an inferential link between *any* two sentences.[8])

7. The operator thus behaves like the notorious **tonk**, tabled by Prior (1960) and discussed by many authors afterwards (Belnap, 1962; Cook, 2005; Wansing, 2006). See also Peregrin (2014, Chapter 8).
8. However, also note that there is a sense in which classical negation is not conservative in this sense. While we will have a conservative extension if we add all the operators of classical propositional operators in one sweep to a language, if we add implication and then negation, the last extension will not be conservative, for it will let us prove new theorems not containing negation.

But despite the fact that some sentential operators may turn out to be this vicious, we can introduce a lot of sentential operators beyond the core ones, some of them trying to capture some logical words of natural language, others for various different reasons.

8.5 Substructural Calculi

We have stated that the structural rules look almost self-evident and that they fit very well with the enterprise of proving. However, we have also indicated that there may be reasons to suspend some of them; this, of course, would lead us to calculi quite different from the ones we have considered up to now.

The fact is that a lot of recent and current researchers investigate logics in which one or more of the structural rules do not hold.[9] Probably the most understandable idea is giving up the rule (Ext). As we have already noted, it is to a certain extent plausible to say that, for instance, $A \wedge B$ is inferable from A and B but not from A, B, and C (for C is "superfluous"). This usually leads to the relevant (or relevance) logics, mentioned in §8.2, with usually quite complex calculi.

Rejection of (Con) is more difficult to substantiate. It leads to viewing the premises of an argument as "resources" that are "spent" by being used. The rejection of this rule is the basis of so-called linear logic.[10]

The most complex problems surround the rejection of the (Cut) rule. Originally, Gentzen took pains to show that in certain calculi, even if we strike out this rule from the rules of derivation, it will continue to be admissible – that is, the particular calculi will obey the (Cut) rule even when it is not explicitly stipulated. This was welcome because calculi without the explicit (Cut) were usually much easier to analyze than those with (Cut); especially, it was easier to prove their consistency and decidability.

An overview of reasons which may make us give up individual structural rules, and also details about (Cut) elimination, are given by Paoli (2013).

8.6 "Expressive Power" of Propositional Languages

Return to the examples of arguments that we listed in §2.2. As is clear, we can formalize three of them on the propositional level:

$$\text{(LFA4)} \qquad \frac{A \wedge B}{A}$$

9. See Restall (2000).
10. See Girard (1987).

(LFA5) $$\frac{A \qquad A \rightarrow B}{B}$$

(LFA6) $$\frac{A \vee B \quad \neg B}{A.}$$

With some inventiveness regarding using the connectives, we can add a fourth one:

(LFA7) $$\frac{\neg A \wedge \neg B}{\neg A.}$$

If we use the definition of the connective ↓ introduced earlier, we can have an even more perspicuous formalization:

(LFA7′) $$\frac{A \downarrow B}{\neg A.}$$

The ability of a formal language to capture instances of following-from ("intuitively" correct arguments) is sometimes called the "expressive power" of the language. (A more exact definition of this concept would be quite complicated, so we leave it at the informal level.) We see that the expressive power of the languages of propositional logic is limited, even such simple arguments like (A1)-(A3) are beyond its ken. This motivates a search for expressively more powerful languages.

Note also that the letters *A* and *B*, as we have just employed them, are parameters, hence the sentences they make up are elements of the formal language of propositional logic, and that (LFA4)-(LFA7) are not arguments but rather argument forms. Indeed, they point out the logical forms of arguments.

Can we also have propositional constants to regiment specific sentences? Of course we can (though they usually are not introduced into propositional languages). We could, for example, introduce the constant ***DogFido*** to regiment the sentence *Fido is a dog* and ***SickFido*** to regiment *Fido is sick*. (Of course, in order for them to become genuine constants – rather than mere constants "in spe" – they would have to be assigned specific inferential roles.)[11] Then we can regiment (A4) as

(LSA4) $$\frac{\boldsymbol{DogFido} \wedge \boldsymbol{SickFido}}{\boldsymbol{DogFido}}$$

11. We might thus have, for example, ***DogFido*** as the single premise of an argument with the conclusion ***AnimalFido***, etc.

This is a specific argument within a specific *formalized* (rather than *formal*) language. (We can call (LSA4), to distinguish it from (LFA4), the logical *structure* (rather than *form*) of (A4).)[12] We can move from the structure to the form by the process of abstraction in that we replace the constants by parameters.

12. This is a terminology introduced by Peregrin and Svoboda (2017), where it is discussed in greater detail.

9 Quantification

9.1 From Propositional to Predicate Logic

To enhance the "expressive power" of propositional logic (in the sense of the ability to capture more instances of following-from), we need a language that would not take sentences as indecomposable wholes; we need to consider parts of sentences and to reveal some new pieces of "logical vocabulary," which can be regimented as logical constants, inside them. So the language of predicate logic does not contain any sentential words (simple symbols of the category of sentences), and it works only with sentences composed of subsentential parts. Hence, even the atomic sentences – that is, those which do not consist of other sentences – consist of simpler expressions.

An atomic sentence of the language of predicate logic consists of an n-ary predicate word and n subject expressions. Thus, we have new categories of expressions: n-ary predicates (for any natural n) and subjects. The only predicates are words; the subjects can be either words or complex expressions. The latter is the case if we have the categories of n-ary functors (for any natural n), which form a subject expression together with n subject expressions; for the sake of simplicity, we will disregard functors; the reason why we can do this is that functors can be "emulated" in terms of predicates.[1]

Thus, the vocabulary of a language of predicate logic equals that of the language of propositional logic minus sentential words plus predicate and subject words plus quantifiers.

In particular, we assume that we have subject words a_1, a_2, . . . and predicate words (each with an assigned arity, which we mark as the superscript) $P_1^1, P_2^1, \ldots, P_1^2, P_2^2, \ldots$; and we have the syntactic rule

> If P is an n-ary predicate and a, b, . . . is an n-tuple of subject words, then $P(a, b, \ldots)$ is a sentence.

1. Thus, for example, an n-ary functor f can be "emulated" by an (n+1)-ary predicate F such that $F(t_1, \ldots, t_n, t)$ iff $t = f(t_1, \ldots, t_n)$. Thus, instead of $P(\ldots, f(t_1, \ldots, t_n), \ldots)$, we can have $P(\ldots, t, \ldots) \wedge F(t_1, \ldots, t_n, t)$.

We keep assuming that we have unary sentential operators and binary sentential operators plus the rules

> If O is a unary sentential operator and ϕ a sentence, then Oϕ is a sentence.
> If O is a binary sentential operator and ϕ, ψ are sentences, then (ϕOψ) is a sentence.

The most standard version of the predicate calculus is based on the core operators of propositional logic, that is, $\neg$, $\wedge$, $\vee$, and $\rightarrow$, but any kind of modified or additional ones can be imported too.

Now it is clear that, unless we introduce some new logical constants, we will not gain any expressive power greater than the corresponding propositional calculus in the sense that we would not have any new theorematic arguments. The new constants that are standardly added to the language of predicate logic and that make it much more powerful (though also much more complex) than propositional logic are the two quantifiers, $\forall$ and $\exists$. The quantifiers necessitate a brand new kind of expression: (subject) variables x_1, x_2, x_3, ... They belong to the category of subjects but for some purposes they must be distinguished from subject words. In particular, the syntactic rule for quantifiers runs as follows:

> If U is a quantifier, x a variable, and ϕ a sentence, then $Ux\phi$ is a sentence.

Moreover, the introduction of the quantifiers brings about the division of the category of sentences into two kinds: open and closed. The terminology is usually modified so that sentences in general are rechristened as *formulas* and the term *sentence* is reserved for *closed* formulas only. What is the difference between open and closed formulas? Roughly speaking, open formulas are those which contain variables while closed formulas are those which do not. More precisely, open formulas are those which contain variables *essentially*, where the occurrence of a variable x within a formula becomes inessential if it is bound by a quantifier – that is, if it occurs within a subformula of the shape $Ux\phi$.

What is usually added to the predicate calculus is the logical constant =, which is a binary predicate, so that it forms a sentence together with two subjects.

While the language of propositional logic was rather transparent in the sense that it mostly replicated some simple syntactical structures of natural languages (the fact that we can negate sentences, or form more complex sentences by joining simple ones by means of connectives), the structure of the language of predicate logic is much less transparent. Therefore, it is vital to understand the working of quantifiers and of the ways quantified sentences are supposed to regiment those of natural languages. This is what we are going to discuss in the following sections.

A note about the symbols that we employ. Similarly as we employed the Greek letters ϕ, ψ, . . . as propositional "metaparameters," that is, as symbols that we use in order to speak about sentential expressions, we now employ other types of "metaparameters" to talk about other kinds of expressions. Thus, we use the symbols x, y, . . . as the "metaparameters" for variables (note that these symbols are not among the variables of our languages – the variables are x_1, x_2, . . .); for predicate words we will use P, Q, . . . ; for subject words a, b, . . . ; and for quantifiers U. We should probably use Greek letters for all the kinds of metaparameters; however, this would lead to a notation that would be so different from what we usually find in logical textbooks that it could be misleading.

9.2 Quantifiers – Motivation

While the logical constants of propositional logics may be seen as more or less straightforward regimentations of the selected logical expressions of natural language, the situation with quantifiers is much more complicated. It is important to realize how different the usual predicate calculi are from the propositional ones. Propositional calculi, we saw, are originally based on the idea of *direct regimentation*: on the idea that some logical elements of natural language – mostly connectives that help us build more complex sentences out of simpler ones (*and*, *or*, *if-then,* . . .) – can be "replaced" by their exactly delimited counterparts ($\wedge$, $\vee$, $\rightarrow$, . . .). The quantifiers that have come to be employed in the languages of common predicate calculi are different: they do not have such straightforward counterparts in natural languages. Also, they are inseparable from variables, together with which they introduce brand new kinds of philosophical problems.

But let us, for a moment, forget about what we know about standard predicate logic and imagine that we want to extend propositional logic to the predicate one without deviating too much from the process of direct regimentation. We want to find, in natural language, additional logical words or phrases that could be directly regimented (similarly as the connectives which have become imprinted in propositional logic).

When we decompose atomic sentences into predicate and subject parts, we are likely to notice that only some of the subjects are what can be considered names, even in the broad sense of the word in which a name is anything that purports to refer to an object. There are a lot of subjects that are of a different nature, subjects such as

everything
some animal
every boy
most people
more than half of inhabitants of London,
etc.

They do not seem to fit into the same category with prototypical names such as *Duke Ellington*, *the president of the USA*, *the guy over there*,[2] etc. And, moreover, at least some of them look like candidates for *logical* words, especially *every*, *some*, *most*, etc. In a typical case, a word like *some, every*, or *most* constitutes a subject together with a common noun, that is, with what from the viewpoint of logical regimentation falls into the category of unary predicates. Hence, it would seem we might have a new category of logical constants, namely expressions which take unary predicates to subjects. Let us call these kinds of expressions (in agreement with what has become common within formal semantics) *determiners*.

Imagine that we introduce two new logical constants of this category, $\forall$ and $\exists$, to regiment the English words *every* and *some*. Thus, if P is a unary predicate (regimenting, e.g., *animal*), $\forall P$ and $\exists P$ are subject expressions (regimenting *every animal* and *some animal*), $Q(\forall P)$ and $Q(\exists P)$ are thus sentences for any unary predicate Q (regimenting, for example, *Every animal breathes* and *Some animal breathes*). Which rules of inference would be appropriate to characterize these two new constants? What comes to mind immediately is

($\forall$E) $\quad P(\forall(Q)), Q(a) \vdash P(a)$, and

($\exists$I) $\quad P(a), Q(a) \vdash P(\exists(Q))$.

Note that from the viewpoint of constituting sentences, a determiner is an expression that takes two unary predicates to a sentence. Hence, determiners could also be seen as of this different type and we could write $\forall(P, Q)$ and $\exists(P, Q)$ instead of $Q(\forall P)$ and $Q(\exists P)$, respectively. (Viewed in this way, determiners become akin to the subject matter of Aristotle's syllogistic, namely $\forall(P,Q)$ becomes what has come to be abbreviated as $P\mathrm{a}Q$, while $\exists(P, Q)$ becomes $P\mathrm{i}Q$. Thus, we can see syllogistics as an early attempt at the calculization of determiners.)

This approach to quantifiers, as a matter of fact, appeared in logic only relatively recently, in the context of the so-called formal or (model-theoretic) semantics that attempts to create logics that "copy" the syntactic structures of natural languages more closely than ordinary ones. Here determiners moved to the center of attention especially within the so-called *theory of generalized quantifiers*.[3] Here, however, the determiners are usually not considered as primitive constants but are defined in terms of standard logical quantifiers.

To move closer to the standard quantifiers, note that instead of choosing *every* and *some* as the natural language logical words which we want to directly regiment, we could have chosen *everything* and *something*. (If we master them, and if we manage to regiment relative clauses, then we

2. And as Russell (1905) famously showed, even taking the definite descriptions as "names" is not unproblematic.
3. See, e.g., Barwise and Cooper (1981) or Loebner (1987).

can perhaps regiment subject phrases like *some animal* and *every woman* as *something that is an animal* and *everything that is a woman*.) If we use the symbols $\forall$ and $\exists$ to regiment them, we could leave them syntactically on a par with names, hence consider them as constant subject words, and consequently have sentences like $P(\forall)$ and $Q(\exists)$. (Alternatively, we could of course consider quantifiers not as something which is taken by a unary predicate to a sentence but rather as something which takes a unary predicate to a sentence – thus we would have $\forall(P)$ and $\exists(Q)$.)[4] The inferential rule that would seem immediately appropriate for them would then be

($\forall$E) $P(\forall) \vdash P(a)$, and
($\exists$I) $P(a) \vdash P(\exists)$.

There seem to be two problems with this treatment of $\forall$ and $\exists$ (both with their treatment as determiners and as specific subjects). The first is that we have only an elimination rule for $\forall$, whereas we would seem to need an introductory one also (and the other way around for $\exists$). The second problem is that if we consider sentences with predicates of greater arities, this treatment of $\forall$ and $\exists$ seems to lead to some undesired ambiguities. Consider the sentence

$R(\forall,\exists)$.

It might be seen as the form, for example, of

Everything is caused by something.

The trouble, however, is that this sentence appears to be ambiguous: on one of its readings it says that everything has a (possibly different) cause, while on the other it says that there is something that causes everything, that is, that

There is something such that everything is caused by it.

Could the regimentation of this second reading be $R(\exists,\forall)$? Not really, for given $R(\forall,\exists)$ regiments *Everything is caused by something*, $R(\exists,\forall)$ would naturally regiment *Something is caused by everything*, which is, needless to say, something very different from *There is something such that everything is caused by it*.

It would therefore seem that the constants $\forall$ and $\exists$, introduced in this way, lead to the method of regimentation that lacks means for

4. This, as we will see, is what usually happens when we consider semantics of the quantifiers, especially using lambda-categorial grammar (see §10.4), where we need the "quantified subject expressions" to be of a different category than the "ordinary" ones.

accommodating substantial differences present in natural language. The problem, it would seem, is this. Given a binary predicate (or, for that matter, a predicate of a greater arity), we need to know the order of its arguments, *viz*. which of the arguments fills which place in the predicate (for $R(a,b)$ is, of course, something different from $R(b,a)$). But in case the arguments are quantifiers, there is also a *different* kind of order that has to be captured, the order that distinguishes *for every x there is some y* and *there is some y such that for every x*.

There are, to be sure, various ways of representing this order. We may, for example, introduce superscripts to quantifiers, to distinguish between $R(\forall^1,\exists^2)$ (regimenting *Everything is caused by* (*a*) *something*) and $R(\forall^2,\exists^1)$ (*Everything is caused by* (*the same*) *something*, that is, *Something causes everything*). There are many other ways of accomplishing the same thing; for example, we can factor out the quantifiers before the predicate, so that what was marked by the superscript is now marked by their order, namely $\forall\exists$ vs. $\exists\forall$. However, we must still keep the information as to which quantifier fills which slot in the predicate, which might again be accomplished by means of the superscript, so that now we have $\forall^1\exists^2R$ instead of $R(\forall^1,\exists^2)$ and $\exists^2\forall^1R$ instead of $R(\forall^2,\exists^1)$.

An interesting alternative, introduced by Quine (1960), is to assume that quantifiers always fill the slots of the predicate in the order in which they are listed and to introduce ways of permuting the slots. Thus, if we have a functor **Ex** which exchanges the slots of a binary predicate (that is, it maps a binary predicate R on a predicate $\mathbf{Ex}(R)$ such that $\mathbf{Ex}(R)(a,b)$ is $R(b,a)$), then $\exists^2\forall^1R$ can be simply $\exists\forall R$ while $\forall^1\exists^2R$ is $\forall\exists\mathbf{Ex}(R)$. (Quine introduces a minimalistic set of such functors with which we can accomplish everything we need.)

However, so far we have solved only a part of the problem, for up to now we considered ways of combining quantifiers directly with *simple* predicates, whereas what we would need, over and above this, is to be able to regiment sentences such as

Everything is either caused by something or it causes something.

In view of the fact that in standard predicate logic the regimentation would be $\forall x\exists yR(x,y)\vee R(y,x)$, we could think of regimenting the sentence as

$$\forall^{1,4}\exists^{2,3}R\vee R,$$

the translation of which into the Quinean idiom would be

$$\forall(\exists(\mathbf{Refl}(\mathbf{Refl}(\mathbf{Perm}(\mathbf{Pad}(\mathbf{Refl}(\mathbf{Perm}(\mathbf{Perm}(R\vee R)))))))),$$

where Refl, Perm, and Pad are functors accomplishing certain basic permutations of slots in predicates:

$$\mathbf{Pad}^{n}(P)(x_0, x_1, \ldots, x_n) = P(x_1, \ldots, x_n);$$
$$\mathbf{Perm}^{n}(P)(x_1, \ldots, x_n) = P(x_1, x_3, \ldots, x_n, x_2);$$
$$\mathbf{Refl}^{n}(P)(x_1, \ldots, x_n) = P(x_1, x_1, \ldots, x_n).$$

However, the trouble is the component "$R \vee R$," which is not a well-formed expression since a sentential operator such as $\vee$ interconnects sentences, not predicates. Hence, we now would need to define a version of $\vee$ that can connect predicates, and similarly for other sentential operators. Thus, the final Quinean version might look as follows

$$\forall(\exists(\mathbf{Refl}(\mathbf{Refl}(\mathbf{Perm}(\mathbf{Pad}(\mathbf{Refl}(\mathbf{Perm}(\mathbf{Perm}(R \cup R)))))))))),$$

where $\cup$ is a disjunction able to interconnect predicates.

Now we can, of course, also use the most standard way of connecting "factored out" quantifiers with the respective slot of the predicate: we fill the slot by a placeholder symbol and put the same symbol beside the quantifier that belongs to that place. Thus, we have $\forall x \exists y R(x,y)$ instead of $\forall^1 \exists^2 R$, and we have $\exists y \forall x R(x,y)$ instead of $\exists^2 \forall^1 R$. And, indeed, we would have $\forall x \exists y(\mathrm{R}(x,y) \vee R(y,x))$ instead of $\forall^{1,4} \exists^{2,3} R \cup R$.

The reason why it took us so long to work our way toward this standard (and perhaps *prima facie* obvious) solution to the problem is that, given the motivation discussed in this section, we do not really need variables – they are dispensable auxiliaries that we need as part of a handy technical solution, not indispensable parts of the language of the predicate calculus. This, I think, is fundamentally important.

9.3 Quantifiers as Infinitistic Sentential Operators?

Let us now look at the way in which quantifiers, and their underpinning machinery of variables, as a matter of fact, came into being in the writings of the ur-fathers of modern logic. Gottlob Frege, when introducing quantifiers, considered decomposing a sentence into two parts and then removing one of the parts, replacing it by a special kind of symbol. Then we may formulate a specific kind of sentence which is true iff replacing this special symbol by any expression of the suitable kind in our language always produces a true sentence (Frege, 1879, p. 19):

> In the expression for a judgment we can always regard the combination of signs to the right of $\vdash$ as a function of one of the signs occurring in it. *If we replace this argument by a German letter and*

> *if in the content stroke we introduce a concavity with this German letter in it, as in*:
>
> 
>
> *this stands for the judgment that, whatever we may take for its argument, the function is a fact.*

Here Frege introduced what in the contemporary notation would be "$\forall x P(x)$" by stipulating that it is true iff "the function is a fact whatever we take its argument to be." This, needless to say, is somewhat blurry; it is especially not clear as to how far this is to be interpreted in what is now considered as a *substitutional* way (as saying "whichever name of the language in question we substitute for 'x' in '$P(x)$' we gain a true sentence") or in the *objectual* one ("whichever object from the universe we take, it will satisfy '$P(x)$' "); but let us, for the time being, stick to the first option. Thus, $\forall x P(x)$ comes to be something as an infinite conjunction of all the sentences of the form $P(\ldots)$.

Something very similar emerges from the explanation of quantifiers given by Russell (1905, p. 480):

> Then *everything* and *nothing* and *something* (which are the most primitive of denoting phrases) are to be interpreted as follows:
> C(everything) means "$C(x)$ is always true";
> C(nothing) means " '$C(x)$ is false' is always true";
> C(something) means "It is false that '$C(x)$ is false' is always true."

Again, there is, from the current viewpoint, the ambiguity between the substitutional and the objectual reading, but if we stick to the former, we get the equivalence of "C(everything)" with the possibly infinite conjunction of the sentences of the form "$C(x)$."[5]

5. What might appear strange is that both Frege and Russell take recourse to what we would call *metalanguage* when explaining quantificational sentences. Why do they talk about the *truth* of the sentences of the object language, thus having to do the "semantic ascent" to the metalanguage? One explanation is that their employment of the truth-predicate is only a non-committal expedient. As disquotationalists of all sorts point out, we can have the truth predicate in a language, and we can say such things as *Everything Aristotle wrote was true* without thereby committing ourselves to truth as a substantial property. The truth predicate can be seen merely as a "syntactic tool": if we were able to assert everything that Aristotle wrote, we would not need to assert *Everything Aristotle wrote was true*, but as this is not really possible, the shortcut comes in very handy. And similarly we could take Frege's and Russell's employment of the truth predicate as not a semantic ascent to a metalanguage but rather as a syntactic shortcut. What both Frege and Russell seem to convey is that the sentence "C(everything)" is equivalent to the set of all sentences of the shape "$C(\ldots)$", i.e., it is something like an infinite conjunction.

Anyway, from the assimilation of the general quantifier (and its natural language counterpart, *everything*) to infinite conjunction (and the existential quantifier to infinite disjunction), what the introduction rule for general quantifiers should look like follows: it should have an infinite number of premises. Hence, it would look like

(∀I) $P(a), P(b), \ldots \vdash \forall x P(x)$, where a, b, . . . are all the subject words (or all "canonical" subject words)[6] of the language.

And this looks like a problem, for something with an infinity of premises does not really look like a *rule*.

Moreover, there is another, well-known problem. Even if we accepted infinitistic rules, it is not clear whether the substitutional construal of quantifiers we have considered so far is really satisfactory. Is it substantiated to conclude that everything is P on the basis of all the sentences of the form $P(\ldots)$? Intuitively, it is only if we have names for everything. But can we assume that we always have names for every object?

9.4 Variables as "Generic Names"?

Hence, from the viewpoint of building a calculus, the assimilation of quantifiers to infinitistic operators is not quite satisfactory. There is, however, a different line of thought that may look promising: the assimilation of *every* to *any*. The basic idea is that we can conclude that P is proven to hold for *everything* once it is proved to hold for *anything* – namely for something of which we do not assume anything specific. Which is, that is to say, unspecific enough to be able to represent *any* object.

This would necessitate a new kind of symbol, some "generic names," hence names that can name anything. We have already seen that we may use variables to underpin the working of quantifiers, and that purely in this role variables are dispensable. Now we can reuse them in a new way: they can act as our generic names. The basic idea is that if $P(x)$ is a theorem, then so is $\forall x P(x)$. This can be embodied into

(Gen) $\vdash P(x) \,//\, \vdash \forall x P(x)$,

but this is a metarule. Can we have it in the form

(Gen+) $P(x) \vdash \forall x P(x)$?

6. It may be the case that there are distinguished kinds of subjects such that every other subject is equivalent to one of them, such as the numerals within the language of arithmetic.

This is not quite clear, and it reveals certain perplexities concerning the concept of "generic name." One thing is that if we have the operator $\rightarrow$ fulfilling (Ded+), we would have

$$\vdash P(x) \rightarrow \forall x P(x).$$

This does not seem quite right – for according to the sense we gave to x it appears to read *If anything is P, then everything is P*, which is usually understood as synonymous with *If something is P, then everything is P*. In fact, in classical logic this does yield

$$\vdash \exists x P(x) \rightarrow \forall x P(x),$$

which is clearly absurd. Hence, we would need to reject (Gen+) or (Ded+). The usual option (see, for example, Mendelson, 1964, §§2.3, 2.4) is sacrificing (Ded+), so that we can keep (Gen+). In this case, (Ded+) only holds for a restricted part of language, where it does not get into conflict with the effect of variables.

Viewed thus, variables appear to be a much more substantial part of the language of logic than mere dispensable auxiliaries. Can we see them as directly regimenting something which is present – in such or another way – in natural language? Do we have, in the language, something like "generic names"?

Can we take the idea that variables regiment *anything* at face value? In fact, if we were to have only a single variable in our language, this might be a plausible view. However, we have many; and there is certainly a substantial difference between $R(x,y)$, $R(y,x)$, and $R(x,x)$, which would come out the same if *every* variable were to be seen as regimenting *anything*.[7]

However, it would be possible to think about variables in the context of regimentation of the refined kind of linguistic practices that we especially find in mathematics. Perhaps "$R(x,y)$" can be seen as regimenting something like "Take an arbitrary object, call it x, and another arbitrary object, y, and assume $R(x,y)$." But note that this brings us some way away from our original idea about regimentation as the process of replacing certain words or phrases of natural language by their more precise and more definite artificial variants.

7. We could also consider other expressions of natural language as playing the role of "generic names," as, for example, pronouns. However, despite the fact that the role of such words in language might be seen as including a "generic ingredient," their role is much more complicated than to be considered as "generic names." (On the inferential role of pronouns, see, for example, Peregrin, 2000a.)

9.5 Variables and Syntactic Structure

Variables and open formulas may serve still one more purpose, namely to dramatically enhance, in a covert way, the syntactic structure of the language to which they belong. Consider the English connective *and*. It can be used to join not only sentences, but also, for example, predicates. We can not only say *Somebody is tall and somebody is bald* but also *Somebody is tall and bald* (which, needless to say, is something different – it requires the conjunctive predicate). We can do no such thing with $\wedge$ within the standard language of predicate logic – it is restricted to joining sentences.

Now if we have variables, we can form the conjunctive open sentence, $P(x) \wedge Q(x)$, which we can view as a "pseudopredicate": when we take a subject, plug it in place of all occurrences of x, we gain a (closed) sentence. Thus, in this sense, $P(x) \wedge Q(x)$, as well as any other open formula, can be seen as a complex predicate.

This amplifies the otherwise austere syntactic structure of the predicate calculus in an immense way. However, it does so in a covert way – an open formula is *not* really a predicate. An attempt to make this explicit and transparent might be found in the language of the *lambda calculus*.[8] According to it, an open formula is not yet a predicate, but there is an explicit way of turning it into one. The lambda-abstraction rule, restricted to the needs of predicate calculus, goes as follows:

> If ϕ is a formula and $x_1, \ldots, x_n$ are variables, then $\lambda x_1 \ldots x_n\phi$ is an n-ary predicate.

Moreover, the lambda calculus is a *calculus* in that it stipulates a certain equivalence between the expressions. In particular, it stipulates that (where $\phi[a_1/x_1, \ldots, a_n/x_n]$ is ϕ in which all free occurrences of $x_1, \ldots, x_n$ are replaced by $a_1, \ldots, a_n$, respectively)

$$(\lambda x_1 \ldots x_n \phi)(a_1, \ldots, a_n) \text{ is equivalent with } \phi\left[a_1/x_1, \ldots, a_n/x_n\right].$$

Given that any open formula can be transformed into a predicate, we can see quantifiers simply as second-order predicates (that is, predicates of predicates) taking predicates to sentences:

> If U is a quantifier and P a unary predicate, then $U(P)$ is a sentence.

8. See Barendregt (1980).

Indeed, instead of $Ux\phi$ we will now have $U(\lambda x\phi)$. Moreover, we can dispense with open formulas altogether, for we can reformulate the rule of lambda-abstraction as

> If ϕ is a (closed) formula and $x_1, \ldots, x_n$ are variables then $\lambda x_1 \ldots x_n\phi[x_1/a_1, \ldots, x_n/a_n]$ is an n-ary predicate such that $\lambda x_1 \ldots x_n\phi[x_1/a_1, \ldots, x_n/a_n](b_1, \ldots, b_n)$ is equivalent with $\phi[b_1/a_1, \ldots, b_n/a_n]$.

Then we can also get rid of variables, for instance, by Quine's method of predicate functors.

How do we build, for example, a formula corresponding to $\forall x\exists yR(x,y)$? Well, we start from $R(a,b)$, produce $\lambda yR(a,y)$, we apply the existential quantifier $\exists(\lambda yR(a,y))$, then we produce $\lambda x\exists(\lambda yR(x,y))$ by another lambda-abstraction, and then we, finally, apply the general quantifier: $\forall(\lambda x\exists(\lambda yR(x,y)))$.

Though the lambda-abstraction rule is not an explicit part of the usual language of the predicate logic, we can see it as being at work in it implicitly. It is implicit to the understanding of open sentences as "pseudopredicates," and it makes us articulate the introduction and elimination rules for quantifiers in a way much more general than we did it previously:

(∀E) $\quad \forall x\phi \vdash \phi[a/x]$

(Gen+) $\quad \phi \vdash \forall x\phi.$

But once we have lambda-abstraction, there is again no reason either for this generalization or for variables. The only variable-binding rule is now that of lambda-abstraction (and in it the employment of variables can be replaced by any of the techniques mentioned previously). Thus, instead of, for example,

$$\forall x(P(x)\wedge Q(x))$$

we will now have

$$\forall(\lambda x(P(x)\wedge Q(x))).$$

Moreover, now we can formulate (∀E) in a simpler way:

(∀E) $\quad \forall(P) \vdash P(a).$

On the other hand, having gotten rid of open formulas, we can no longer formulate (Gen+) as we would again need an infinite number of premises.

9.6 Variables and Natural Language

To summarize the morals we have assembled throughout this chapter, we must, first and foremost, stress that *there are no variables in natural language*; that is, variables are nothing that a logician discovers as (covertly) present in natural language, they are the logician's expedients of giving an account for the language and its workings.[9] In particular, they are an integral part of a certain technical package, involving quantifiers, which was introduced to capture certain kinds of "quantificational" subject expressions of natural language. It made the enterprise of logical regimentation of natural language largely deviate from the "direct" route it used to take on the level of propositional languages.

Variables and quantifiers have turned out to be expedients indispensable in the sense that they have made for some technically very efficient ways of regimentation and for certain relatively transparent formal languages of logic. However, we should not forget that the same which we achieve in terms of variables could be, in principle, also achieved by other means, especially without engaging anything like variables. It is crucial to realize this, for it reveals the true roles of variables as being, practically, extremely useful items, but theoretically inessential.

Thus, we can say that variables are "merely" (handy) tools of the logicians who need to account for some phenomena concerning natural language. Hence the rules, etc., involving variables do not have any *direct* counterpart in natural language; rules like that of lambda-abstraction are not means of direct regimentation. In general, the employment of variables put logic on a track in which logical regimentation became more a matter of a creative art than a mechanical replacement of natural language expressions by artificial constants and parameters, and it made the relationship between the artificial languages of logic and natural language quite convoluted.[10]

9. Thus, I reject the view of Shapiro (1998, p. 139) that while "parentheses are artifacts, serving to make the formulas unambiguous," "variables, connectives, and quantifiers do have counterparts in natural languages, more or less."
10. It is not that the languages of propositional logic cannot be employed, for the purposes of logical formalization, in various creative ways. They can and often are. However, their elements do have direct counterparts in natural language, so usually even a very inventive formalization of a sentence can be seen as a matter of a paraphrase of the sentence. For propositional languages, this is often much harder.

10 Calculi of Predicate Logic

10.1 The Calculus of Classical (First-Order) Predicate Logic

We have already sketched the language of predicate logic (and it is summarized in Figure 10.1).

THE LANGUAGE OF CLASSICAL (FIRST-ORDER) PREDICATE LOGIC

Vocabulary:

Subject words: $a_1, a_2, \ldots$

Subject variables: $x_1, x_2, \ldots$

N-ary predicate words (for every n): $P^n{}_1, P^n{}_2, \ldots$;

specific binary term $=$

Unary sentential operator: $\neg$

Binary sentential operators: $\wedge, \vee, \rightarrow$

Quantifiers: $\forall, \exists$

Syntactic rules:

A subject word or a subject variable is a subject

If P is an n-ary predicate and a,b,... is a sequence of n subjects, then $P(a,b,...)$ is an (atomic) formula

If O is a unary sentential operator and ϕ a formula, then $O\phi$ is a formula

If O is a binary sentential operator and ϕ and ψ are formulas, then $(\phi O \psi)$ is a formula

If U is a quantifier, x a variable, and ϕ a formula, then $Ux\phi$ is a formula.

Delimitation of sentences

A variable x is *free* in a formula ϕ iff: ϕ is atomic and x occurs in ϕ; or ϕ is $O\psi$ and x is free in ψ; or ϕ is $\psi O \xi$ and x is free either in ψ or in ξ; or ϕ is $Uy\psi$ and x is free in ψ and it is not y.

A formula is open if it contains a free variable; otherwise it is closed.

A closed formula is a sentence.

Figure 10.1

The calculus of classical predicate logic, built atop of this bare language, is an extension of that of classical propositional logic. What we must add, first and foremost, are the rules (∀E) and (Gen+). In the Hilbertian variant we usually present (∀E) in the internalized form, namely as an axiom

$$(\forall E) \qquad \forall x\phi \to \phi[a/x].$$

(Gen+), on the other hand, cannot be internalized (for the reasons discussed previously) so we have

$$(\text{Gen+}) \qquad \phi \vdash \forall x\phi.$$

Therefore, in the Hilbertian calculus we also need the relativization of this rule, which is

$$(\text{RGen+}) \qquad (\psi \to \phi) \to (\psi \to \forall x\phi).$$

However, what this would yield is the internalization of (Gen+); that is,

$$\phi \to \forall x\phi,$$

which we do not want. So we need to restrict (RGen+) to "innocent" cases, which are usually taken to be the cases when x is not free in ψ. However, even in such a case we can derive the internalization of (Gen+), *namely* if we instantiate ψ as a logical truth (for in such a case $\psi \to \phi$ would become equivalent with ϕ). A solution is to universally quantify the antecedent of (RGen+) (which, given (∀E) and (Gen+), which make $\forall x(\psi \to \phi)$ interderivable with $\psi \to \phi$, is obviously equivalent to (RGen+)):

$$(\text{RGen+}^*) \qquad \forall x(\psi \to \phi) \to (\psi \to \forall x\,\phi).$$

In the calculi of natural deduction, the introduction and elimination rules for ∀ usually look as follows:

$$(\forall E) \quad \forall x\phi \vdash \phi[a/x]$$

$(\forall I)$ $\quad \phi \vdash \forall x\phi[x/a]$, if a does not occur in any formula on which ϕ depends.

The notion of *dependence*, invoked in (∀I), restricts the possibility of adding $\forall\phi[x/a]$ to an existing derivation of ϕ. For example, if I derive $P(a)$ from $P(a)$ and then use (→I) to derive $P(a) \to P(a)$, then this derivation has no premises, that is, $P(a) \to P(a)$ does not depend on anything, so I can go on applying (∀I) and derive $\forall x(P(x) \to P(x))$. On the other hand, if I derive $P(a)$ from the premises $Q(a)$ and $\forall x(Q(x) \to P(x))$, then it depends on the assumption $Q(a)$ that contains a, hence I cannot go on deriving $\forall xP(x)$.

The logical constant =, which is usually considered an integral part of the language of predicate logic, deserves special attention. Aside from quantifiers, it is the sole, new logical constant that the language of predicate calculi usually has over and above the sentential operators (though versions of predicate calculi without it are also possible, only they are usually considered much less interesting). Consider the argument

Fido is a dog
Rex is a dog
____
There are (at least) two dogs.

CLASSICAL (FIRST-ORDER) PREDICATE LOGIC

Hilbertian calculus

($\wedge$I′) $\phi\to(\psi\to(\phi\wedge\psi))$ ($\wedge$E1′) $(\phi\wedge\psi)\to\phi$ ($\wedge$E2′) $(\phi\wedge\psi)\to\psi$

($\vee$I1′) $\phi\to(\phi\vee\psi)$ ($\vee$I2′) $\psi\to(\phi\vee\psi)$

($\vee$E′) $(\phi\to\xi)\to((\psi\to\xi)\to((\phi\vee\psi)\to\xi))$

($\to$1) $\phi\to(\psi\to\phi)$ ($\to$2) $(\phi\to(\psi\to\xi))\to((\phi\to\psi)\to(\phi\to\xi))$

($\neg$E′) $\phi\to(\neg\phi\to\psi)$ ($\neg$I′) $(\phi\to\psi)\to((\phi\to\neg\psi)\to\neg\phi)$

(=1) $\forall x(x=x)$ (=2) $\forall x\forall y((x=y)\wedge\phi)\to\phi[y/x]$

($\forall$E′) $\forall x\phi\to\phi[a/x]$ ($\exists$I′) $\phi[a/x]\to\exists x\phi$

(RGen+*) $\forall x(\phi\to\psi)\to(\phi\to\forall x\psi)$, where x is not free in ϕ

($\forall\exists$) $\forall x(\phi\to\psi)\to(\exists x\phi\to\psi)$, where x is not free in ψ

(MP) $\phi\to\psi,\ \phi\vdash\psi$ (Gen+) $\phi\vdash\forall x\phi$

Gentzenian calculus

($\wedge$I) $\phi,\ \psi\vdash\phi\wedge\psi$ ($\wedge$E1) $\phi\wedge\psi\vdash\phi$ ($\wedge$E2) $\phi\wedge\psi\vdash\psi$

($\vee$I1) $\phi\vdash\phi\vee\psi$ ($\vee$I2) $\psi\vdash\phi\vee\psi$

($\vee$E) $\Gamma,\ \phi\vdash\xi;\ \Gamma,\ \psi\vdash\xi\ //\ \Gamma,\ \phi\vee\psi\vdash\xi.$

($\to$I) $\Gamma,\ \phi\vdash\psi\ //\ \vdash\Gamma,\ \phi\to\psi$ ($\to$E) $\phi\to\psi,\ \phi\vdash\psi$

($\neg$I) $\Gamma,\ \neg\phi\vdash\psi;\ \Gamma,\ \neg\phi\vdash\neg\psi\ //\ \vdash\Gamma,\ \phi$ ($\neg$E) $\phi,\ \neg\phi\vdash\psi$

(=I) $a=a$ (=E) $a=b,\ \phi\vdash\phi[b/a]$

($\forall$I) $\phi\vdash\forall x\phi[x/a]$, where a does not occur in any formula on which ϕ depends

($\forall$E) $\forall x\phi\vdash\phi[a/x]$ ($\exists$I) $\phi[a/x]\vdash\exists x\phi$

($\exists$E) $\phi[a/x]\vdash\psi\ //\ \exists x\phi\vdash\psi$, where a does not occur in ψ and any formula on which ψ depends other than ϕ

Figure 10.2

Is it correct? Well, it may seem so, but, strictly speaking, we can infer the conclusion from the premises only if *Rex* is not merely a different name for Fido – that is, in the case that Rex is a different individual from Fido. Thus, the correct form of the argument would be

Fido is a dog
Rex is a dog
Rex is not Fido
____
There are (at least) two dogs.

And to regiment the third premise we need the operator =.

In general, the constant enhances the "expressive power" of the language in a significant way. Consider the quantifier *there are* (*at least*) *two*, which occurs in the conclusion of the previous argument and which we may regiment as $\exists_2$. As indicated by the previous example, with the help of = we can define it in terms of the standard quantifier:

$$\exists_2 xF[x] \equiv_{\text{Def.}} \exists x\exists y\big(F[x]\wedge F[y]\wedge\neg(x = y)\big).$$

This is not possible without the help of =.

10.2 "Expressive Power" of Predicate Languages

It is clear that the expressive power of predicate logics, that is the possibilities they offer us in respect to regimentation and formalization of natural language arguments, is significantly greater than those of propositional ones. Of the arguments listed in §2.2 they allow us to articulate the common logical form of (A2) and (A3) (see §2.2), namely

$$\text{(LFA3)} \qquad \begin{array}{l} P(a) \\ \underline{\forall x(P(x)\rightarrow Q(x))} \\ Q(a). \end{array}$$

Besides this, they also allow us to analyze a lot of natural language locutions that are far beyond the possibilities of propositional languages.

And yet their "expressive power" remains limited. The structure of the language of predicate logic is still incomparably simpler than those of natural languages: the few categories of expressions on which this language is based are no match for the syntactic wealth of natural languages. Additionally, there may be other natural language candidates for logical expressions of the kind that go beyond the framework of this language. This may again lead us, like in the case of propositional logic, to pondering both the *modifications* and the *extensions* of the standard language and calculi of predicate logic.

In the case of propositional languages, we considered a lot of modifications and extensions of sentential operators – that is, of the logical operators on which these calculi are based. The surplus logical operators of predicate logic are the quantifiers: can we therefore consider modifying the existing ones, or adding more of them?

Due to the fact that quantifiers, as logical constants, do not regiment their natural language counterparts in a similarly direct way as sentential operators regiment connectives and other elements of natural language, we must first clarify what it is they relate to "most directly" in natural language. We have noticed that what we can see as "quantifiers" in natural language are expressions like *every*, *some*, etc., which, in English, associate with predicate expressions to form subject expressions.

It is important to realize that, however straightforward it may come to us to formalize (A3) as (LFA3), it already presupposes a nontrivial skill of "logical analysis," for, to formalize

All dogs are animals

as

$$\forall x(P(x) \rightarrow Q(x)),$$

or, for that matter,

Some dogs are animals

as

$$\exists x(P(x) \wedge Q(x)),$$

is not just a matter of replacing logical expressions by constants and extralogical ones by parameters. (Note that we must engage more logical constants, $\rightarrow$ in the first case and $\wedge$ in the second, to relate the logical quantifiers to their natural language counterparts.) It was Russell who indicated how the "logical analysis" of such sentences should be carried out and initiated us into the art, which we nowadays take almost for granted.

Given this correspondence between the quantified sentences of natural language and their logical formalizations, we may think about other quantificational phrases of natural language. Consider the celebrated regimentation of "definite descriptions" proposed by Russell (1905). As it seems that a sentence of the form *Some P is Q* says something very similar as its variant in which *some* is replaced by the indefinite article *a*, we can see its analysis in terms of $\exists$ as also an analysis of the article; and we may come to ponder the definite one, namely the analysis of sentences of the shape *The P is Q*.

As Russell's analysis suggests, what this sentence does say over and above *Some P is Q* is that there is one and only one *P*. Hence, if we introduce a new quantifier, $\exists!$, such that $\exists!xP(x)$ amounts to *There is one and only one individual that is P*, we seem to be able to analyze

The P is Q

as

$$\exists x(P(x) \wedge Q(x)) \wedge \exists! xP(x).$$

However, Russell also showed how to achieve this by the standard quantifiers of predicate logic, namely as

$$\exists x(P(x) \wedge Q(x) \wedge \forall y(P(y) \rightarrow (y = x))),$$

hence showing how to reduce $\exists!$ to the logical constants we standardly have.

In this way, with the language of predicate logic, logical analysis became not merely the replacing of the logical words of natural language by logical constants and other words by parameters, but rather a very creative art that can lead us from a natural language sentence with a syntactic form to a sentence of a formal language with a very different form. As a consequence, there appeared the question whether we need a richer language, or only enough ingenuity to analyze anything in terms of the austere language of the standard predicate logic.

In this way, logic started to move away from what we were talking about as direct regimentation. While originally logical regimentation looked like a more or less "piece-for-piece matter" (we take parts of the analyzed sentence or argument and replace some words or phrases by logical constants and others by parameters), when we look at what is practiced under the heading of "logical analysis" by Russell and his followers we see that it is often very far from a piece-for-piece replacement. Logical constants are no longer understood as straightforward proxies for logical expressions of natural language but as parts of a toolbox that we use to articulate logical forms, forms that may be far removed from the surface forms of the sentences and arguments that they formalize.[1]

Given this character of logical analysis, it was no longer clear that the poverty of the grammar of the language of predicate logic, vis-à-vis that of natural languages, should be seen as a problem. This is not to say that

1. What becomes important here are some *criteria* of logical analysis, as in criteria determining what is an acceptable regimentation and what is not; or at least what is a better regimentation than something else. Surprisingly, such criteria are given very little attention in the literature – with the notable exception of Brun (2003). See also Peregrin and Svoboda (2013, 2017).

various ways of enriching it were not still being proposed and considered, but it was not clear that the need for this was really urgent.

Consider a few relatively simple arguments that do not seem to be straightforwardly formalizable by the means of standard predicate logic:

(A11) John is brave
Peter is brave
―――――――――――――
John and Peter share a property

(A12) John is brave
Bravery is laudable
―――――――――――――
John has a laudable property

(A13) John runs quickly
―――――――――――――
John runs.

Take (A11). Its conclusion, at least taken at face value, seems to require predicate variables and quantifiers binding them because the straightforward capturing of its logical form would seem to be

(LFA11)
$$\frac{\begin{array}{l} P(a) \\ P(b) \end{array}}{\exists p(p(a) \wedge p(b)).}$$

This, needless to say, goes beyond the resources offered by the language of first-order predicate logic.

Considering (A12), we see that to regiment it straightforwardly we seem to also need, over and above what is given, a "second-order" predicate; that is, a predicate which forms a sentence with a "first-order" predicate instead of with a subject. Thus,

(LFA12)
$$\frac{\begin{array}{l} P(a) \\ R(P) \end{array}}{\exists p(p(a) \wedge R(p)).}$$

And it seems that nothing stands in the way of extending the language of predicate logic by adding such means.

As for (A13), what seems to make it correct is that we have adverbs that take a predicate to another predicate and that they are "restrictive" in the sense that whatever the new predicate applies to, the old one also did so:

(LFA13)
$$\frac{(R(P))(a)}{P(a)}$$

But, needless to say, the language of predicate logic does not offer us any expression that would take predicates to predicates.

We can certainly extend predicate logic by all of these means (and we will discuss this in detail in the upcoming sections). However, due to the development sketched out previously, we also have an *alternative* possibility: namely, to use the modest means of standard predicate logic in a creative way to account for cases like this in an indirect way. As for the first and second arguments, we may have something like

(LFA11′)
$$\frac{\begin{array}{l}\mathbf{Has}(a,c)\\ \mathbf{Has}(b,c)\end{array}}{\exists x(\mathbf{Has}(a,x)\wedge\mathbf{Has}(b,x)),}$$

(LFA12′)
$$\frac{\begin{array}{l}\mathbf{Has}(a,b)\\ R(b)\end{array}}{\exists x(R(x)\wedge\mathbf{Has}(a,x)),}$$

where **Has** is a logical (?) constant expressing, informally, the relation which connects an individual and a property if the former has the latter. (This, of course, presupposes that we "emulate" predicates by means of subjects.) Providing an alternative, first-order logical form for (A13), we may take an inspiration from Davidson (1967) and assume that what the sentence *Johns runs quickly* talks about is an event of running, the protagonist of which is John and which is quick:

(LFA13′)
$$\frac{\exists e(\mathbf{Ev}(e)\wedge\mathbf{Pr}(e,a)\wedge P(e)\wedge R(e))}{\exists e(\mathbf{Ev}(e)\wedge\mathbf{Pr}(e,a)\wedge P(e)).}$$

10.3 Higher-Order Logic

The examples discussed in the previous section indicate that we might need quantification not only of subjects but also of predicates; and that we also may need predicates predicable of predicates, rather than subjects. And indeed it might seem that there is no deep reason to restrict quantification to subjects, that we may also introduce variables for predicates and let quantifiers bind them.

We get the second-order predicate calculus if we add, to the language of the first-order predicate calculus, variables of each of the category of n-ary predicates ($p^1{}_1, \ldots, p^1{}_n, \ldots, p^2{}_1, \ldots, p^2{}_n, \ldots. p^m{}_1, \ldots, p^m{}_n, \ldots$), plus the rule (where we use p as a metasymbol for predicate variables):

> If U is a quantifier, p a predicate variable, and ϕ a formula, then $Up\phi$ is a formula.

In this way, we will be able to form formulas of the kind of

$$\exists p(p(a) \wedge p(b)),$$

which allows for the articulation of (LFA13) and hence for the straightforward regimentation of (A13).

Another, deeper example of a useful second-order formula would be

$$\forall x \forall y((x = y) \leftrightarrow \forall p(p(x) \leftrightarrow p(y))).$$

This particular formula is a direct regimentation of the so-called *Leibniz principle*, claiming that two objects are identical iff they share the same properties.[2]

Let us consider one more second-order formula which may appear very useful, especially within the context of mathematics, and this is the formula expressing Dedekind's definition of the infinity of a range of items.[3] According to the definition, the range is infinite iff it can be mapped, by a one-to-one function, on its own proper part. The second-order formula that expresses this property of the extension of the predicate P can be articulated in the following way:

$$\exists p(\forall x \exists y(p(x,y) \wedge \forall z(p(x,z) \rightarrow (y = z))) \wedge \forall x \forall y \forall z((p(x,z) = p(y,z)) \\ \rightarrow (x = y)) \wedge \forall x \forall y((P(x) \wedge p(x,y)) \rightarrow P(y)) \wedge \exists y(P(y) \wedge \forall x \forall z(P(x) \wedge p(x,z) \\ \rightarrow \neg(z = y))).$$

To explain the content of this cumbersome formula, notice that the outermost bracket, after the quantifier, $\exists p$, contains four conjuncts. Their respective content is the following:

- the formula $\forall x \exists y(p(x,y) \wedge \forall z(p(x,z) \rightarrow (y=z)))$ states that p is a function (that is, for every x there exists one and only one y so that $p(x,y)$);
- the formula $\forall x \forall y \forall z((p(x,z)=p(y,z)) \rightarrow (x=y))$ states that the function expressed by p is one-to-one (that is, for every y from its range there exists one and only one x so that $p(x,y)$;
- the formula $\forall x \forall y((P(x) \wedge p(x,y)) \rightarrow P(y))$ states that p maps the extension of P on itself;
- the formula $\exists y(P(y) \wedge \forall x \forall z(P(x) \wedge p(x,z) \rightarrow \neg(z=y))$ states there is an object that is P and that is the value of p for no argument.

Having now introduced predicate variables, we might also think about introducing second-order predicates that form sentences not with subjects

2. See Rodriguez-Pereyra (1999).
3. See Dotterer (1918).

but rather with first-order predicates, so that we could also straightforwardly formalize (A14) as (LFA14).

The rules for second-order quantification are straightforward analogues of those for first-order ones; but consider the elimination one

$$\forall p\phi \vdash \phi[P/p].$$

On the right hand side, what we can substitute for the variable p are predicate words. As a matter of fact, we saw that in the predicate calculus we have much more predicate-like expressions – namely, open formulas – that can be taken as "pseudopredicates." If our language contained lambda-abstraction, then everything would be easier as all such "pseudopredicates" could be turned into predicates proper. A variant would be to generalize the elimination rule so that p can be replaced, in the suitable way, by any formula with an appropriate number of free variables.

CLASSICAL SECOND-ORDER PREDICATE LOGIC

Vocabulary:

The same as that of the language of first-order logic adding

N-ary predicate variables (for every n): $p^n{}_1, p^n{}_2, \ldots$;

Delimitation of sentences

like in the first-order case

Syntactic rules:

The same as that of the language of first-order logic adding:

An n-ary predicate term or an n-ary predicate variable is an n-ary predicate

If U is a quantifier, p a predicate variable, and ϕ a formula, then $Up\phi$ is a formula.

Hilbertian calculus

The same as that of the language of first-order logic, adding

$\forall p\phi \rightarrow \phi[P/x]$ $\quad \phi[P/p] \rightarrow \exists p\phi$

$\forall p(\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \forall p\psi)$, where p is not free in ϕ

$\forall p(\phi \rightarrow \psi) \rightarrow (\exists p\phi \rightarrow \psi)$, where p is not free in ψ

$\phi \vdash \forall p\phi$

and replacing the axioms for = by

$\forall x\forall y((x=y) \leftrightarrow \forall p(p(x) \leftrightarrow p(y)))$

Figure 10.3

10.4 Categorial Grammar and Lambda-Categorial Grammar

The idea behind introducing second-order predicates was that we can not only have predicates that form a sentence together with (one or more) subjects, but that we can also have ones that form sentences together with (first-order) predicates. But then we can go further and ask: why cannot we have predicates (or "predicates"?) that form sentences together with still other kinds of expressions – with sentences, for example? And when we pose the question in this way, we can see that in fact we already do have such expressions in the form of sentential operators. A unary sentential operator, for example, is precisely an expression that forms a sentence together with a sentence.

But we can keep on asking: If we have, for every category (or categories) of expressions, a "predicate" which forms a sentence together with the expression(s) of that category (or categories), why could we not have, for every category (or categories) of expressions, a "predicate" which forms an expression of a given category (not necessarily a sentence) together with the expression(s) of that category (or categories)? In particular, when we have syntactic categories X and Y, why cannot we always have a category of expressions, denote it as X/Y, the expressions of which form an expression of the category X together with an expression of the category Y? Or, even more generally, a category $X/Y_1, \ldots, Y_n$, the expressions of which form an expression of the category X together with expressions of the respective categories $Y_1, \ldots, Y_n$? And such considerations lead us to what is usually called *categorial grammar*.[4]

Note that categorial grammar is not a language but a *type* of language. We then get specific languages of this type by fixing the collection of basic categories and the vocabulary. A categorial language based on the basic categories $B_1, \ldots, B_n$ is defined in the following way:

- categories: $B_1, \ldots, B_n$ are categories; and if $X_1, \ldots, X_m$, Y are categories, then so is $Y/X_1, \ldots, X_{m}$;
- vocabulary: there are elementary expressions (words) of some of the categories;
- syntactical rules: if $E_1, \ldots, E_m$, E are expressions of the respective categories $X_1, \ldots, X_m$, $Y/X_1, \ldots, X_m$, then $E(E_1, \ldots, E_m)$ is an expression of the category Y.

The language of propositional logic can be considered as such a categorial language, based on the single category S (of sentences). Unary sentential operators are then seen as expressions of the category S/S, whereas binary ones as those of the category S/S,S.

4. The idea goes back to Ajdukiewicz (1935) and Bar-Hillel (1950). See Oehrle et al. (1988).

We can even make categorial grammar much more flexible if we extend it by the rule of lambda-abstraction that we mentioned previously. Thus, what is called *lambda-categorial grammar*[5] is categorial grammar with variables added to every category and with the following additional kind of rule:

- if E is an expression of the category Y, and $x_1, \ldots, x_m$ are variables of the respective categories $X_1, \ldots, X_m$, then $\lambda x_1 \ldots x_m E$ is an expression of the category $Y/X_1, \ldots, X_m$.

We assume that there is a kind of equivalence defined among expressions of a language of lambda-categorial grammar (in our case it is $\leftrightarrow$), and we further stipulate

$$(\lambda x_1 \ldots x_m E)(E_1, \ldots, E_m) \text{ is equivalent with } E[E_1/x_1, \ldots, E_m/x_m].$$

This is called the rule of lambda-conversion.

Lambda-categorial languages are very general, with a relatively flexible structure. The language of predicate logic can be seen as a lambda-categorial language based on two categories, namely S of sentences and T of subjects. Are there still some languages that do not fit into this framework?

Suppose we have a language with the categories C and C' and we want to have two syntactic rules – R_1, which maps C and C' on C, and R_2, which maps C and C' on C'. To fit R_1 into the categorial framework, we would have to make C' into C/C. And then there would be no way of fitting R_2. What might come to mind is making the framework even more general by simply abolishing the categories and letting any expression be combinable with any other.

If we add the rule of lambda-abstraction to such a relaxed language, what we have is lambda-calculus in its original, type-free form.[6] But the unrestricted generality of languages of this kind may bring about some complications, at least when we want to use them for logical purposes. Suppose we have this kind of language which contains negation, $\neg$, and also logical equivalence, $\leftrightarrow$, which acts as the kind of equivalence among sentences occurring in the definition of lambda-abstraction. Then we can take a variable x and form the expression $\lambda x \neg x(x)$. Call this expression x^* and form the expression $x^*(x^*)$. Given the definition of x^*, it is equivalent to $(\lambda x \neg x(x))(x^*)$, and using the rule of lambda conversion, we get that it is equivalent with $\neg x^*(x^*)$. Thus, $x^*(x^*) \leftrightarrow \neg x^*(x^*)$.

There is a useful way to simplify lambda-categorial grammar – namely, getting rid of all types $Y/X_1, \ldots, X_m$ for $m>1$. The point is that any function that takes more than one argument can be "emulated" by a function taking the arguments one after another; therefore, instead of

5. See Cresswell (1973).
6. While lambda calculus was first introduced as a type-free calculus (Church, 1941; Barendregt, 1980), here we will mostly be interested in its typed-variant (Barendregt, 1992).

an expression E of the category $Y/X_1, \ldots, X_m$, we can have E^* of the category $(\ldots(Y/X_m)/\ldots/X_1)$ such that

$$E(E_1, \ldots, E_m) = (\ldots(E^*(E_1))\ldots)(E_m).$$

Thus, for example, instead of $\wedge$ of the category S/S,S, we will have $\wedge^*$ of the category (S/S)/S so that

$$\wedge(\phi_1, \phi_2) = (\wedge^*(\phi_1))(\phi_2).$$

Then, both kinds of rules constitutive of lambda-categorial grammar can be reduced to the unary case:

- if E, E' are expressions of the respective categories X and Y/X, then $E'(E)$ is an expression of the category Y;
- if E is an expression of the category Y and x is a variable of the category X, then λxE is an expression of the category Y/X.

FIRST-ORDER LOGIC AS A LAMBDA-CATEGORIAL LANGUAGE

Categories of expressions:

S and T are categories

if X and Y are categories, then so is X/Y

Vocabulary:

S/S: $\neg$

(S/S)/S: $\wedge$, $\vee$, $\rightarrow$

S/(S/T): $\exists$, $\forall$

T: $a_1,\ldots; x_1,\ldots$

(...(S/T).../T): $P^n{}_1,\ldots$

(S/T)/T: =

Syntactic rules:

If E, E' are expressions of the respective categories X and Y/X, then $E'(E)$ is an expression of the category Y

If E is an expression of the category Y and x is a variable of the category X then λxE is an expression of the category Y/X

A shortcut:

we write $E(E_1,\ldots,E_m)$ for $(\ldots(E(E_1))\ldots)(E_m)$

Hilbertian calculus

The standard axioms plus

$(\lambda xE')(E) \leftrightarrow E'[E/x]$

Figure 10.4

10.5 New Constants?

We have seen that propositional logic is usually based on the core operators, and that we can add a lot of other ones to them. All these constants, of course, can be imported into predicate logic; so that we will have, for example, modal predicate logic and, for that matter, a lot of modal predicate logics.

Some new problems may arise out of the interaction between the sentential operators and quantifiers. Thus, for example, in the case of modal propositional logic it is not clear what the relationship between formulas such as $\Box\forall xP(x)$ and $\forall x\Box P(x)$ should be. If it is necessary that everything is P, is everything necessarily P? And, conversely, if everything is necessarily P, is it necessary that everything is P?[7]

What can be called the core operators of *predicate* logic are, aside from the core operators of propositional logic, the two quantifiers $\forall$ and $\exists$ (or maybe only one of them, for the other is usually considered as reducible to it) and possibly =. Can we add some new quantifiers to extend this collection? It turns out that a whole lot of quantifiers can be reduced to $\forall$ or $\exists$.

Take, for example, the quantifier *there is exactly one*, which we have denoted as $\exists!$. We saw that Russell rendered it, in effect, as reducible to the standard quantifier in the following way:

$$\exists! x\phi[x] \leftrightarrow \exists x(\phi[x] \wedge \forall y(\phi[y] \rightarrow (x = y))).$$

We can similarly define quantifiers like *there is exactly n*, *there is more than n*, *there is at most n*, etc.[8]

Let us consider a quantifier that is often presented as an example of one that is not reducible to the core ones, the quantifier *infinitely many*. Let us denote it as $\exists_\infty$, so that $\exists_\infty xP(x)$ will be *there exist infinitely many P's*. The thing is that this sentence is not inferable from any finite set of sentences, though it may appear to be inferable from some *infinite* sets. And because in standard predicate logic there is no room for this possibility, no such sentence can exist within its framework.

However, how would we define such a quantifier and how could we delimit its inferential role? It is clear that to introduce it, we would need an infinite number of premises, such as $P(a_1)$, $P(a_2)$, $a_2{\neq}a_1$, $P(a_3)$, $a_3{\neq}a_1$, $a_3{\neq}a_2$, ... But this is what we cannot have – an argument, by our definition, has only a finite number of premises. This shows that a quantifier like $\exists_\infty$ not only cannot be defined within first-order predicate logic, it is hard to imagine that it could be part of any calculus whatsoever.

7. These problems were first discussed by Barcan (1946).
8. See van Benthem (1984).

Can we perhaps think about new logical constants other than quantifiers? We can, so let us consider some examples.

Russell introduced the *iota inversum* operator ɩ, which was meant as a counterpart of the English definite article. It was governed by the following equivalence

$$Q(\iota x P(x)) \leftrightarrow \exists x(P(x) \wedge Q(x) \wedge \forall y(P(y) \rightarrow (x = y))).$$

Maybe more interesting is Hilbert's epsilon operator,[9] which perhaps might be considered as a counterpart of the indefinite article. It is governed by the axiom

$$P(a) \rightarrow P(\varepsilon x P(x)).$$

It is interesting that as soon as we have this constant, we can define both quantifiers in its terms:

$$\exists x P(x) \equiv_{\text{Def.}} P(\varepsilon x\, P(x))$$
$$\forall x P(x) \equiv_{\text{Def.}} P(\varepsilon x \neg P(x)).$$

In view of the fact that, informally, we can read "$\varepsilon x P(x)$" as "some P," we can read the two axioms as "there is a P iff some P is P" and "everything is P iff even some non-P is P," respectively. (While the first one obviously makes good sense, the other reads rather strangely – to understand it, we must realize that the epsilon-calculus is build atop the standard predicate calculus, and hence each of its terms denotes an individual from the universe. If this is the case, then given that all individuals in the universe are P's, even $\varepsilon x \neg P(x)$ is bound to denote a P.)

We have a logical constant that is, syntactically, a predicate, namely =. Are there other predicates that would be candidates for logical constants? One such might be the predicate $\in$ of set theory, which has been traditionally considered as an integral part of logic.

10.6 Truth and Gödel Again

A predicate which may aspire to the status of logical constant is the (unary) predicate of truth. Which are the axioms to govern this predicate? Since the time of Tarski[10] it is commonly accepted that ***Tr*** is a truth predicate in a language L iff for every sentence ϕ of L it holds that (where $<\phi>$ is a name of the formula ϕ):

$$\phi \vdash \boldsymbol{Tr}(<\phi>) \qquad\qquad \boldsymbol{Tr}(<\phi>) \vdash \phi$$

9. See Zach (2003).
10. See Tarski (1932, 1944).

However, as Tarski also shows, this predicate simply cannot be admitted into certain languages. If we take the formalized language of PA, then the predicate ***Tr*** can be part of the language only at the cost of the language being contradictory. To explain this, let us present some results Gödel achieved in the course of the proof of the incompleteness of arithmetic; this will also help us to explain the proof in greater detail.

The first thing that Gödel showed was that we can enumerate the formulas of the language of PA (or, for that matter, any language we artificially produce in terms of definitions); which is to say that we can define an algorithm which assigns, to every formula, a unique natural number so that the formula can be recovered from the number assigned to it. But as the language of PA deals with natural numbers, we can see it as dealing – indirectly – with its own formulas, via their numbers. Thus, if we have an arithmetical operation, which takes the numbers n_1 and n_2 into a number n_3, we can see it as an operation that takes the formulas with the respective numbers n_1 and n_2 to that with the number n_3.

A second thing Gödel showed has come to be called the *fix-point lemma*. It says that if $\psi[x]$ is a formula of arithmetic with the single free variable x, then there is a sentence ϕ_ψ such that (where $<\phi>$ now is a specific name of ϕ, namely its Gödel number)

$$\phi_\psi \vdash \psi[<\phi_\psi>] \qquad\qquad \psi[<\phi_\psi>] \vdash \phi_\psi.$$

It follows that if ***Tr***, governed by the previous axioms, is part of the language of PA, we have the formula $\phi_{\neg Tr}$, such that

$$\phi_{\neg \boldsymbol{Tr}} \vdash \neg \boldsymbol{Tr}[<\phi_{\neg \boldsymbol{Tr}}>] \qquad\qquad \neg \boldsymbol{Tr}[<\phi_{\neg \boldsymbol{Tr}}>] \vdash \phi_{\neg \boldsymbol{Tr}}.$$

But as, at the same time,

$$\phi_{\neg \boldsymbol{Tr}} \vdash \boldsymbol{Tr}[<\phi_{\neg \boldsymbol{Tr}}>] \qquad\qquad \boldsymbol{Tr}[<\phi_{\neg \boldsymbol{Tr}}>] \vdash \phi_{\neg \boldsymbol{Tr}},$$

it follows that $\neg \boldsymbol{Tr}[<\phi_{\neg Tr}>] \vdash \boldsymbol{Tr}[<\phi_{\neg Tr}>]$ and $\boldsymbol{Tr}[<\phi_{\neg Tr}>] \vdash \neg \boldsymbol{Tr}[<\phi_{\neg Tr}>]$; hence, that the language is contradictory. It follows that if the language of PA is not to be contradictory, it cannot contain ***Tr*** – which is what is sometimes called Tarski's theorem.[11]

Let us now return to Gödel's proof, which meant such a turning point in the history of modern logic and which we can now lay out in greater detail. Gödel enters the scene one last time, showing us that arithmetic contains a formula ***Pr***$[x]$ such that it (provably) holds of all numbers of provable formulas;[12] that is, that

$$\vdash \boldsymbol{Pr}[<\phi>] \text{ if} \vdash \phi.$$

11. See Boolos et al. (2002, p. 223).
12. In fact, we consider an improved version of the predicate, due to Rosser (1936), which allows us to get rid of one more assumption that Gödel needed.

Given this, it follows from the fix point lemma that we have the sentence $S_{\neg Pr}$ such that

$$\boldsymbol{\phi_{\neg Pr}} \vdash \neg \boldsymbol{Pr}(<\boldsymbol{\phi_{\neg Pr}}>) \qquad \neg \boldsymbol{Pr}(<\boldsymbol{\phi_{\neg Pr}}>) \vdash \boldsymbol{\phi_{\neg Pr}}.$$

It follows that if $\vdash \phi_{\neg Pr}$, then both

$$\vdash \boldsymbol{Pr}(<\boldsymbol{\phi_{\neg Pr}}>) \qquad \text{and} \qquad \vdash \neg \boldsymbol{Pr}(<\boldsymbol{\phi_{\neg Pr}}>).$$

In a somewhat more complicated way, it is also possible to show that the assumption that $\vdash \neg\phi_{\neg Pr}$ leads to the same result.[13] Hence, both $\vdash \phi_{\neg Pr}$ and $\vdash \neg\phi_{\neg Pr}$ lead to contradiction, and $\phi_{\neg Pr}$ is the sentence that we denoted as ***G*** in §7.2. Hence, if the calculus is consistent, the sentence must be neither provable nor refutable. This is how Gödel showed that in the case of some formalized languages, in particular the language of PA, we cannot attain deductive completeness, which revealed a crucial limitation of logical calculi.

13. See Smith (2013, §25.3).

11 From Calculi to Formal Semantics

11.1 Can We Calculize *Following-From*?

The original idea behind logical calculi was straightforward: to capture, and thus to make calculable, what follows from what. Nobody seemed to have any doubt that this was a task that could be accomplished; perhaps also because it seemed obvious that what followed from what was itself a matter of some tacit rules of inference that we only needed to bring to light. However, this idea was to a great extent devastated by Gödel and Tarski in the 1930s.

We saw that Gödel (1931) showed that however we fine-tune a calculus for the language of PA, we cannot make it deductively complete (unless we make it contradictory). However, it seemed that every sentence of the language must be true or false, and that this is so because it has the meaning it has, which is in turn because this meaning is conferred on it by the axioms of PA. Hence, there seemed to be sentences that follow from other sentences (the axioms) without being inferable from them.[1] And, moreover, this appeared to be a matter of principle, not just a consequence of our imperfect way of setting up the calculus.

Parallel with this, Tarski (1936) came up with a different objection. There may be cases where a sentence follows from an infinite number of other sentences without following from any of its finite subsets. Thus, for example the conclusion that all natural numbers have a property appears to follow from the infinite number of premises assigning the property to each individual number. If this is the case, then inference can never capture *following-from*, for our arguments are bound to have a finite number of premises.

1. The distinction between first-order and second-order predicate logic established in the second quarter of the twentieth century (Moore, 1980) opened up an alternative escape route: claiming that first-order arithmetic is *not categorical*, hence that its semantics is not unique and that therefore there are sentences that are not uniquely true or false. See §15.5.

Let us call a property of infinite sets *compact* if an infinite set can have it only if it contains a finite subset that also has it. Then Tarski's argument appears to indicate that the relation of *following-from* is not compact in the sense that the property *a set of sentences from which a given sentence follows* need not be compact.

This brought Tarski to the idea that to capture *following-from* adequately we need some means more powerful than those which are offered by calculi. And Tarski's idea was that we could perhaps help ourselves to more powerful tools if we managed to logically capture the concept of *truth* and consequently some other concepts more or less related to truth, such as *consequence*, *satisfaction* (which was not so much a pre-theoretical concept as Tarski's own creation), and semantic concepts such as *denotation*.

This step was considered revolutionary since most logicians before Tarski were convinced that such concepts were beyond the direct reach of logic, that the notion of consequence could be at best approximated by the theoretical notion of inference and that the concept of truth could be at best approximated as inference from an empty sequence of premises. Therefore, Carnap characterized his first encounter with Tarski's ideas as "scales falling from his eyes."[2]

We have already seen that the pre-theoretical *following-from* is often understood as truth-preservation, or something close to it. From this viewpoint, capturing it in terms of a calculus or rules of inference might seem like an account that is indirect; whereas an approach that could be based directly on a logical capturing of the concept of truth appeared to promise a direct account, an account where there could not be the gap that Gödel and Tarski discovered in between *following-from* and inference as produced by calculi.

11.2 "Syntax" vs. "Semantics"?

The situation is often depicted in terms of the distinction between syntax and semantics. The problem with calculi, it is often suggested, is that they remain on the level of syntax while the important things in language, including following-from, happen on the level of semantics – so if logic is to produce a really adequate formalization of language, then it clearly cannot but go to the level of semantics.

I am afraid that this story can be so deeply misleading – and in more than one way – that it is crucial to disentangle its complexities. The first of these is the very opposition between the concepts of *syntax* and *semantics* upon which it is based.

Intuitively, the distinction between syntax and semantics is that which stands between the properties that expressions possess merely due to

2. See Coffa (1991, p. 304).

their form and those that they possess due to their "content" (whatever the "content" is supposed to be). When you study a natural language, your study can be divided into (i) the problem of which expressions constitute the language and (ii) the problem as to what their meaning is – (i) is a matter of syntax, while (ii) is a matter of semantics.[3] From this intuitive viewpoint, it seems obvious that if we want to study the correctness of arguments we cannot remain on the level of syntax, for the fact that something can follow from something else only by the force of its meaning appears to go without saying.

Note, however, that on this level the contrast between syntax and semantics is independent of what we think meaning is. Many philosophers think that the meaning of an expression is a matter of what is *represented* by the expression, but there are others who think it is rather a matter of how the expression is *used*; and the concept of semantics hinted at previously is neutral enough to accommodate both of these approaches.

However, the concepts of syntax and semantics that are in play when these terms are used within logic are often not just the intuitive ones mentioned previously but rather their modifications as produced by Carnap. Carnap (1942) took for granted that expressions are *signs* that *represent* their meanings, he therefore defined semantics as the theory of the relationship between expressions and their "designata." In contrast to this, he defined syntax as the theory of relationships among expressions themselves (p. 8):

> An investigation of a language belongs to pragmatics if explicit reference to a speaker is made, it belongs to semantics if designata but not speakers are referred to, it belongs to syntax if neither speakers nor designata but only expressions are dealt with.

The trouble is that seen in this way, following-from, and its theoretical counterparts like inference or consequence, fall on the side of syntax: indeed, they are clear examples of relationships among expressions.

True, we may be able to delimit a relationship coextensive with the relation of following-from in purely syntactical terms without mentioning any semantic property – but this possibility exists for almost any kind of property or relation. We might well be able to delimit, purely syntactically, the set of all (and only) true sentences of a language (we can even imagine that, by a strange coincidence, it will be just those sentences which have some relatively simple syntactic property); and we certainly would not want to say that therefore truth is a syntactic property.

3. See also Peregrin (1999).

From this viewpoint, calculi cannot be seen as syntactic matters: they are models of the relation of following-from, which, by its nature, is not syntactic. Thus, relations that are classified as syntactic from the Carnapian viewpoint may not really be syntactic in the intuitive sense. But there is also the converse problem: if semantics is defined in terms of designata, then it is not clear which relations are truly semantic. Certainly not all relations between linguistic and extralinguistic items are so – but which kinds of entities must expressions be mapped on to make the mapping a *truly semantic* relation?

Imagine some paradigmatically syntactic property, like the syntactic category of an expression. We can certainly consider the categories as objects and define a mapping of expressions on them. This mapping will certainly not amount to anything semantic. Similarly, we can account for a relation of following-from in terms of an assignment of some entities. For example, we can map every sentence on the set of all sequences of sentences from which the sentence follows.

It is clear that this is only following-from in a different guise, and so by doing this we do not cross any boundary between syntax and semantics. This indicates that the Carnapian taxonomy is problematic, both because it can mislead us into classifying semantic relations among expressions (like following-from) as syntactic, and because it can mislead us into classifying as semantic such mappings of expressions on extralinguistic entities that have nothing to do with semantics.

A straightforward way of delimiting *following-from* is defining a relation between sequences of sentences and sentences and a way of generating it from a finite basis. This is the way of a calculus. Alternatively, we can do it in a different way: for example, define a set of mappings of sentences on {0,1} which delimits a relation that preserves the value 1 (in the sense that A follows from $A_1, \ldots, A_n$ iff every mapping of the set which maps all of $A_1, \ldots, A_n$ on 1 also maps A on 1). These two alternative ways may well come to the same result, and there is no reason to assume that one of them is syntactic while the other is semantic. The relation of following-from itself is a semantic matter, while the mechanisms of capturing it are neither syntactic, nor semantic – they are auxiliary tools beyond this classification.

Consider that you have two sentences, *Fido je pes* and *Fido je savec*. You do not understand them (unless, like me, you understand Czech), and thus you do not know whether one of them follows from the other. Now suppose you add designations: *je pes* designates the set of all dogs and *je savec* designates the set of all mammals, hence a superset of the previous one. Now we know that the second sentence follows from the first one. Moreover, we seem to know not only *that* this is the case but also *why* it is the case.

In this way we may come to think that it is only (Carnapian) semantics that allows us to get a real hold on following-from. But this is misleading.

How do we know that the set of dogs is included in the set of mammals? We certainly have not checked the sets (of course we do not know all their members!) – what we know is that

$$\frac{a \text{ is a dog}}{a \text{ is a mammal}}$$

is a valid argument scheme.

How do we know this? Well, we understand the words and hence know their inferential roles. Thus, knowing extensions derives from knowing inferential roles, not vice versa.

So the situation is such that we know what follows from what (for we understand the language), and we try to articulate the corresponding criteria in various ways, in terms of the calculi we discussed in the previous sections or in terms of systems of formal semantics we are going to discuss in the upcoming ones. The difference between these two kinds of articulation has little to do with the distinction between syntax and semantics.

11.3 Calculi vs. Formal Semantics

The switch from calculi, as the basic tool of accounting for following-from, to systems of formal semantics is a significant one. However, this significance can be understood in more than one way.

One way to understand it is in the way that we rejected in the previous section – to see it as a move away from methods based merely on the surfaces of expressions ("syntax") and those based on their meanings ("semantics"). As the semantics of the expressions of our languages is a matter of their links to the entities of the real world (the meaningful expressions being essentially *representations*), and as formal semantics (or model theory, how it is often called - see §13.5) deals with the links between signs and objects, the latter is, thereby, a theory of the former. Consider the description of the situation put forward in a recent book about the philosophical aspects of model theory (Button & Walsh, 2018, p. 3):

> Here is an old philosophical chestnut: *How do we (even manage to) represent the world?* Our most sophisticated representations of the world are perhaps linguistic. So a specialised – but still enormously broad – version of this question is: *How do words (even manage to) represent things?*
>
> Enter model theory. One of the most basic ideas in model theory is that a structure assigns interpretations to bits of vocabulary, and in such a way that we can make excellent sense of the idea that the structure makes each sentence (in that vocabulary) either true or false. Squint slightly, and model theory seems to be providing us

> with a perfectly precise, formal way to understand certain aspects of linguistic representation. It is no surprise at all, then, that almost any philosophical discussion of linguistic representation, or reference, or truth, ends up invoking notions which are recognisably model-theoretic.

On this understanding, there is an essential discontinuity between the "syntactic" methods of calculi and the "semantics" contributed by model theory. It may seem that only with the semantics methods do we get to the heart of the matter.

However, there is also the possibility of understanding it in a much more modest way: as a move from a set of methods with some limitations to some alternative methods that overcome these limitations (though possibly for nontrivial costs). This view is nicely articulated by Shapiro (1998, p. 140):

> [O]ne might claim that models aren't models of anything – they are just artifacts of the logical system. Model theory would be only an elaborate technical device, a tool to study the deductive system. On this view, a model-theoretic semantics only has value to the extent that its consequence relation matches the deductive consequence relation.

This is the understanding we pursue here.

Again, as in the case of calculi, understanding the nature of the systems of formal semantics requires understanding their development and the way this development responded to problems related to accounting for *following-from*. There is, also, an interplay of this newer approach to following-from and the older ones working with calculi, within which some systems of formal semantics may bolster some already existing languages and calculi, whereas others may support languages more or less different from those underlying calculi.

Now, there is also a new kind problem that arises when we have alternative accounts for following-from in the same language. There may be a relation of inference as constituted by a calculus and a relation of consequence induced by a formal semantics, which may or may not coincide. All of this induces new kinds of philosophical problems, which we will tackle in the rest of the book.

12 Truth-Functional Semantics

12.1 Following-From as Consequence

We have already pointed out that one of the most basic intuitive ideas behind the notion of *following-from* is that ϕ follows from Γ iff ϕ is true whenever all elements of Γ are; hence, that following-from, in essence, is truth-preservation. (Though identifying *following-from* with truth-preservation might overgenerate, for we need not consider every ϕ that is true whenever all of Γ are as following from it, for we might require, for example, a relevance of Γ for ϕ, etc.) Thus, a promising approach to *following-from*, alternative to its calculization, might be one that takes this at face value.

Let us, for the time being, identify a language with the set of its sentences, and call a function which maps L on the set of the two truth values a *truth-valuation* of L. Can we explicate following-from in terms of truth-valuations? Note that, in respect to a natural language, it will not be possible to pin down its definite truth-valuation – not only because to know all the truth values would mean to be omniscient, but because the truth values of many sentences are indeterminate or they change quite rapidly. The truth value of a sentence like *It is raining outside* is not determined merely by the semantics of the language and is thus not a subject matter of logic.

Logic, as Frege (1918) stressed, does not focus directly on truth but on the *laws of* truth. Thus, the truth value of a sentence is of interest for logic only insofar as it pertains to the sentence as a matter of "law" – hence, if it is *constant* (the sentence is *necessarily* true or false). More generally, logic is interested in constant *dependencies* of truth values in cases where a sentence always has the same definite truth value when other sentences have certain definite truth values.

All such truth-dependencies can be seen as restrictions on acceptable truth-valuations. From the viewpoint of logic, a language is thus equipped with a delimitation of the space of acceptable truth-valuations. Then we can say that ϕ follows from Γ iff every acceptable truth-valuation that maps all elements of Γ on 1 also maps ϕ on 1.

It is obvious that this approach works very well for classical propositional logic: we can delimit the space of acceptable truth-valuations by means of the well-known truth tables so that it yields the same relation as the standard calculus. Of course, the point of the semantic method is to go *beyond* calculi, which does not happen here. We will have to look at more complicated calculi to see how it fares.

Let us note that even though Tarski is often considered to be the inventor of the kind of "formal semantics" that was yielded by these investigations, by no means was it the case that all the ideas that made up this theory were new. Some of them – for example, the idea that predicates should be seen as denoting functions ranging from the universe to truth values – were, in so many words, already put forward by Frege. Besides this, there was the "algebraic tradition in logic" (Schröder, Löwenheim, Skolem, etc.), the proponents of which tended to assume a kind of "semantic" standpoint quite naturally and who thus solved some important problems of formal semantics before formal semantics was officially on the table. (The shining example, of course, is the Löwenheim-Skolem theorem.)[1]

12.2 Spaces of Acceptable Truth-Valuations

A semantic system $<L,V>$ is a set of sentences plus a set of truth-valuations of the sentences; the elements of the set are called *acceptable* truth-valuations. (Let us note that, as a truth-valuation is a characteristic function of a subset of L, we can, occasionally, identify it with the set it characterizes. Viewed thus, V is a set of subsets of L.) If $S=<L,V>$ is a semantic system, we define

$$\Gamma \models_S \phi \text{ iff } v(\phi) = 1 \text{ for every } v \in V \text{ such that } v(\psi) = 1 \text{ for every } \psi \in \Gamma.$$

If $\Gamma \models_S \phi$, then we will say that ϕ is an *S-consequence* of Γ. S-consequences of the empty sequence are called *S-tautologies*. (Consequence can naturally be considered as a relation between sets of sentences and sentences. However, we can also view it as a relation between *sequences* of sentences and sentences, to make it of the same kind as the inference considered previously. Where no misunderstanding is likely, we will fluctuate between the two views.)

Let us call a set plus a relation of inference over it an *inferential structure*. Then, as it is obvious that $\models_S$ is an inferential relation over L, every semantic system determines an inferential structure, namely the structure $<L, \models_S>$. This structure will be called *the inferential structure of S*. On the other hand, every inferential structure determines a semantic system. We can see this when we consider any argument $\Gamma \vdash \phi$ as excluding certain truth-valuations, namely those which map all elements of Γ on 1

1. See Grattan-Guinness (2000).

while mapping ϕ on 0. Then an inferential structure $<L, \vdash>$ determines the semantic system $<L,V>$ such that V is the set of all those valuations which are not excluded by any argument of $\vdash$.

Given this, there might seem to be a straightforward correspondence between inferential structures and semantic systems. Any inferential structure determines a semantic system and any semantic system determines an inferential structure. Are these relationships inverse to each other, which is to say, is any semantic system determined by its own inferential structure, and is every inferential structure the structure of the semantic system it determines? Unfortunately, the situation is more complicated.

To see this, let us consider the semantic system based on a language that consists of the sentences A and B consisting of the valuations $\{A\}$ and $\{B\}$; hence, the system is $<\{A,B\},\{\{A\},\{B\}\}>$. What is the inferential structure of the system? It is clear that the valid instances of consequence within this system are the following ones:

$$A \models A$$
$$A, B \models A$$
$$B \models B$$
$$A, B \models B.$$

Hence, the structure of the system is the inferential structure whose inference relation consists of the corresponding four arguments:

$$A \vdash A$$
$$A, B \vdash A$$
$$B \vdash B$$
$$A, B \vdash B.$$

It is readily seen that none of these arguments excludes *any* valuation. In other words, this inferential structure does not determine the original semantic system $<\{A,B\}, \{\{A\},\{B\}\}>$ but rather the "full" system $<\{A,B\}, \{\varnothing,\{A\},\{B\},\{A,B\}\}>$. It is also readily seen that extending the inferential relation would not help, for let us list all the arguments that could be added to the structure, and let us list the valuations of S that they would exclude along with each of them:

$\vdash A$	$\varnothing, \{B\}$
$\vdash B$	$\varnothing, \{A\}$
$B \vdash A$	$\{B\}$
$A \vdash B$	$\{A\}$.

We can see that no combination of the arguments is capable of excluding the valuation $\{A,B\}$; also that no combination is capable of excluding all possible truth-valuations of the sentences *A*, *B*, *C*, *D*

A	*B*	*C*	*D*	...
1	1	1	1	...
1	1	1	0	...
1	1	0	1	...
1	1	0	0	...
1	0	1	1	...
1	0	1	0	...
1	0	0	1	...
1	0	0	0	...
0	1	1	1	...
0	1	1	0	...
0	1	0	1	...
0	1	0	0	...
0	0	1	1	...
0	0	1	0	...
0	0	0	1	...
0	0	0	0	...

acceptable truth-valuations of *A*, *B*, *C*, *D* if *C* is to be the conjunction of *A* and *B*

A	*B*	*C*	*D*	...
1	1	1	1	...
1	1	1	0	...
~~1~~	~~1~~	~~0~~	~~1~~	~~...~~
~~1~~	~~1~~	~~0~~	~~0~~	~~...~~
~~1~~	~~0~~	~~1~~	~~1~~	~~...~~
~~1~~	~~0~~	~~1~~	~~0~~	~~...~~
1	0	0	1	...
1	0	0	0	...
~~0~~	~~1~~	~~1~~	~~1~~	~~...~~
~~0~~	~~1~~	~~1~~	~~0~~	~~...~~
0	1	0	1	...
0	1	0	0	...
~~0~~	~~0~~	~~1~~	~~1~~	~~...~~
~~0~~	~~0~~	~~1~~	~~0~~	~~...~~
0	0	0	1	...
0	0	0	0	...

the rows that are struck out are precisely those which are excluded by
($\wedge$E1) $C \vdash A$,
($\wedge$E2) $C \vdash B$,
($\wedge$I) $A, B \vdash C$.

acceptable truth-valuations of *A*, *B*, *C*, *D* if *C* is to be the conjunction of *A* and *B*; and *D* is to be the disjunction of *A* and *C*

A	*B*	*C*	*D*	...
1	1	1	1	...
~~1~~	~~1~~	~~1~~	~~0~~	~~...~~
~~1~~	~~1~~	~~0~~	~~1~~	~~...~~
~~1~~	~~1~~	~~0~~	~~0~~	~~...~~
~~1~~	~~0~~	~~1~~	~~1~~	~~...~~
~~1~~	~~0~~	~~1~~	~~0~~	~~...~~
1	0	0	1	...
~~1~~	~~0~~	~~0~~	~~0~~	~~...~~
~~0~~	~~1~~	~~1~~	~~1~~	~~...~~
~~0~~	~~1~~	~~1~~	~~0~~	~~...~~
~~0~~	~~1~~	~~0~~	~~1~~	~~...~~
0	1	0	0	...
~~0~~	~~0~~	~~1~~	~~1~~	~~...~~
~~0~~	~~0~~	~~1~~	~~0~~	~~...~~
~~0~~	~~0~~	~~0~~	~~1~~	~~...~~
0	0	0	0	...

some of the additional rows that are struck out are those excluded by
($\vee$I1) $A \vdash D$
($\vee$I2) $C \vdash D$,
but there is no way of excluding the penultimate row, for example.

Figure 12.1

∅ without excluding either {*A*} or {*B*}. In other words, no inferential structure determines the system <{*A*,*B*},{{*A*},{*B*}}>.

12.3 The Inferential Structures of Semantic Systems*

The considerations presented in the previous section indicate that the relation between semantic systems and inferential structures that results from associating a structure with the system it determines is many-one: there may be many structures determining the same system; whereas, on the other hand, there are systems determined by no structure. There is, obviously, only one structure that is the structure *of* the system – it is the structure whose relation of inference coincides with the relation of consequence of the system. The fact that there may be more structures *determining* the same system, then, is due to the fact that some arguments do not exclude any valuation, and hence adding them to a structure determining a semantic system produces *a different structure* determining *the same system*. Only when we add all such arguments does the structure become "saturated" and it becomes the structure of the system. Hence, there exist arguments that are "idle" in the sense of not excluding any valuations.

Similarly, the relation that results from associating a system with the structure of the system is many-one, whereas there is only one system that is determined by the structure. Again, additions of some valuations to a system causes no change of the relation of consequence of the system (and hence of the structure of the system), and it is only when all such valuations are present that we gain a system that is "saturated" and hence determined by its own structure. Thus, besides arguments that are "idle" in that they do not exclude any valuations, there are valuations that are "idle" in the sense that their presence/absence does not influence the structure of the system.

Both the "idle" arguments and the "idle" valuations were characterized, in effect, by Hardegree (2005), who wanted to answer the question which systems are determined by their own structures and, conversely, which structures are the structures of the systems they determine. (This is not directly Hardegree's terminology but rather the result of translating his results into our conceptual framework.) Hardegree has shown that the former systems are those that are closed w.r.t. what he called *supervaluations* of all their subsets, and the latter structures are those that are closed w.r.t. certain closure conditions.

A *supervaluation* of a set of valuations V is a valuation v such that a sentence is mapped on 1 by all elements of V if and only if it is mapped on 1 by v. The closure conditions, which characterize those structures that are the structures of the systems they determine, are the following:

$\Gamma \models \phi$ if ϕ is an element of Γ

$\Gamma \models \phi$ if $\Gamma \models \psi$ for every element ψ of an Δ such that $\Delta \models \phi$.

Hardegree's result indicates that a structure is a structure of a semantic system iff it complies with the Gentzenian structural rules. (Here, as we count the premises of an inference as a set, rather than a sequence, two of the rules, namely (Con) and (Perm), are superfluous. The second of Hardegree's rules amounts to (Cut), whereas the first one is a generalized version of (Ref), which follows from the (Ref) via (Exp) and yields (Exp) together with (Cut).) Let us call the arguments that are obtainable from some given ones by means of the Gentzenian structural rules their *structural consequences*.

We can therefore summarize that a valuation is "idle" in the previous sense iff it is a supervaluation of a set of admissible valuations and that an argument is "idle" iff it is derivable, by the structural rules, from already given arguments. One of the consequences of all of this is that not every inferential structure is the structure of a semantic system. Let us call an inferential relation *I truth-preservational* iff there is a semantic system *S* so that $\vdash_I = \models_S$. Then the following holds:

Theorem. An inferential relation is truth-preservational iff it is structural.

Proof: The direct implication is trivial; let us prove the inverse one.

Let $<L,I>$ be an inferential structure and let *I* be structural. A set *C* of sentences of *L* will be called *I-closed* if $\phi \in C$ for every Γ and ϕ such that all elements of Γ belong to *C* and $\Gamma \vdash_I \phi$. Let *V* be the set of all closed sets of sentences of *L*. Then $S = <L,V>$ is a semantic system and we will prove that $\Gamma \vdash_I \phi$ iff $\Gamma \models_S \phi$. The direct implication is straightforward: if $\Gamma \vdash_I \phi$ and all elements of Γ belong to *U* for some $U \in V$, then, as *U* is closed, $\phi \in U$. So we only have to prove the inverse implication.

Hence, let $\Gamma \models_S \phi$. This means that whenever $U \in V$ and all elements of Γ belong to *U*, $\phi \in U$; that is, $\phi \in U$ for every *U* such that (i) *U* is closed, and (ii) all elements of Γ belong to *U*. As before, let Γ^* be the set of all elements constituting the sequence Γ and let $Cn(X)$ be the smallest closed set containing *X*. As *I* obeys (Ref), $\Gamma^* \subseteq Cn(\Gamma^*)$. As it obeys (Cut), $Cn(Cn(\Gamma^*)) = Cn(\Gamma^*)$. This means that $Cn(\Gamma^*)$, in the role of *U*, satisfies (i) and (ii), and hence $\phi \in Cn(\Gamma^*)$. Thus, $\Delta \vdash_I \phi$ for some sequence Δ all of whose members belong to Γ. Due to the (Ext) and (Con), this means that $\Delta' \vdash_I \phi$ for some sequence Δ' with the same elements as Γ and hence, due to the (Perm), $\Gamma \vdash_I \phi$.

If we want a moral to be drawn from this, we may consider: *as inference is truth-preservation, it must be structural*. (This would move any substructural logic on a sidetrack.) However, there is an alternative moral that might be considered: *insofar as inference is structural* (*which we saw is the case, for example, for inference capturing proving in mathematics*), *nothing prevents us from seeing it as truth-preservation*. (More precisely, we can see it as a preservation of something, and we can call this something *truth*.) But we can of course also reject the whole construal of inference as preservation and, together with it, (some of) the structural rules.

Note the following point. We have stressed that for a relation of inference generated by a calculus we do not necessarily have a decision procedure, in the sense that we would be able, given an Γ and a ϕ, to decide whether ϕ is inferable from Γ. Yet the relation is determined at least in this weaker sense: we can generate a list of all and only its instances, that is, a list such that if ϕ is inferable from Γ, then $<\Gamma, \phi>$ will appear, sooner or later, on the list. Note that if a relation of inference is generated by a semantic system, it need not be determinate even in this weak sense. This might raise the suspicion that the possibility of getting hold of *following-from* directly, without the mediation of a calculus, may come at a price (which need not be a bargain).

12.4 Truth-Functional Logic

There is a distinguished set of sentential operators – those which can be seen as directly manipulating truth values and can be characterized by the notorious truth tables. An *n*-ary operator can be called *truth-functional* if the truth value of any output sentence which arises out of the combination of the operator with *n* input sentences is uniquely determined by the truth values of the input sentences.

What is remarkable – though it is quite well-known, for this is one of the first things a student of logic learns – is that there is a version of the core operators of propositional logic such that all of them are truth-functional. What is even more remarkable (though no less well-known) is that this semantics fits with the calculus of classical propositional logic, in the sense that the formulas which are valid w.r.t. the truth-functional semantics are precisely the theorems of classical propositional logic. This is what has raised classical logic on the pedestal of an exceptional logical system.

We have already pointed out that though Tarskian systematic formal semantics only entered the scene much later than the first calculi, some semantic ideas were an integral part of the doctrine of modern logic from its very beginning. Thus, it was not the case that there were calculi of classical logic that were only later found to align with truth-functional semantics; the idea that the operators should work truth-functionally were certainly already present in the mind of many of those who developed the calculi.

We should stress that the calculus of classical logic and truth-functional semantics fit together in the sense that the theorems of the former coincide with the tautologies of the latter – it is not so that the calculus would delimit the semantics in the sense of §12.2. Indeed, the rules of the calculus do not – and cannot – exclude all non-classical valuations; in particular, they do not exclude, for example, all valuations that map a sentence and its negation both on 0 (we saw they cannot be excluded by any arguments). We will return to this in the next section.

From the conceptual viewpoints, truth-functional semantics can be understood in more than one way. One way is to see it as not literally a *semantics* (not really trading in *meanings* or their approximations) and to see it just as an apparatus capturing how truth values propagate from simpler to more complex sentences. This is possible because every complex sentence of a propositional language contains some sub-sentence(s) and we can consider the truth value of the former as determined by the latter (mediated by the operator which binds the sub-sentences together).

Viewed in this way, truth-functional semantics is extraordinarily perspicuous; and the fact that classical propositional logic has this semantics provides, for example, for its decidability.

However, we can construe truth-functional semantics as a *semantics* in the more literal sense of the word: we can see truth values as "denotations" of sentences, where the "denotations" are taken as explications or regimentations of meanings, or of some relevant part or level thereof. We can also then assign "denotations" to the sentential operators, namely the corresponding truth-functions. This lets us make the denotation-assignment compositional in a very neat way: if ϕ is a sentence consisting of the sub-sentences $\phi_1, \ldots, \phi_n$ bound together by the operator O, we have (where $\|E\|$ is the denotation of E):

$$\|\phi\| = \|O\|(\|\phi_1\|, \ldots, \|\phi_n\|).$$

This kind of semantics is standardly called extensional, for Frege called the truth value of a sentence its *Bedeutung* ("meaning"),[2] and Carnap later changed this to *extension*.[3]

12.5 Inference vs. Consequence

The definition of a semantic system is a matter of determining the acceptable truth-valuations; and a calculus, we saw, can be considered as a means for the delimitation of such a space.[4] Going semantic should now mean becoming able to delimit spaces of truth-valuations that cannot be delimited by means of calculi.

If we start to talk about truth-valuations explicitly, are we now able to specify *every* truth-valuation and *every* space of truth-valuations? Certainly not. We cannot present the whole infinite set of sentences and stipulate which value is assigned to each of them. We can still only talk about valuations and spaces of valuations that can be described by some finite means. Thus, a valuation that is a matter of a wholly haphazard assignment of truth values to sentences is still beyond our ken; and so are many sets of truth-valuations.

So even if we go semantic, we can still delimit sets of truth-valuations only by means of some finite restrictions: especially restrictions of the kind of *if certain sentences have such and such truth values, then a certain sentence is bound to have such and such truth value*. Writing $|\phi|$ for

2. See Frege (1892b).
3. See Carnap (1947).
4. See Peregrin (2010a), see also Garson (2013).

the truth value of ϕ, the restrictions we use when we define the semantics for classical propositional logic may look like the following:

$$\text{if } |\phi| = 1 \text{ and } |\psi| = 1, \text{ then } |\phi \wedge \psi| = 1.$$

In general, such a restriction has the form

$$\text{if } |\phi_1| = t_1 \text{ and} \ldots \text{and } |\phi_n| = t_n, \text{ then } |\phi| = t,$$

where $\phi_1, \ldots, \phi_n$ are sentential parts of ϕ.

Note that this indeed lets us go beyond what we can do in terms of arguments, for arguments can be seen, from this viewpoint, as special cases of such restriction, in which $t_1 = \ldots = t_n = t = 1$.

There are semantic rules that are directly in this shape and which can thus be directly transformed into arguments. This is the case of the previous rule for the conjunction, which can be directly seen as the argument

$$\phi, \psi \vdash \phi \wedge \psi.$$

Besides them, there are semantical rules that are not directly in this shape, but can be transformed into it. This is the case of, for example,

$$\text{if } |\phi| = 0, \text{ then } |\phi \wedge \psi| = 0,$$

which is equivalent (insofar as the truth value of every sentence is either 1 or 0) to

$$\text{if } |\phi \wedge \psi| = 1, \text{ then } |\phi| = 1$$

and can thus be captured as the argument

$$\phi \wedge \psi \vdash \phi.$$

But there are also rules that cannot be brought into the shape in which they would amount to arguments. We saw an example concerning negation:

$$\text{if } |\phi| = 0, \text{ then } |\neg\phi| = 1.$$

How does this square with the fact that the semantic delimitation of the classical propositional calculus, which contains these rules, is equivalent to the delimitation in terms of the calculus? The fact is that the equivalence does *not* mean that the spaces of acceptable truth-valuations delimited by the semantics and by the calculus are identical. There is, for example, no way of excluding all the valuations violating the previous semantical rule for disjunction in terms of a calculus – from the viewpoint of the calculus, some such valuations are acceptable.[5]

Despite this, the relation of consequence as delimited by the semantics coincides with the relation of inference as specified by the calculus – hence, the difference between the two spaces remains "under the surface." This means that, as far as propositional logic goes, the advance enabled by switching from calculus to semantics is not substantial. But this is okay, for the limitations of calculi come to the surface only at the level of predicate logic.

5. See, e.g., Raatikainen (2008).

13 Extensional Denotational Semantics

13.1 Truth-Valuational Semantics for Predicate Logic?

We can have a truth-functional semantics fitting together with a calculus of propositional logic, namely with classical propositional logic. (Of course we might prefer another calculus of propositional logic for which there is no such semantics; but the possibility of building a propositional logic purely truth-functionally is here.) Is it possible to extend the truth-functional notion of logic to the predicate variety?

We know that it is, as the formal semantics for the language of predicate logic tends to be an integral part of logic courses. However, the step from the truth-functional semantics of propositional logic to this semantics is not trivial. The obvious trouble is that while, in the case of propositional logic, we can always consider the truth value of a non-atomic sentence as determined by the truth values of its parts (mediated by a logical operator, as a complex sentence always consists of an operator plus a sentence or sentences), the situation is different in the case of predicate logic. In the language of predicate logic, a quantified sentence need not contain any (closed) sentences as its parts – hence, its truth value cannot be generally seen as determined by those of its sub-sentences.

It would be perhaps possible to consider a quantified sentence of the shape $Ux\phi[x]$ as containing, as its ("virtual") parts, all sentences of the shape $\phi[\ldots]$. This would be of a piece with the understanding of quantifiers as generalized operators: the general quantifier as a generalized (possibly infinitistic) conjunction and the existential quantifier as a generalized (possibly infinitistic) disjunction. And it would bring about all the problems of the approach: its infinitistic nature and also the substitutional notion of quantification that it engenders.

Another possibility is to give up the idea that truth-functional semantics can restrict itself to studying truth values and their propagation from simpler to more complex sentences, and to instead switch to a more general theory of denotation. In this case, we can go beyond truth values and consider a richer repertoire of entities that might be eligible as denotations of expressions of the language of predicate logic. What we need is for the truth values of sentences to not necessarily be determined by the truth values of their parts but rather by their denotations. And as we consider denotations, instead of merely truth values, we can do what Tarski proposed to do: we can take the truth values of at least some sentences to be determined not only by the denotations of their parts, and not even only by the denotations of any other expressions of our language, but directly by the objects which are eligible as (though in fact they need not be) denotations of the expressions.

We can see this Tarskian move as consisting of three relatively independent upgrades on truth-functional semantics. We start from conditions of the form

(*) if $|\phi_1|=t_1$ and . . . and $|\phi_n|= t_n$, where $\phi_1, \ldots, \phi_n$ are parts of ϕ, then $|\phi|=t$;

where $t_1, \ldots, t_n, t$ are truth values. The first upgrade, which corresponds to the already discussed way of taking quantifiers as infinitistic operators, is that of releasing the "are parts of" restriction of (*) and going infinitistic; that is, instead of (*) we have something like

> if the truth values of all sentences of the shape *s* are such and such, then $|\phi|=t$;

for example,

> if all the sentences of the shape $P(\ldots)$ are true, then $|\forall xP(x)|=1$;

or, somewhat more sloppily,

> if $|P(a)|=1$ for every subject a, then $|\forall xP(x)|=1$.

The second upgrade on (*) is a step from truth values to denotations. Instead of considering just sentential parts of ϕ, we consider more general parts, assuming that each expression has an object as its denotation. If we write $\|E\|$ for the denotation of E, we can have

> if $\|E_1\|=d_1$ and . . . and $\|E_n\|= d_n$, where $E_1, \ldots, E_n$ are parts of ϕ, then $|\phi|=t$.

Thus, we can, for example, have

> if $\|a\|$ is an individual and $\|P\|$ is a set containing the individual, then $|P(a)|=1$.

Now consider combining the two upgrades, in that we plug the second one into the first one in the following way:

> if $\|a\|$ is an individual and $\|P\|$ is a set containing the individual for every subject a, then $|\forall xP(x)|=1$;

that is,

> if $\|a\|$ is an element of $\|P\|$ for every subject a, then $|\forall xP(x)|=1$.

This can be further reformulated as

> if o is an element of $\|P\|$ for every denotation o of a subject, then $|\forall xP(x)|=1$.

Now enter the third upgrade, by which we consider not only actual, but also potential denotations, thus breaking the bounds of our current language:

> if o is an element of $\|P\|$ for every potential denotation o of a subject, then $|\forall xP(x)|=1$.

Given that all the potential denotations of subjects make up what we call the *universe* (*of discourse*), we have

> if o is an element of $\|P\|$ for every element o of the universe, then $|\forall xP(x)|=1$.

Let us now have a closer look at the individual upgrades. We have already considered the first one, that is, taking quantifiers as infinitistic operators. The second one leads us to abandoning the idea that the truth values of complex sentences are necessarily determined by those of their sub-sentences, instead of which we assume that the truth value of any sentence is determined by the denotations of its sub-expressions. This is obviously no minor upgrade, because it necessitates the assumption that aside from every sentence having a truth value, every expression has a denotation; and hence it presupposes a theory of denotation. Also the third upgrade is substantial – it leads us to abandoning the whole idea that the truth values of sentences are determined by the semantic properties of other expressions. We now assume that it is directly some constellations of (potential) denotations that make some sentences true or false.

Let us start fixing the terminology. Just as sentences were mapped onto truth values in truth-valuational semantics, here expressions are, more generally, mapped on *denotations* (some of which may be truth values), where each category of expression is associated with a type of denotation (subject expressions with individuals, predicates with properties, etc.). While the assignment of truth values was called a truth-valuation, the assignment of denotations is called an *interpretation*. (As this is a notion of interpretation different from our interpretation$_{\mathrm{M}}$, we will denote it as *interpretation*$_{\mathrm{F}}$ – but as in this and in the next two chapters we will deal exclusively with this notion, we will leave out the subscript.) And just like the semantics of a language, on the level of truth-theoretical semantics, was fixed by delimiting the range of its acceptable truth-valuations, here the semantics of a language is fixed by delimiting the range of *acceptable interpretations*.

13.2 The Universe as the Bunch of Everything

Despite an obvious parallel with truth-functional semantics for the language of propositional logic, it seems that the denotational semantics for the language of predicate logic is of a different nature. Making the move sketched out in the previous section seems to turn the whole enterprise of the semantic attitude to logic into something incomparably more complex – and especially philosophically more convoluted – than it was until we were able to make do with truth-functions. To assume that sentences are true/false and to investigate the dependencies of their truth values is a straightforward, philosophically not overly committed, enterprise. However, once we assume that there are domains of individuals, of properties, etc., we make the whole enterprise incomparably more involved and less transparent.

The point is that the truth-functional account, though it is presented as an alternative to the "syntactic," Hilbertian, or Gentzenian calculi, can be seen again as a kind of calculus. Indeed, it is easy to see the truth values 0 and 1 as mere expedients of a mechanism singling out (necessarily) true sentences or valid arguments, differing from the mechanisms of the calculi only as to their complexity, not as to their nature. Now if we turn to the richer semantics involving individuals, properties, etc., we seem to be switching to something really different; we seem to be engaging no mere expedients but rather some "real" entities of the extralinguistic world.

The usual version of semantics for predicate logic is simplified by the trick introduced by Frege (1892a) in the form of the identification of properties and relations with *functions* mapping *n*-tuples of individuals on truth values. (This really does look like a trick – for it would seem to imply that to know a property we would need to know exactly which individuals possess it, which does not seem realistic.) Then we can take the truth value of an atomic sentence to be determined by the denotations of its parts: to take it as the function value of the application of the denotation of its predicate to those of its subjects. And we can take the truth value of a quantified sentence as determined by whether all/some individuals belong to a relevant set. In this way, the only new entities that we need to build a semantics, aside from truth values, are individuals. (But, still, individuals seem to be "out there" in the world, and basing our semantics on them appears to give it a suspiciously empirical dimension.)

Thus, the core of the new semantics is the universe, the set of all individuals that are potential denotations of the subjects and the entities to which the predicates potentially apply; hence, it is important to see its nature clearly. If I say *Every prime is odd*, then what I am talking about are primes, and more generally numbers, and it would make little sense to evaluate it with respect to a universe in which numbers are not present. And if what we are after is the actual truth value of the sentence, it would also make little sense to evaluate it with respect to a universe in which *not all* primes are present. So what should be in our universe?

An almost self-evident answer is "everything." But here we have to ask *what is everything*? and we are likely to face a couple of problems.[1] We may be certain that, say, a table is an object in the universe, but is this also true of each square centimeter of its surface? Does the universe contain abstract, ideal, and fictional entities, or just entities occupying spatiotemporal regions of our actual world? Does it contain all its subsets, that is, all the sets of the things it contains? (If so, and if we agree to the Cantorian proof that a power set of a set is always bigger than the set itself, it would seem that the universe must be "bigger than itself.")[2]

Given these problems, logicians usually consider not one huge, all-encompassing universe but instead see the universe to be somehow "context-dependent," to always be given somehow *ad hoc* according to what we want to talk about in a given language. (Thus, the language of PA may be associated with the universe of all and only natural numbers; while the language of zoology could perhaps be associated with the universe of all and only living organisms.)[3]

13.3 The Universe as a Structure

As was indicated in the previous section, it is often taken for granted that denotational semantics is about the real world, and that the relationships between the expressions of artificial languages and their denotations somehow depict, imitate, or regiment the relationships that, in fact, exist within the real world between expressions of natural languages and their meanings or referents. However, it is important to realize that the actual practices of logic show us the semantic approach in a rather different light.

First, as logic is concerned with logical forms and hence with logical, rather than extralogical, expressions, it cares neither about the nature of the universe nor about the actual links between subject and predicate expressions and their denotations. It just assumes that every subject denotes *some* individual and every predicate denotes *some* property or relation, regardless of which one it is and regardless of what exactly an individual is. Moreover, the abandonment of the idea of a unique, all-encompassing universe in favor of opportunistic, partial ones lets us rest

1. Some of them are discussed in Rayo and Uzquiano (2006).
2. See, e.g., Grim (1991).
3. The historical background of switching from the all-encompassing universe to a lot of opportunistic ones is described by Goldfarb (1979).

content with the assumption that the individuals are taken from *some* universe, regardless of its nature. Thus, we should perhaps replace the claim that we are especially interested in sentences that are true w.r.t. any acceptable assignment of individuals of the universe to subjects and relations over the universe to predicates, with the claim that we are interested in those that are true w.r.t. any acceptable assignment of individuals from *any* universe (or perhaps any *nonempty* universe[4]). The universe thus comes to figure in our accounts in a merely ghost-like manner.

Moreover, for all a logician cares, any job that can be done by an interpretation can be equally well done by any other interpretation that is isomorphic with it. (Two interpretations i and i' of the language of predicate logic are isomorphic iff there is a one-to-one mapping m of the universe of the first one on that of the second one such that $i'(s) = m(i(s))$ for every subject s; and $<o_1, \ldots, o_n>\in i(P)$ iff $<m(o_1), \ldots, m(o_n)>\in i'(P)$ for every predicate P.) This indicates that as far as logic is concerned, our universe can be composed of any kind of objects whatsoever, the only relevant thing is the "structure" of the interpretation – what is invariant across isomorphic interpretations.

Quine (1968) envisaged this in terms of his notion of "proxy function." Suppose, he suggested, that we have an ordinary interpretation of a language, according to which the subjects of our language refer to some objects like people, animals, houses, etc., and the predicates of the language refer to the relations among the individuals in the sense of sets of n-tuples of the objects. Now suppose I replace every individual by its "proxy" so that a subject now refers to the proxy of its previous referent and a relation holds among objects iff they are proxies of the objects among which it held previously. Suppose, for example, that the proxy of an object is its shadow;[5] hence, *Barack Obama* now refers to the shadow of Barack Obama and *Trump Tower* refers to the shadow of Trump Tower; and *is higher than* relates the shadow of Trump Tower to the shadow of Barack Obama iff the former is a shadow of an object higher than that which casts the latter shadow. Everything that could interest us on the language in question stays the same: the truth of its sentences, and hence the consequence relation, logical truth, etc.

Quine thus concluded that in most cases (when we are considering a denumerable universe) we could restrict our attention to a single universe, such as that which is constituted by natural numbers. (Quine calls such a stance, half-jokingly, "ontological Pythagoreanism.") Any kind of universe that we could desire to work with could be transformed into this one via a suitable proxy function.

4. $\exists x(P(x)\vee\neg P(x))$, for example, is a theorem of the standard calculus of classical predicate logic. (The reason is that it can be derived from $P(x)\vee\neg P(x)$, which is an instance of (EM), and $(P(x)\vee\neg P(x))\rightarrow\exists x(P(x)\vee\neg P(x))$, which follows from $\forall x((P(x)\vee\neg P(x))\rightarrow\exists x(P(x)\vee\neg P(x)))$, which in turn is equivalent with $\exists x(P(x)\vee\neg P(x))\rightarrow\exists x(P(x)\vee\neg P(x))$.

5. An example presented by Davidson (1979).

Let us, moreover, notice that while the universe of a specific theory (like PA) is usually thought of as determining the truth values of relevant sentences, the current view may lead us, as it did Quine, to an inverse view. We may see the universe as not determin*ing*, but rather determin*ed* by, the values. The universe of a theory contains those and only those objects that are needed to make the sentences of the theory true. Or, as Quine (1953, p. 13) vividly puts it: "a theory is committed to those and only those entities to which the bound variables of the theory must be capable of referring in order that the affirmations made in the theory be true." Note that this view on the concept of the universe basically challenges the idea that by including the universe in our semantics we make the semantics rest on specific entities of the real, extralinguistic world.

Thus, there is a sense in which even denotational semantics may be seen as something on a par with a calculus – not as a matter of interconnection of language with actual extralinguistic things, but as a complex apparatus of expedients that allow us to specify the relation of consequence and the set of tautologies.

13.4 Formal Semantics for Predicate Languages

An interpretation of the language of (first-order) predicate logic, we saw, is an assignment of the elements of a set U (the universe) to subjects and of the elements of the sets $B^{U\times \dots \times U}$ (of all functions mapping n-tuples of elements of U on elements of B, where B = {0,1} is the set of the two truth values) to predicates. Such an interpretation allows us to compute the truth values of all closed atomic formulas ($\|P(a_1, \dots, a_n)\| = \|P\|(\|a_1\|, \dots, \|a_n\|)$) and consequently of any combination of such formulas by means of the sentential operators.

But we still have variables, open formulas, and quantified formulas. To assign denotations to them, we need to consider the assignment of objects to variables, which we will call a *valuation*. Hence, a valuation is simply a mapping of the set of all variables on U; the set of all valuations will be denoted as V. Given a valuation, a variable denotes the object which the valuation maps it on. Moreover, given an interpretation and a valuation, we can compute denotations also for open atomic formulas (and, as a consequence, of all combinations of atomic formulas by means of the sentential operators).

As for the quantified formulas, their denotations are determined in the expected way. The value of $\forall x\phi$, $\|\forall x\phi\|$, with respect to an interpretation and a valuation v is 1 iff $\|\phi\|$ is 1 with respect to the interpretation and any valuation which differs from v at most in the value assigned to x; while the value of $\exists x\phi$, $\|\exists x\phi\|$, with respect to an interpretation and a valuation v is 1 iff $\|\phi\|$ is 1 with respect to the interpretation and at least one valuation which differs from v at most in the value assigned to x.

Note that though logical operators and quantifiers are often treated as "syncategorematic" in the sense that they are not assigned their own denotations, the denotations for them are not difficult to find; indeed, if we treat all expressions of the language of the predicate calculus as having denotations, the whole picture becomes more transparent for we can see the formal semantics as an algebraic structure similar to that of the corresponding language, and we can see the semantic interpretations as something like homomorphisms of the former into the latter.

It is already clear that we can take logical operators as obtaining denotations from B^B, or from $B^{B\times B}$, in particular the well-known truth-functions; given this, we have the denotations of logically complex formulas determined by their parts in a transparent way, such as, for example, $\|\phi\wedge\psi\|=\|\wedge\|(\|\phi\|,\|\psi\|)$. The situation is a little bit more complicated with quantifiers, for they interact with variables in a way which does not seem to point to any such transparent semantic treatment. But we have seen that if we engage the rule of lambda-abstraction, we can treat quantifiers as taking unary predicates to sentences and hence obtaining denotations from $B^{(B^U)}$. Then it will be the case that, for example, $\|\forall(P)\| = \|\forall\|(\|P\|)$.

Let us define the semantics for the language of predicate logic exactly, first in the traditional way in which logical constants are considered syncategorematic. An *interpretation* in a universe U is a function mapping every subject on an element of U and every *n*-ary predicate on an *n*-ary relation over U (an element of $B^{U\times\cdots\times U}$). A *valuation* is a function mapping every variable on an element of U. The satisfaction of a formula by an interpretation and a valuation is defined so that

- i and v satisfy an atomic formula $P(a,b,\ldots)$ iff the function interpreting P yields 1 if applied to the values of $a,b,\ldots$, where the value of a subject is either its interpretation or its valuation, depending on whether it is a constant or a variable;
- satisfaction is extended to the logically complex sentences in the obvious way (i and v satisfy $\neg\phi$ iff they do not satisfy ϕ, etc.);
- i and v satisfy $\forall x\phi$ iff i satisfies ϕ together with $v[x\rightarrow o]$ for every $o\in U$, where $v[x\rightarrow o]$ is identical with v with the single possible exception that it maps x on o.

Alternatively, we can define interpretation in the following way: A denotation assignment based on an interpretation i and a valuation v is a function mapping every expression of the language on a denotation so that

- the denotation of a constant is its interpretation;
- the denotation of a variable is its valuation;
- the denotation of an atomic formula is the value of the denotation of the predicate applied to those of the subjects;
- the denotation of a logically complex formula is the value of the denotation of the operator applied to the denotations of the subformulas;
- and the denotation of a quantified formula is the denotation of the quantifier applied to the function that maps an element o of U on the

denotation which the formula would have if we changed the current valuation of the variable, which is subject to the quantification, to *o*.

An interpretation which maps a sentence on 1 is called a *model* of the sentence and similarly for a set of sentences or a theory (insofar as the theory is identified with a set of sentences). Every interpretation, and hence every model of a theory, determines a universe and a collection of relations over the universe (namely those which are the function values of the interpretation for some predicates). This is an algebraic ("relational") structure, which we will call *model structure* (sometimes it is simply called a *model*, which, however, is misleading as it introduces an ambiguity in the term).

SEMANTICS FOR THE LANGUAGE OF PREDICATE LOGIC

In terms of satisfaction

i,v sat $P(a,b,...)$ iff $i(P)(||a||,||b||...) = 1$, where $||a|| = i(a)$ if a is a constant and $||a|| = v(a)$ if a is a variable, etc.

i,v sat $\neg\phi$ iff not i,v sat ϕ

i,v sat $\phi\wedge\psi$ iff i,v sat ϕ and i,v sat ψ

i,v sat $\phi\vee\psi$ iff i,v sat ϕ or i,v sat ψ

i,v sat $\phi\rightarrow\psi$ iff i,v sat ψ or not i,v sat ϕ

i,v sat $\forall x\phi$ iff $i,v[x\rightarrow o]$ sat ϕ for every $o\in$U

i,v sat $\exists x\phi$ iff $i,v[x\rightarrow o]$ sat ϕ for at least one $o\in$U

In terms of denotation

$||E||_{i,v} = i(E)$ if E is a subject term or a predicate term

$||E||_{i,v} = v(E)$ if E is a variable

$||\neg||_{i,v}$ is the element of B^B such that $||\neg||_{i,v}(t) = 1$ iff $t=0$

$||\wedge||_{i,v}$ is the element of $B^{B\times B}$ such that $||\wedge||_{i,v}(t_1,t_2) = 1$ iff $t_1=1$ and $t_2=1$

$||\vee||_{i,v}$ is the element of $B^{B\times B}$ such that $||\vee||_{i,v}(t_1,t_2) = 1$ iff $t_1=1$ or $t_2=1$

$||\rightarrow||_{i,v}$ is the element of $B^{B\times B}$ such that $||\rightarrow||_{i,v}(t_1,t_2) = 1$ iff $t_1=0$ or $t_2=1$

$||\forall||_{i,v}$ is the element of $B^{(B^U)}$ such that $||\forall||_{i,v}(m) = 1$ iff $m(o)=1$ for every $o\in$U

$||\exists||_{i,v}$ is the element of $B^{(B^U)}$ such that $||\forall||_{i,v}(m) = 1$ iff $m(o)=1$ for at least one $o\in$U

$||P(a,b,...)||_{i,v} = ||P||_{i,v}(||a||_{i,v},||b||_{i,v},...)$ for every n-ary predicate P and subject terms $a,b,...$

$||O\phi||_{i,v} = ||O||_{i,v}(||\phi||_{i,v})$

$||\phi O\psi||_{i,v} = ||O||_{i,v}(||\phi||_{i,v},||\psi||_{i,v})$

$||Ux\phi||_{i,v} = ||U||_{i,v}(f)$ where f is the function that maps an element o of U on the value $||\phi||_{i,v[x\rightarrow o]}$

Figure 13.1

13.5 Model Theory

The kind of semantics we have just presented is often also dubbed *model-theoretic*. Indeed, the terminology to which logic has converged is such that the two most significant areas of its interest are proof theory (which deals with calculi) and model theory (which deals with systems of formal semantics). However, the term *model theory* may be misleading and it deserves some explanation.

We have seen that the term *model* entered logic especially via the theory of interpretation and satisfaction that originated with Tarski (1936). A model, we saw, is an interpretation satisfying a given sentence or a set of sentences (a theory). Later, Tarski, together with his collaborators and followers, developed model theory into a mathematically complex theory, taking in the results achieved within the previous "algebraic tradition" of logic, as well as those of Gödel and others.

However, instead of starting from a theory and looking for the interpretations (and their model structures) that are its models, we can also assume a converse view and start from a model structure or a set of model structures trying to find a theory of which they are models, and especially a theory of which they are the only models and which thus *determines* them in this sense. This perspective goes back to the so-called "American postulate theorists"[6] and to thinkers who were more interested in algebra than in logic.[7]

The discipline of model theory[8] thus found itself in the intersection of mathematical logic and what has come to be called *universal algebra*.[9] Some of its protagonists consider the way the adjective "model-theoretic" is employed in logic (namely for everything plus minus of a semantic nature) to be a misnomer.[10]

13.6 Generalization

The extensional semantics for the language of the first-order predicate logic can be easily generalized. We have seen that, in the case of first-order predicate calculus, we can make do with individuals plus the truth values, namely with the domains B and U, for the denotations of

6. See Scanlan (2010).
7. Like Birkhoff (1935) or Mal'cev (1941).
8. See, e.g., Chang and Keisler (1973) or Hodges (1993).
9. See Cohn (1981).
10. Thus, Hodges (2016, p. 174), comments on the claim of Prawitz (1974, p. 66), "In model theory, one concentrates on questions like what sentences are logically valid and what sentences follow logically from other sentences," in the following way: "I can say with absolute confidence that I never met a model theorist who 'concentrates on questions like what sentences are logically valid and what sentences follow logically from other sentences.' "

predicates (and also for those of logical operators and quantifiers) can be built from individuals and truth values. It would seem, then, that we can consider more complex denotations of the same kind and, consequently, richer languages based on these denotations. We can, for example, have functions from properties (which are functions from individuals to truth values) to truth values, which can serve as denotations of second-order predicates. Thus, this kind of semantics can be built for predicate logics of higher-orders.

Introducing second-order predicates as denoting functions from the denotations of first-order predicates (namely functions from individuals to truth values) to truth values, we can notice that quantifiers can be seen as specific cases of such predicates. A quantifier, we saw, can be taken as taking a "pseudopredicate" (or a predicate proper, if we make use of lambda-abstraction) to a sentence. The general quantifier produces a true sentence iff applied to a predicate denoting the function mapping every individual on truth; the existential one produces a true sentence iff applied to a predicate denoting a function mapping at least one individual on truth.

Similarly, we can introduce third-order predicates as denoting functions from denotations of second-order predicates to truth values; and generally predicates of any order. But we can even think about further generalization of this kind of semantics. Which kinds of functions, in general, can be built from individuals and truth values? The general answer is as follows. Let U be the universe and B be the set of the two truth values. Then U and B are *domains*. Moreover, if $D_1, \ldots, D_n, D$ are domains, then the set $D^{D_1 \times \cdots \times D_n}$ of all n-ary functions from $D_1, \ldots, D_n$ to D is a domain. Now we can have a category of expressions for each of the domains: the set T (of subjects, taking denotations from U) and the set S (of sentences, taking denotations from B) are categories; moreover, if $C_1, \ldots, C_n, C$ are categories, then there is a set $C/C_1, \ldots, C_n$ which is a category (taking denotations from $D^{D_1 \times \cdots \times D_n}$). And if we add rules for all the possible applications,

> if $E_1, \ldots, E_n, E$ are expressions of the respective categories $C_1, \ldots, C_n$, $C/C_1, \ldots, C_n$, then $E(E_1, \ldots, E_n)$ is an expression of the category C,

we can see that what we reach is precisely the language of categorial grammar.

In this way, the generalization of formal semantics for predicate logic yields us a semantics that fits very well with this maximally general logical language, making all its rules into functional applications so that the denotations of all complex expressions are the function values of the application of the denotations of one of their parts to those of the other parts:

$$\| E(E_1, \ldots, E_n) \| = \| E \| (\| E_1 \|, \ldots, \| E_n \|).$$

We will see later that after making a semantic sense of the rule of lambda-abstraction, we can have the semantics for the whole lambda-categorial grammar (see §10.4).

13.7 The "Expressive Power" of Higher-Order Logic

We have already seen in §10.3 that the first-order predicate calculus is readily extensible to a second-order, or even a higher-order one. Now we saw that this extension comes naturally even from a semantic viewpoint, and that we can naturally extend predicate logic all the way to categorial grammar, offering us a very flexible toolbox for the regimentation of natural language. Should we thus simply forget about first-order predicate logic and embrace this much more general system?

We also saw that formulas of second-order logic can be, with some ingenuity, transformed into those of first-order logic, and the same may hold more generally for sentences of lambda-categorial grammar. It thus seems that perhaps everything that can be regimented in terms of lambda-categorial grammar can already be regimented by means of the first-order predicate logic. True, while lambda-categorial grammar allows us to carry out the regimentation in a very straightforward manner, first-order predicate logic requires invoking the "art of Russellian logical analysis" and leads us to formulas that are very different from the regimented sentences. However, the austerity and transparency of the means we thus restrict ourselves to may be something to value.

Hence, should we go for a direct and transparent regimentation using a baroque artificial language, or rather for a very indirect one using an austere and transparent language? Before taking a stance on this problem, we need, however, to take into account one more way in which the step beyond first-order logic enhances our "expressive power."

Let us return to Dedekind's definition of infinity, which we briefly considered in §10.3:

$$\begin{aligned}&\exists p(\forall x\exists y(p(x,y)\wedge\forall z(p(x,z)\rightarrow(y=z)))\wedge\forall x\forall y\forall z((p(x,z)=p(y,z))\\&\rightarrow(x=y))\wedge\forall x\forall y((P(x)\wedge p(x,y))\rightarrow P(y))\wedge\exists y(P(y)\wedge\forall x\forall z(P(x)\wedge p(x,z)\\&\rightarrow\neg(z=y))).\end{aligned}$$

It is clear that like other second-order formulas it too can be transformed into a first-order one, perhaps in the following way

$$\begin{aligned}&\exists u(\forall x\exists y(\mathbf{Val}(u,x,y)\wedge\forall z(\mathbf{Val}(u,x,z)\rightarrow(y=z)))\wedge\\&\forall x\forall y\forall z(\mathbf{Val}(u,x,z)\wedge\mathbf{Val}(u,y,z))\rightarrow(x=y))\wedge\\&\forall x\forall y(((P(x)\wedge\mathbf{Val}(u,x,y)))\rightarrow P(y))\wedge\exists y(P(y)\wedge\\&\forall x\forall z(P(x)\wedge\mathbf{Val}(u,x,z)\rightarrow\neg(z=y))),\end{aligned}$$

where **Val** is a ternary predicate (intuitively relating a function, now understood as an object, and an argument to the value of the function for the argument). Is it so that P satisfies this formula iff it has an infinite number of instances? Unfortunately, unlike in the second-order case, it is generally *not* the case. We will see why shortly.

Return to the case of attributing a property to a property, for which we seem to need second-order properties. Suppose we take properties as (specific cases of) objects (or entities, if you wish), which hence fall into our universe of discourse, and we have a logical (?) binary predicate **Has** (see §10.2) which relates an object to a property iff the former has the latter. The logical form of *Charles and Peter share a property* would then be

$$\exists x(\mathbf{Has}(a,x)\wedge\mathbf{Has}(b,x)).$$

Having properties within the universe, along with individuals, might seem implausible. However, we can do away with this by changing our predicate logic to a two-sorted one, which is a not very problematic variant of predicate logic with two separate universes (and two separate sorts of subjects, and arguments of predicates marked as to from which of the universes they are to come). Such two-sorted logic is easily translatable into a one-sorted one, in that we introduce two predicates delimiting the two parts of a common universe. Then the previous formula may get amended as

$$\exists x(\mathbf{Prop}(x)\wedge\mathbf{Has}(a,x)\wedge\mathbf{Has}(b,x)).$$

This is what leads us to the question: do we *really* need second-order logic? Cannot we make do with its first-order emulation that we just sketched?

Return to the definition of infinity. The criterion of Dedekind, which this definition is to articulate, is that a set is infinite iff there exists a one-to-one mapping of the set on its proper subset. This is what is exactly articulated by the second-order formula. In the case of the first-order counterpart of the formula, if it is true, then there exists such a mapping and the extension of P is indeed infinite. However, if the formula is false, can we be sure that the extension of P is finite?

In this case, we understand functions as objects in the universe. However, as we know, the universe of a first-order theory is arbitrary, it need not contain anything (save for the fact that it is nonempty, that is, that it contains at least one object). Hence, how can we be sure that it contains objects corresponding to all possible functions? And if it does not, how can we be sure that if the formula is false, then it is because there is no function mapping the extension of P on a proper part of itself, or only because such a function fails to be included in our *arbitrary* universe?

This might seem to be a surmountable problem. After all, when we formalize arithmetic, logic alone does not guarantee us that the universe will contain the infinite sequence of natural numbers we need to do arithmetic; we must secure this by means of axioms. So it would seem that here again we need an axiom securing that our universe contains all the functions. So what we need, it would seem, is something like the following axiom

$$\forall P \exists x \forall y \forall z \mathbf{Val}(x,y,z) \leftrightarrow P(y,z).$$

But alas, this is a *second-order* axiom! Thus, to be able to use it within first-order logic, we would need to turn it into a first-order one, perhaps

$$\forall u \exists x \forall y \forall z \mathbf{Val}(x,y,z) \leftrightarrow \mathbf{Val}(u,y,z).$$

But this is a tautology saying that for every function which already is within our universe there is the corresponding function in our universe. What we would need is to quantify over not only the functions that are in the universe but over all functions whatsoever.[11]

This means that such an axiom could do the required work only if it is second-order, and hence the definition of infinity is also adequate only if it is second-order or if it is supported by something that is.

This gave rise to the discussion between proponents of second-order logic, who maintained that without it we are not able to apply logic to the foundations of mathematics (where we cannot make do without a firm grasp on the concept of infinity), and their opponents, who claimed that second-order logic (with standard semantics) is no logic at all, for it lacks any calculus. The discussion is nicely anatomized by Shapiro (1991).

In this way, second-order logic, as compared to the first-order one, brings us a new kind of enhancement of "expressive power." In this case, it is not so that it brings us a richer repertoire of syntactic categories and/or a broader spectrum of syntactic rules to regiment those of natural language, as this kind of "expressive power" is purely semantic. There seem to be concepts, especially the concept of infinity, which cannot be (directly) captured except in second-order logic.

11. See Shapiro (1991, Chapter 4) for more details.

14 Denotations Beyond Extensions

14.1 Modal Propositional Logic

The route that has taken us from purely truth-functional semantics to a denotational one has done so by way of the accommodation of quantification and the engagement of the universe of individuals. There is also a completely different possible route, the route we take when we want to move to modal propositional logic; and it leads via the accommodation of modalities and via the engagement of the "universe" of possible worlds.

The problem with truth-functional semantics, when we want to accommodate modalities such as *possibly* or *necessarily*, is that these cannot be taken to denote truth-functions. The basic trouble is that, in general, the truth value of $\Box A$ is not uniquely determined by that of A: if A is false, then $\Box A$ is bound to be false too; but if A is true, then $\Box A$ may in some cases also be true and in other cases false. From this viewpoint, it would seem that it might be enough to split the truth value *truth* (1) into the values *necessary truth* (11) and *contingent truth* (01). Then we can say that $\Box A$ is true iff A is necessarily true and it is false otherwise:

A	$\Box A$
11	11
01	0
0	0

However, having done this refinement, we would have to rewrite the truth tables for the other operators for the new values. In particular, we would have to say whether it is contingent truth or necessary truth on which negation maps falsity. And it seems that the only reasonable way to do it leads via the splitting of falsity. Once this value (0) is split up into contingent falsity (00) and necessary falsity (10), we can say that the negation of contingent falsity is contingent truth and that of necessary falsity is necessary truth:

A	$\neg A$
11	10
01	00
00	01
10	11

The table for necessity can then be easily amended accordingly:

A	$\Box A$
11	11
01	10
00	10
10	10

But this is far from the end, for now we must rewrite the tables for conjunction, disjunction, etc. Take disjunction – what will the table for it look like with the new values? It is clear that the disjunction of necessary truth with anything will yield necessary truth; and that the disjunction of necessary falsity with any value will yield the latter value. It is also clear that the disjunction of a truth with anything will be a truth and that the disjunction of two falsities will be a falsity. What remains unclear, however, is the precise value of the disjunction of two contingent values:

$A \vee B$		B			
		11	**01**	**00**	**10**
A	**11**	11	11	11	11
	01	11	?1	?1	01
	00	11	?1	?0	00
	10	11	01	00	10

Take two contingent truths: it is clear that their disjunction will be a truth, but will it be a necessary one or a contingent one? Intuitively, it would seem that this value is again not determined uniquely. The disjunctions of some kinds of contingent truths will be true only contingently, whereas some other kinds (for example, the disjunctions of a contingent truth with its negation) will be true necessarily.

Therefore, what we obviously need, if we want to reach a compositional valuation, is a further refinement – we must split up contingent truth (and similarly contingent falsity) into still more values. But this time the situation is rather tricky: we cannot say that we simply split contingent truth into two values, one such that its disjunction with contingent truth will yield contingent truth, whereas the other will yield necessary truth. The idea rather is that the values of some pairs of sentences the disjunction of which is true are "complementary" (their disjunction yields necessary truth), whereas those of others are not (their disjunction yields contingent truth). As a result, we gain a more or less nontrivial Boolean algebra of values, which can be represented as the well-known algebra of sets of "possible worlds" (possibly with the superstructure of an "accessibility relation").[1]

Hence, what we arrive at in this way is the solution proposed in Kripke's seminal papers:[2] we enrich our semantics by entities which have come to be called *possible worlds*. These are entities which the truth of sentences is relative to. Sentences thus become true w.r.t. individual worlds; they are true *simpliciter* if they are true w.r.t. the world which is the "actual" one. And they are necessarily true in a world w if they are true either in all worlds altogether, or in some worlds that are "distinguished" from the viewpoint of w.

What are the possible worlds? Is there a way of reducing them to something simpler? Well, there is: we can identify them with acceptable truth-valuations of sentences. But though this is in some respects natural, there are also problems with such an identification. First, it makes worlds language-dependent. Second, it leads to a not very "well-behaved" modal logic, a logic proposed, in effect, by Carnap, and known as "C."[3] If we, on the other hand, leave possible worlds as primitive entities, as Kripke did, we can easily help ourselves to many different kinds of modal logic and choose among a variety of potential explications of necessity and possibility.

Another option is to see possible worlds as generated by a basic set of individuals, and a set of some elementary properties, as all possible distributions of the properties among the individuals. (This is a method of, for

1. See Peregrin (2006b).
2. See Kripke (1963a, 1963b, 1965b).
3. See Schurz (2001) or Punčochář (2012).

example, Tichý, 1988) This, however, leads to possible worlds sharing the same universe of individuals, which may have some counterintuitive consequences.[4] Another possibility might be to look at possible worlds as mere expedients of rendering the truth of some sentences as dependent on "external factors." Thus, Stalnaker (1986, p. 117): "What is a possible world? It is not a particular kind of thing or place; it is what truth is relative to, what is the point of rational activities such as deliberation, communication and inquiry to distinguish between."

Notice that when we moved from propositional to predicate logic, we were able to remain on the *extensional* level – denotations of sentences could be identified with their truth values, so that the denotation assignment merely extended the truth-valuation to other categories of expressions; here we dissociate the denotations of sentences from their truth values. Thus, the denotation assignment ceases to be extensional, and it is usually called *intensional*.

Note that in this way we make denotations of sentences into something notably closer to the meanings of the sentences (in the intuitive sense of the word). As long as denotations of sentences are truth values, they can hardly be considered as nontrivial approximations of their *meanings* (and the same goes for other kinds of expressions): we would certainly not want to claim that all true sentences are synonymous.

14.2 Intensional Logic

Philosophers and logicians who were convinced that the explication of the intuitive concept of meaning is an integral part of the mission of logic, notably Carnap, were, long before Kripke, pondering the possibility of an "intensional logic," which would provide the means for such an explication.[5] An obvious idea was that the truth value of a sentence may be relative to something ("circumstances"), and that its meaning is not a truth value but the way in which its truth value depends on this something. Hence, the idea of "possible worlds" and intensions as functions constituted by their help had been in the air for a relatively long time;[6] and so, when Kripke presented his semantics, it could also be seen as the realization of the plans of Carnap and others.

Note that the simplest way of moving from extensional to intensional semantics is such that what we add to the basic building blocks of the

4. For example, as our actual world contains Donald Trump, there cannot be a possible world not containing this individual. The individual need not be named "Donald Trump"; in fact, it need not share a single property with our Trump (it can have, for example, all the properties of our current Barack Obama). It will still, however, be *the same* individual (in some obscure metaphysical sense).
5. See Carnap (1947).
6. See Copeland (2002).

semantics are just possible worlds. Hence, just like all extensions were functions built atop of the sets U and B, all intensions may be thought of as functions built atop of the sets U and B and a set W of possible worlds. A basic category *C* which was associated with the domain *D* will now be associated with the domain D^W: thus, the denotations of sentences will no longer be truth values but rather functions from possible worlds to truth values, and those of subjects will be no longer elements of the universe but rather functions from possible worlds to the universe.

The situation will be more complicated in the case of non-basic categories. If we have, for example, the category C_1/C_2, such that C_1 and C_2 are basic and their respective domains now are $D_1{}^W$ and $D_2{}^W$, then its domain will have to be $((D_1{}^W)^{(D_2{}^W)})^W$. Take unary sentential operators, which, on the level of extensions, received denotations from B^B. They take sentences to sentences, so now, on the level of denotations, they take the new denotations of sentences (functions from possible worlds to truth values) to the new denotations of sentences. Moreover, the denotation of such an operator cannot be just such a function (an element of $(B^W)^{(B^W)}$, for they should also be intensions, functions from possible worlds – so their denotations should now be elements of $((B^W)^{(B^W)})^W$.

It seems then that having switched from extensional to intensional logic we will have to completely rebuild its semantics, even throwing away denotations of expressions that we had already taken for adequate on the level of extensions. Take a core sentential operator like negation. We may feel that its semantics is fully determined by its extension – the truth-function. But now it will have to be assigned a denotation from $((B^W)^{(B^W)})^W$, which seems to be an unwanted complication.

Fortunately, the intensions of those expressions which we take to be in this sense "extensional" can be produced out of their extensions in a straightforward way. First, though intensions are functions from possible worlds, they may be *constant* functions – hence, the denotations of expressions like negation may have the same value in every possible world (Kripke, 1972, would call such expressions "rigid designators").

But this only solves a minor problem. The major one is that the value in a possible world is now not to be the truth-function (an object of the domain B^B), but rather something more complicated (an object of the domain $(B^W)^{(B^W)}$). But there is a simple way of transforming the good old extension of such an expression – that is, in the case of the negation of the truth-function – into the required object. If $\|\neg\|_E$ is the original extension of $\neg$ (an object from the domain B^B), we can define the required function from the domain $(B^W)^{(B^W)}$) so that it maps an element $\|A\|$ of B^W on an element $\|\neg A\|$ of B^W so that for every $w \in W$

$$\|\neg A\|(w) = \|\neg\|_E(\|\neg A\|(w)).$$

Let us generalize this idea, let us call this kind of operation *quasiapplication*: the result of the quasiapplication of an object o of the domain $(D^{D_1 \times \ldots \times D_n})^{\mathrm{W}}$ to objects $o_1, \ldots, o_n$ of the domains $D_1^{\mathrm{W}}, \ldots, D_n^{\mathrm{W}}$ will be an object f of the domain D^{W} (a function from possible worlds to objects of the domain D) such that $f(w) = (o(w))(o_1(w), \ldots, o_n(w))$. Hence, if we have a categorial grammar – that is, a language all the syntactic rules of which correspond, on the semantical level, to applicative rules – we can make it (trivially) intensional straightforwardly so that all the rules come to be quasiapplicative.

In this way, we can transport all the denotations which we have on the extensional level and with which we are satisfied to the intensional level. But, of course, the point of going intensional is that we can also have denotations that are not just extensions refashioned as intensions. The point is that now we can have *genuine* intensions, that is, such that they have no counterparts among extensions.

Let us illustrate this by means of the sentential operator of necessity, which, we saw, could not be captured on the extensional level. Like $\neg$, $\Box$ too is to denote an element of $((\mathrm{B}^{\mathrm{W}})^{(\mathrm{B}^{\mathrm{W}})})^{\mathrm{W}}$, and it too can be taken as a constant function. However, there is no $\|\Box\|_{\mathrm{E}}$ so that

$$\|\Box A\|(w) = \|\Box\|_{\mathrm{E}}(\|A\|(w)).$$

Let us call 1^w the function from B^{W} that maps every world on 1, and let us call 0^w that which maps every world on 0. Then we can define $\|\Box A\|$ in the following way:

$$\begin{aligned} \|\Box A\| &= 1^w \text{ iff } \|A\| = 1^w \\ &= 0^w \text{ iff } \|A\| \neq 1^w. \end{aligned}$$

In this way, we will also be able to assign denotations to other expressions that were problematic from the viewpoint of extensional semantics, and we will be able to assign better denotations to those that were not treated in a satisfactory way within extensional semantics. (Thus, for example, the extension of a binary predicate is a relation between individuals, which makes it an unsatisfactory explication of the meaning of a verb like *seek* – for if you seek something, there need not be any such individual.[7] A better denotation would connect you not to an extension, but to an intension.)

Technically, we can handle the shift from extensional to intensional semantics in either of two ways. The first possibility is to enrich the

7. The sentence *John seeks a unicorn* is one of the most famous examples discussed by Montague (1973).

syntax of the language by a category I of expressions corresponding to the domain W (usually containing only variables), in which case it keeps being a categorial grammar, only with three basic types instead of two (it is sometimes called two-sorted type theory, for the basic type S somehow does not count;[8] I call it "globally intensional" logic – see Peregrin, 2006a.). What becomes more complicated in this case is the relationship between this kind of artificial language and the natural language it is used to regiment: it is no longer the category S but rather the category S/I that corresponds to sentences, unary predicates are regimented by expressions of the category (S/T)/I, and subjects by those of the category T/I. As a result, it is also no longer the applications but rather the quasiapplications that correspond to the most basic rules of natural language syntax: if *P* is the expression regimenting a predicate of a natural language sentence and *a* its subject, then the whole sentence will be regimented not by $P(a)$ but rather by the quasiapplication of *P* to *a*, namely what can be articulated, in terms of lambda-abstraction, as $\lambda w((P(w))(a(w)))$.

The other possibility is to leave the shift from extensional to intensional semantics maximally hidden in the semantics. That is, we keep the syntactic part of the language almost as it is, only we change the types of "intensionally sensitive" predicates: hence, we keep, for example, the binary predicate *find* as of the type S/T,T (or (S/T)/T), while we change the type of *seek* to S/T,(T/I) (or (S/T)/(T/I)) to account for the fact that the object position of the latter creates an intensional context. As regards the semantics, we change the values from extensions to intensions and change applications to quasiapplications accordingly. (We may even keep claiming that the "standard" semantic values of expressions are extensions, treating intensions only as "asides" and introducing a mechanism that promotes the intension of an expression temporarily to its extension. This is the way of Montague, 1974, which results in what I call "locally intensional" logic – see Peregrin, 2006a.)

The fact that we can accommodate intensional semantics in a language that differs from the extensional lambda-categorial grammar merely in that it contains one more basic category raises obvious doubts: how can we turn an extensional language into an intensional one in this trivial way? However, we must distinguish two things. The language in itself, that is, taken as a self-contained artificial language, certainly does not cease to be extensional. However, it can be used for the purposes of intensional analysis of natural language if we do not regiment the sentences of natural language as expressions of the category S but rather of the category S/I, and similarly for other categories of expressions.

8. See Gallin (1975).

14.3 The Semantics of Lambda-Abstraction

On the most general level, a semantic interpretation is just a homomorphism of the algebra of syntax into the algebra of denotations. True, the simpler and the more perspicuous the algebra of denotations is, the better – the easier it is to work with. Therefore we try to build the infinite system of domains of this algebra atop of a small set of basic domains. And we try to make the rules of the semantic algebra as simple as possible.

We have seen that one of the ways of making the rules simple is making them functional applications. This could have been accomplished, without a remainder, for propositional logic on the extensional level. However, already in the case of predicate logic there is a problem: quantifiers are standardly understood as variable-binding devices and corresponding rules cannot be made into applications.

In the most perspicuous case, all the variable-binding was concentrated into the rule of lambda-abstraction and all the other rules were applicative (including the rules for quantifiers); however, it seemed that at least this (kind of a) rule cannot be squeezed into the applicative framework. Therefore, what was probably the richest type of extensional language of logic, the lambda-categorial grammar, was based on applicative rules *plus the rule of lambda-abstraction*. What is the denotational counterpart of this rule?

Consider a simple example. We have a unary predicate P, applied to a variable x: $P(x)$. On the extensional level, P denotes a function from individuals to truth values, x denotes an individual, and hence $P(x)$ denotes a truth value. Now the rule of lambda-abstraction takes x and $P(x)$ to the unary predicate $\lambda xP(x)$, which we expect to denote the function identical with that denoted by P. But how can we get this function, the denotation of $\lambda xP(x)$, from an individual and a truth value, the denotations of its parts?

A way out of this is to reconsider the denotations of variables and open formulas. A variable denotes the object that the actual valuation assigns to it (just as a subject denotes the object which the actual interpretation assigns to it). Thus, denotations are relative to interpretations plus valuations. But maybe valuations should not be seen as going hand in hand with interpretations, maybe an interpretation should leave the valuation of variables open. This can be done so that we take the denotation of any expression as a function from valuations of variables: in the case of a constant it will be a constant function, assigning the constant the same value for every valuation; in the case of a variable it will be the function taking us from a valuation to the value of the variable according to this valuation; and in the case of a complex expression, the value of which was taken to result from the application of the denotation of one of its parts to those of the other parts, we now replace the application by a quasiapplication.

In this way, we can make use of the same trick as in the case of accommodating intensions. Hence, if we have the expression $E(E_1, \ldots, E_n)$, where $E_1, \ldots, E_n, E$ are of the categories $C_1, \ldots, C_n, C/C_1, \ldots, C_n$, then $\|E(E_1, \ldots, E_n)\|$ will no longer be $\|E\|(\|E_1\|, \ldots, \|E_n\|)$, it will be the function such that $\|E(E_1, \ldots, E_n)\|(v) = (\|E\|(v))(\|E_1\|(v), \ldots, \|E_n\|(v))$ for every valuation v. This is because expressions of the categories $C_1, \ldots, C_n$, $C/C_1, \ldots, C_n$ no longer obtain denotations from the domains $D_1, \ldots, D_n$, $D^{D_1 \times \cdots \times D_n}$; they rather obtain them from the domains $D_1{}^{\mathrm{V}}, \ldots, D_n{}^{\mathrm{V}}$, $(D^{D_1 \times \cdots \times D_n})^{\mathrm{V}}$. Similarly, $\|E\|$ is no longer an object of the domain D but rather an object of D^{V}.

Once we amend the semantics in this way, we can make a semantic sense of lambda-abstraction. Return to $\lambda x P(x)$. Notice that now the denotation of P is not an element of B^{U}, but rather of $(\mathrm{B}^{\mathrm{U}})^{\mathrm{V}}$, and that of x is an element of U^{V}. Now, the denotation of $P(x)$ will be, according to the definition, the function $\|P(x)\|$ from B^{V} such that $\|P(x)\|(v) = (\|P\|(v))(\|x\|(v))$. Hence, as $\|x\|(v)$ is $v(x)$ and $\|P\|(v)$ is a constant function, we have $\|P(x)\|(v) = \|P\|_{\mathrm{E}}(v(x))$, where $\|P\|_{\mathrm{E}}$ is the constant extension of $\|P\|$. Thus, the value of $\|P(x)\|$ for a valuation v is 1 iff the object assigned to x by v belongs to the extension of P.

The application of the rule of lambda-abstraction to x and $P(x)$, yielding $\lambda x P(x)$, can be seen, on the semantic level, as taking us from the objects $\|x\|$ and $\|P(x)\|$ of the respective domains U^{V} and B^{V} to the object $\|\lambda x P(x)\|$ of the domain $(\mathrm{B}^{\mathrm{U}})^{\mathrm{V}}$, so that $(\|\lambda x P(x)\|(v))(o) = \|P(x)\|(v[x \rightarrow o])$,[9] as taking us from a valuation v to the function from B^{U} which assigns, to an element o of the universe, the value which $\|P(x)\|$ would have if x were to denote o. More precisely, let $v[x \rightarrow o]$ be the valuation that is just like v with the single possible exception that it maps x on o; then $(\|\lambda x P(x)\|(v))(o) = \|P(x)\|(v[x \rightarrow o])$ for every $o \in \mathrm{U}$ and every $v \in \mathrm{V}$. A rule that has this kind of semantics can be called *abstractive*, and we can see that we could consider the rules of our semantic algebra as quasiapplicative and abstractive ones.

The semantics we have sketched assumes that the language we are working with contains variables. How does it square with the claim we made previously that variables are dispensable? Can we produce a version of lambda-categorial grammar without variables?

We can dispense with them by, for example, a generalization of Quine's method of predicate functors. And in fact, this kind of generalization already existed before Quine proposed his method.

14.4 Combinatory Logic

The idea of the so-called *combinatory logic*, which can be seen as a variable-less version of lambda-calculus, goes back to Schönfinkel (1924), and it was developed into a fully-fledged logical system by Curry et al. (1958).

9. Obviously here $\|\lambda x P(x)\|$ is computed out of $\|P(x)\|$ and not $\|x\|$, but rather x itself; but x and $\|x\|$ are interdefinable, as not only $\|x\|$ is the only f such that $\forall v(f(v) = v(x))$, but also x is the only y such that $\forall v(\|x\|(v) = v(y))$.

Remember the idea behind Quine's "predicate functors" (§9.2): the idea of replacing variables, as they are standardly employed by the predicate calculus, by a set of predicate-transforming operators, so that, in effect, λxF, for every formula F, becomes equivalent to $\lambda xP(x, \ldots)$ for a predicate P. And as now lambda always binds the *first* variable of a predicate, we can simply write λP. Now, if we are not in the language of the predicate calculus but in the richer language of categorial grammar, the Quinean functors can be directly incorporated into it. According to Quine, we assume that for every n we have

- an operator $\mathbf{Pad}^n$ which maps an n-ary predicate on an $(n+1)$-ary one so that $(\mathbf{Pad}^n(P))(x_0, x_1, \ldots, x_n) = P(x_1, \ldots, x_n)$
- an operator $\mathbf{Perm}^n$ which maps an n-ary predicate on an n-ary one so that $(\mathbf{Perm}^n(P))(x_1, \ldots, x_n) = P(x_1, x_3, \ldots, x_n, x_2)$
- an operator $\mathbf{Refl}^n$ which maps an $(n+1)$-ary predicate on an n-ary one so that $(\mathbf{Refl}^n(P))(x_1, \ldots, x_n) = P(x_1, x_1, \ldots, x_n)$.

Now we can assume that we have such a triple of functors not only for every category of n-ary predicates but rather, more generally, for every "predicative" category $Y/X_1, \ldots, X_n$.

In view of the fact that, as we saw, within lambda-categorial grammar we can make do with only unary "predicates" (more generally, only with types Y/X instead of $Y/X_1, \ldots, X_n$), we can understand $x(x_1, \ldots, x_n)$ as a shortcut for $(\ldots((x(x_1))(x_2))\ldots(x_n)$ (thus x being of the type $(\ldots((Y/X_n)/X_{n-1})/\ldots/X_1)$. And we can see that we can reduce the predicate functors to more primitive terms. Then, for example,

$$
\begin{aligned}
&(\mathbf{Pad}^{(\ldots((Y/X_n)/X_{n-1})/\ldots/X_1)}(x))(x_0, x_1, \ldots, x_n) \\
&\qquad = ((\mathbf{Pad}^{Y/X_1}(x))(x_0, x_1))(x_2, \ldots, x_n).
\end{aligned}
$$

In this way, we could reach a basis of a few kinds of operators, like $\mathbf{Pad}^{Y/X_1}$, which would be able to do all the work of Quinean predicate functors; moreover, not merely with respect to predicates in the narrow sense but for predicates in the sense of expression of all "predicative" types (that is, of the types of the shape X/Y). In this way, we can transform lambda-categorial grammar into a variable-less shape.

Combinatory logic went for this goal from the very beginning and thus reached a more succinct basis of "predicate functors," now called *combinators*, than those we would end up with when we took the Quinean detour. Imagine we have a closed lambda term

$\lambda xE.$

Now E can either be a variable, a constant, an expression of the form $E'(E'')$, or an expression of the form $\lambda x.E'$. In the first case, as the term is closed, E must be identical with x. In this case, the term is identical with the combinator $\boldsymbol{I}^X$ (where X is the category of E), which is defined so that

$$\boldsymbol{I}^X(E) = E \qquad \text{(hence } \boldsymbol{I}^X \text{ is } \lambda xx\text{).}$$

In the case where E is a constant, the term is identical with $\boldsymbol{K}^{X,Y}E$, where $\boldsymbol{K}^{X,Y}$ is a combinator such that

$$(\boldsymbol{K}^{X,Y}(E))(E') = E \qquad \text{(hence } \boldsymbol{K}^{X,Y} \text{ is } \lambda x\lambda yx\text{).}$$

In the case when X is $E'(E'')$, we can use the fact that

$$\lambda xE'(E'') = \lambda x(((\lambda x E')(x))((\lambda xE'')(x))),$$

and make the term identical to $(\boldsymbol{S}^{X,Y,Z}(\lambda xE'))(\lambda xE'')$, where

$$((\boldsymbol{S}^{X,Y,Z}(E'))(E''))(E''') = (E'(E'''))(E''(E'''))$$
$$\text{(hence } \boldsymbol{S}^{X,Y,Z} \text{ is } \lambda x\lambda y\lambda z((x(z))(y(z)))\text{).}$$

Given this, we can reduce any closed lambda term to a combination of simpler lambda terms; and we can use this to show, by induction, that any closed lambda term is equivalent to a term without a lambda. The same holds for the last possibility, when E is $\lambda xE'$.

In this way, we can show that we can get rid of (bound) variables not only in the language of first-order predicate logic but, more generally, in the language of lambda-categorial grammar, which, it would seem, is the most general type of language involving the mechanism of variable binding.

14.5 The Algebraic Structure of a Semantic System*

From a formal viewpoint, a semantic interpretation is just a homomorphism of the algebra of language in a different algebra such that its sort corresponding to that of sentences has a distinguished subset. (In the simplest case the set is {0,1} and the distinguished subset is {1}.) Then we can say that the interpretation satisfies (or is a model of) those sentences which it maps on a distinguished element. More generally, if we have a set of interpretations, we can say that it satisfies (or is a model of) those sentences which all its elements map on the distinguished elements.

More precisely, let us recollect that what we called the bare language was a finitely generated many-sorted algebra with a distinguished sort (of sentences). Thus, a bare language is a pair $L = \langle\langle\langle C_i\rangle_{i\in I}, \langle R_j\rangle_{j\in J}\rangle, i_s\rangle$ $(= \langle\langle[\langle W_i\rangle_{i\in I}], \langle R_j\rangle_{j\in J}\rangle, i_s\rangle)$, where $\langle\langle C_i\rangle_{i\in I}, \langle R_j\rangle_{j\in J}\rangle$ is a many-sorted algebra and $i_s\in I$. A formal semantics for L is an ordered quadruple $S = \langle\langle\langle D_i\rangle_{i\in I}, \langle R^*_j\rangle_{j\in J}\rangle, T, H\rangle$, where $\langle\langle D_i\rangle_{i\in I}, \langle R^*_j\rangle_{j\in J}\rangle$ is a many-sorted algebra, R^*_j is of the same arity as R_j for every $j \in J$, T is a subset of D_{i_s}, and H is a nonempty set of homomorphisms from $<<C_i>_{i \in I}, <R_j>_{j \in J}>$ to $<<D_i>_{i \in I}, <R_j^*>_{j \in J}>$. If $h \in H$, then a sentence s of L, that is, an element of C_{i_s}, is called *satisfied by h* iff $h(s) \in T$. s is called valid (or a tautology) in S iff it is satisfied by all elements of H. A sentence s is called a consequence of a sequence Γ of sentences of L iff s is satisfied by all those elements of H which satisfy all elements of Γ.

A formal semantics is usually based on some specific, more easily manageable kind of algebra of denotations. In particular, the rules of the algebra tend to be of transparent kinds. A rule R^*_j mapping $D_1, \ldots, D_n$ on D is called *applicative* iff there is a $k \in \{1, \ldots, n\}$ such that $D_k = D^{D_1 \times \cdots \times D_{k-1} \times D_{k+1} \times \cdots \times D_n}$ and $R^*_j(a_1, \ldots, a_n) = a_k(a_1, \ldots, a_{k-1}, a_{k+1}, \ldots, a_n)$ for every $a_1, \ldots, a_n$ from its domain; and an algebra is called *applicative* if it has only applicative rules. A domain of an applicative algebra is called *basic* if it is not of the shape $D^{D_1 \times \cdots \times D_n}$ for any other domains $D_1, \ldots, D_n$ and D.

We have seen that the semantic algebra for a categorial language may be applicative, indeed that an applicative algebra appears to form a natural semantics for such a language. The point is that to a category $C/C_1, \ldots, C_n$ there naturally corresponds the domain $D^{D_1 \times \cdots \times D_n}$, where $D, D_1, \ldots, D_n$ are the domains corresponding to the categories $C, C_1, \ldots, C_n$, respectively; and if $E, E_1, \ldots, E_n$ are expressions of the respective categories $C/C_1, \ldots, C_n, C_1, \ldots, C_n$, then $\|R_j(E,E_1, \ldots, E_n)\| = R^*_j(\|E\|,\|E_1\|, \ldots, \|E_n\|) = \|E\|(\|E_1\|, \ldots, \|E_n\|)$.

But we can consider other kinds of more or less transparent rules. A rule R^*_j mapping $D_1, \ldots, D_n$ on D is called *Y-quasiapplicative* iff there are sets $X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_n, X$ such that $D_1 = X_1^Y, \ldots, D_{k-1} = X_{k-1}^Y, D_k = (X^{X_1 \times \cdots \times X_{k-1} \times X_{k+1} \times \cdots \times X_n})^Y$, $D_{k+1} = X_{k+1}^Y, \ldots, D_n = X_n^Y$, $D = X^Y$, and $R^*_j(a_1, \ldots, a_n)$ is such a function f that $f(y) = (a_k(y))(a_1(y), \ldots, a_{k-1}(y), a_{k+1}(y), \ldots, a_n(y))$ for every $a_1, \ldots, a_n$ from the domain of R^*_j and for every $y \in Y$; and an algebra is called *Y-quasiapplicative* if it has only Y-quasiapplicative rules.

We call an applicative semantic algebra *extensional* if it has only two basic domains and if the domain corresponding to the category S has only two elements ($D_{i_S} = \mathrm{B} = \{0,1\}$), one of them being distinguished ($\mathrm{T} = \{1\}$).

We have seen that if we have an applicative semantic algebra, take a set W and make the algebra W-quasiapplicative, we get a trivially intensionalized version of the semantic algebra. We replace every domain D by the domain D^{W} and we make the rule R^*_j of the applicative semantic algebra corresponding to the rule R_j of the algebra of the bare language into a quasiapplication: if $E, E_1, \ldots, E_n$ are expressions of the respective categories $C/C_1, \ldots, C_n, C_1, \ldots, C_n$, then $\|R_j(E,E_1, \ldots, E_n)\| = R^*_j(\|E\|,\|E_1\|, \ldots, \|E_n\|)$ will be the function f from D^{W} such that $f(w) = (\|E\|(w))(\|E_1\|(w), \ldots, \|E_n\|(w))$ for every $w \in \mathrm{W}$.

A nontrivial version of intensional categorial grammar can be reached by explicitly enriching the bare language by a new category, I, corresponding to the domain W. Now the domain D^{W} will correspond to the category C/I, where C is the category corresponding to the domain D. The quasiapplicative rules that straightforwardly correspond to the applicative ones of the original categorial grammar will now combine expressions of the categories $(C/C_1, \ldots, C_n)$/I, C_1/I, . . . , C_n/I into an expression of the category C/I. But aside from those we will have the applicative rules of the new algebra, some of which will not have counterparts in the old one.

Recall that the usual languages of propositional logic are based on a single basic category, the category S of sentences. Aside from it, their expression inhabits only two functional domains: S/S and S/S,S (or (S/S)/S). And the semantics of this language can not only be applicative but also extensional. (Producing an intensional version of this language in the way sketched previously will equip us, for example, with the category (S/I)/(S/I), which would be suitable for the operators of necessity and possibility.)

We saw that it is also possible to see a propositional language as the one-sorted algebra $<S,\{R_\wedge,R_\vee,R_\neg,R_\rightarrow\}>$. Correspondingly, we can represent the semantic algebra as one-sorted, the carrier of which is the set of all propositions (denotations of sentences). The rules of inference of classical logic make this algebra into what is known as a *Boolean* algebra, which, as the theorem of Stone (1936) tells us, is always isomorphic to a set of subsets of an underlying set. And in the case of classical logic, we know that the underlying set can be a singleton, making B into a two-element set (a singleton has precisely two subsets, the empty set and the singleton itself), that is, {0,1}.

If the logic is different, what we reach in this way may be a different kind of algebra, in the case of intuitionistic logic it is what has come to be called Heyting algebra; three-valued logic require the set B consisting of three elements, etc.

The languages of predicate logic are based on two basic categories: aside from S there is also the category T of subjects. Their expressions inhabit, aside from the categories S/S and S/S,S, also the categories S/T, . . . ,T (predicates) and S/(S/T) (quantifiers) (provided that we disregard, as we do, functors – which would be in categories T/T, . . . ,T). Again, the semantics is extensional if there are only two potential denotations for sentences and one of them is distinguished. However, in their case we cannot make do with an applicative algebra of denotations. The most straightforward way to make their algebra of denotations at least close to an applicative algebra is to modify the algebra of language in that we enhance it with variables.

An *enhanced bare language* is a bare language such that its words (elements of C_i's) are divided into *constants* and *variables*. Thus, we can see an enhanced bare language as a pair whose first component is a triple $L = <<[<W_i \cup V_i>_{i \in I}],<R_j>_{j \in J}>,i_s>$, such that $<[<W_i \cup V_i>_{i \in I}], <R_j>_{j \in J}>$ is a many-sorted algebra and $i_s \in I$. (Elements of W_i are called *constants* of the category i; those of V_i are called *variables* of the category i.) A formal semantics for L is an ordered quadruple $S = \langle\langle\langle D_i^{\mathrm{V}}\rangle_{i\in I}, \langle R^*_j\rangle_{j\in J}\rangle, T, H\rangle$, where $\langle\langle D_i^{\mathrm{V}}\rangle_{i\in I}, \langle R^*_j\rangle_{j\in J}\rangle$ is a many-sorted algebra, V is the set of all sort-preserving mappings of $\langle V_i\rangle_{i\in I}$ into $\langle D_i\rangle_{i\in I}$, T is a subset of D_{i_s}, and H is a nonempty set of homomorphisms from $\langle[\langle W_i \cup V_i\rangle_{i\in I}], \langle R_j\rangle_{j\in J}\rangle$ to $\langle\langle D_i^{\mathrm{V}}\rangle_{i\in I}, \langle R_j^*\rangle_{j\in J}\rangle$.

If $h\in H$, then a sentence s of L, that is, an element of D_{i_s}, is called *satisfied by h with v* iff $h(s)(v)\in T$; it is called *satisfied by h* iff it is satisfied by h with every valuation v, s is called valid (or a tautology) in S iff it is satisfied by all elements of H. A sentence s is called a consequence of a sequence Γ of sentences of L iff s is satisfied by all those elements of H which satisfy all elements of Γ.

The core rules of the algebra of denotation for an enhanced bare language are quasiapplicative. In this case, however, we need, aside from quasiapplicative rules, also one more kind of rule, namely that corresponding to lambda-abstraction. A rule R^*_j mapping D^{V} on $(D^{D_1\times\ldots\times D_n})^{\mathrm{V}}$ is called *abstractive* iff $R^*_j(a)$ is such a function f that $f(v)(o_1, \ldots, o_n) = a(v[x_1\to o_1, \ldots, x_n\to o_n])$ (where $v[x_1\to o_1, \ldots, x_n\to o_n]$ is the valuation which is identical with v save for the possible difference that it maps x_1 on $o_1, \ldots, x_n$ on o_n) for all objects $o_1, \ldots, o_n$ from the domains $D_1, \ldots, D_n$.

Now we can consider the language of the classical first-order predicate logic as a simple kind of lambda-categorial language. It contains the expressions of the categories S, T, S/S, S/S,S, S/T, . . . ,T, and S/(S/T); it contains the quasiapplicative rules for all these categories; and it also contains an abstractive rule corresponding to $\lambda x_1 \ldots x_n$E for all variables $x_1, \ldots, x_n$ of the category T and E of the category S.

Consider, for example, the expression $\exists x\phi[x]$. Within the framework of the lambda-categorial language, it will have to be considered as a shortcut $\exists(\lambda x\phi[x])$. So consider its denotation w.r.t. a given valuation v, $\|\exists(\lambda x\phi[x])\|(v)$. As the rule for combining the quantifier $\exists$ (of the category S/(S/T)) with the unary predicate $\lambda x\phi[x]$ (of the category S/T) is quasiapplicative, $\|\exists(\lambda x\phi[x])\|(v) = \|\exists\|(v)(\|\lambda x\phi[x]\|(v))$. The rule which produces $\lambda x\phi[x]$, on the other hand, is abstractive, hence $\|\lambda x\phi[x]\|(v)(o) = \|\phi[x]\|(v[x\to o])$. As $\|\exists\|(v)$ is the set of all nonempty sets, this means $\|\exists(\lambda x\phi[x])\|(v)$ is 1 iff $\|\phi[x]\|(v[x\to o])=1$ for at least one o, which is as it should be.

In the case of the classical first-order predicate calculus, it is again also possible to consider the denotational algebra as a one-sorted one, in that we consider $\exists x$ as a new kind of unary sentential operator (for every variable x). This leads to an intricate extension of the Boolean algebra of denotation for the classical propositional logic, which came to be known as a *cylindric algebra*.[10]

10. Henkin and Monk (1974).

15 Matching Calculi and Systems of Formal Semantics

15.1 Soundness$_F$ and Completeness$_F$

We have seen that artificial languages of logic were developed especially because logicians felt that they needed firmer bedrocks for logical calculi – that it is not possible to build exact calculi atop of inexact natural languages, that the languages must be first replaced by some exact proxies. The calculi instituted, over the artificial languages, exact relations of inference and sets of theorems (as the sentences inferable from the empty sequence of sentences). Later, we saw, Tarski launched the enterprise of formal semantics, which led to the result that some artificial languages were supplemented by semantical systems, which instituted, over them, relations of consequence and sets of tautologies (sentences that are consequences of the empty set).

If we have, for a language, both a calculus and a formal semantics, in the most straightforward situation the relation of inference instituted by the former coincides with the relation of consequence instituted by the latter. We call a relation of inference that coincides with the corresponding relation of consequence *sound* and *complete*. However, note that these are concepts different from those of soundness$_M$ and completeness$_M$ that we used to characterize the relationship between a calculus and natural language with its *following-from*; therefore, we will talk about *soundness*$_F$ and *completeness*$_F$ (where the index "F" stands for "formal"). In fact, we now have counterparts for the whole battery of concepts we introduced with the index "M": we have already mentioned interpretation$_F$ as an assignment of denotations to the expressions, and also we have validity$_F$ as satisfaction by every interpretation$_F$.

Let us refine the terminology slightly. Let L be a language, *C* a calculus over it, and *S* a formal semantics for it. *C* is said to be *strongly sound*$_F$ *w.r.t. S* iff for every Γ and ϕ,

if $\Gamma \vdash \phi$ then $\Gamma \models \phi$

(where $\vdash$ is the relation of inference induced by *C* and $\models$ is the relation of consequence induced by *S*).

It is said to be (*weakly*) *sound*$_F$ *w.r.t. S* iff for every ϕ,

$$\text{if} \vdash \phi \text{ then} \models \phi,$$

that is, iff every theorem of *C* is a tautology of *S*. *C* is said to be *strongly complete*$_F$ *w.r.t. S* iff for every Γ and ϕ,

$$\text{if } \Gamma \models \phi \text{ then } \Gamma \vdash \phi.$$

It is said to be (*weakly*) *complete*$_F$ *w.r.t. S* iff for every ϕ,

$$\text{if} \models \phi \text{ then} \vdash \phi,$$

that is, iff every tautology of *S* is a theorem of *C*.

It is clear that a calculus that is strongly sound is weakly sound and a calculus that is strongly complete is weakly complete. However, in certain cases weak soundness may entail a strong one (and thus coincide with it). We have already seen that when $\rightarrow$ is governed by the metarule (Ded+*), it is the case that

$$\phi_1, \ldots, \phi_n \vdash \phi \text{ iff } \vdash (\phi_1 \rightarrow (\ldots (\phi_n \rightarrow \phi) \ldots)).$$

Consider the semantic version of (Ded+*),

(SDed+*) $\Gamma, \phi \models \psi \,//\, \Gamma \models \phi \rightarrow \psi$.

In this case,

$$\phi_1, \ldots, \phi_n \models \phi \text{ iff} \models (\phi_1 \rightarrow (\ldots (\phi_n \rightarrow \phi) \ldots)).$$

Theorem. If (Ded+*) holds for *C* and (SDed+*) holds for *S*, then *C* is weakly sound w.r.t. *S* only if (and hence if and only if) it is strongly sound w.r.t. it.

Proof: Assume that $\Gamma \vdash \phi$ and that *C* is weakly sound. There must be the case that $\Delta \vdash \phi$ for some finite Δ such that $\Delta^* \subseteq \Gamma^*$ (where Γ^* is the set of all elements of which the sequence Γ consists). Assuming that $\Delta = <\phi_1, \ldots, \phi_n>$, $\Delta \vdash \phi$ iff $\vdash (\phi_1 \rightarrow (\ldots (\phi_n \rightarrow \phi) \ldots))$. As the calculus is weakly sound, this can only be if $\models (\phi_1 \rightarrow (\ldots (\phi_n \rightarrow \phi) \ldots))$, and hence that $\phi_1, \ldots, \phi_n \models \phi$ and hence only if $\Gamma \models \phi$.

Can we use a parallel consideration to show that weak completeness entails a strong one? Unfortunately not. The reason why we cannot reverse the previous proof is that we cannot generally say that if $\Gamma \models \phi$, then $\Delta \models \phi$ for some finite Δ such that $\Delta^* \subseteq \Gamma^*$. This would amount to the compactness of $\models$, which is only an *optional* property of consequence.

15.2 A Calculus Without a Semantics?

Can we have a calculus to which there is no matching semantics? It is clear, from the history of modern logic, that various searches for a suitable semantics for some calculi were quite tortuous. This is the case, for example, of the five calculi of modal logic proposed by C. I. Lewis in the thirties (Lewis & Langford, 1932); some of them (S4-S5) were equipped by a commonly accepted semantics by Kripke (1963a, 1963b), while others were even more troublesome (Kripke, 1965b, supplied a semantics for S2 and S3). A still more obvious case appears to be intuitionistic logic, which was the first serious rival to the classical one and which only obtained the first truly formal semantics from Kripke (1965a).

However, these calculi were, in the end, matched by semantic systems – could it be that for some calculus it is not possible to find any semantics at all? To answer the question, we would need to know, more exactly, what a (formal) semantics is.

It is, of course, the case that we know what a formal semantics is in the sense that we have a lot of commonly endorsed systems of formal semantics, such as the extensional semantics for classical logic and its higher-order expansions, the possible worlds semantics for modal and intensional logic, and various other cases of more complex semantic systems. However, what, quite generally, counts as a semantic system?

We saw that a semantic system is based on a set of interpretations (interpretations$_F$, that is, but in this chapter we still talk only about them, so we keep leaving out the subscript). Each of them maps all expressions of the language in question on some entities. (In some cases, like the traditional expositions of interpretations of classical logic, only some expressions are interpreted, others being treated as "syncategorematic"; but we saw that usually such interpretations can be turned into an assignment of entities to all expressions.) Moreover, we have considered interpretations as homomorphisms of the algebra of the language into an algebra of denotations – which translates into a less technical idiom so that an interpretation is a mapping that is compositional. Could we then say that an interpretation is any such homomorphic or compositional mapping of the language on anything?

Hardly. The reason is that an isomorphism is also a homomorphism and so the identity mapping of the language on itself would count as semantic interpretation, which is certainly quite counterintuitive. Moreover, this construal of the concept of semantic interpretation would render the question about an existence of a semantics for a given calculus trivial – we could always define the semantics as consisting of a single interpretation, the identical one, with the set of theorems as the distinguished subset of the set of denotations of sentences. Now, it seems, we need to have to spell out a criterion according to which this is not a semantic interpretation.

What may come to mind is that the mapping of language on itself does not count as interpretations, for denotations must be *extralinguistic* objects. But this, of course, does not help, for it is easy to replace expressions, for the purpose of playing the roles of denotations, by some proxies. (We could, for example, use a Gödelian mapping of expressions on natural numbers.)[1] Hence the criterion excluding the identity mapping needs to cut deeper.

Intuitively, it might have to do with simplicity – the algebra of denotations appears to have to be in some sense simpler than that of expressions. In one sense, the simplicity concerns the number of denotations: it would seem that there should be fewer denotations than expressions, if for no other reason than because there are usually synonymous expressions, that is, different expressions with the same denotation. In another sense, simplicity may concern the operations of the semantic algebra; namely, the ways in which denotations of complex expressions are composed of those of their parts.

As for the first sense, what we may require is that sentences that are equivalent from the viewpoint of a calculus (in the sense of being interinferable) receive the same denotation (and, moreover, that so also do all expressions that are interchangeable w.r.t. inference). This, however, obviously does not concern the definition of interpretation as such, only a definition of interpretation "adequate" to a given calculus. As for the second sense, we may restrict the operations of the semantic algebra to some transparent kind (in the sense in which we did put a premium on applicativeness); however, there does not seem to be a consensus on any such restriction being incorporated within the very definition of semantic interpretation.

All in all, as it does not seem to be quite clear what the boundaries of the concepts of semantics and of interpretation, and consequently of the concept of a system of formal semantics, are, it is not clear how the claim that there is or can be a calculus that cannot have a semantics could be substantiated. (Of course, if we make the concept of semantics more definite, we may be able to show that either every calculus does have a semantics or that there are calculi that do not have any. Thus, for example, Font et al., 2003, show that every calculus has what they call *matrix semantics*.)

1. Moreover, model theoreticians do not shy away from engaging denotations built directly from the language that is to be interpreted (*viz.*, e.g., the proof of completeness of type theory due to Henkin, 1950). (Conversely, there are cases when they build language out of the items of a given model – see, e.g., Goldblatt, 1984, p. 487.)

15.3 A Semantics Without a Calculus?

In contrast to the blurry delimitation of the concept of formal semantics, the delimitation of the concept of calculus is quite exact. So it might then be possible to find something that is generally accepted as a formal semantics and that delimits a set of tautologies that cannot be delimited by a calculus. It is also well-known that classical propositional logic is strongly sound$_F$ and complete$_F$ with respect to truth-functional semantics (Kalmár, 1935); that is, that the truth-functional semantics is adequate to the calculus in the sense that inference instituted by the calculus coincides with the consequence as instituted by the semantics.

It is no less well-known that the calculus of classical first-order predicate logic is at least weakly sound$_F$ and complete$_F$ with respect to its standard semantics (Gödel, 1930): that is, that the semantics is adequate to the calculus in the sense that the set of theorems of the calculus coincides with that of the tautologies of the semantics. (Strong soundness$_F$ and completeness$_F$ depends on how exactly we construe inference and consequence as applied to open formulas: it is possible that we consider $\forall xP(x)$ as inferable from $P(x)$, without being its consequence, in which case we do not have strong soundness$_F$.)[2]

However, the situation is usually considered to be different in the case of second-order predicate calculus. We have already seen that from the viewpoint of semantics, there is a significant dividing line between first- and second-order logic. Now it turns out that there is also a significant distinction between the two logical systems, namely that there are second-order tautologies that are not theorems of any calculus of second-order logic; hence, that the semantics for second-order (and, of course any higher order) logic is "non-calculizable." Why is this the case?

Return to Gödel's proof of incompleteness: it proved the deductive incompleteness of PA and of any axiomatization of arithmetic. As a by-product of the other Gödel's proof, the proof of completeness$_F$ of first-order logic, we get the proof that any consistent theory in first-order logic has a model. Hence, if we take the theory PA, we have a formula $\boldsymbol{G}$ such that neither $\boldsymbol{G}$ nor $\neg\boldsymbol{G}$ is provable in PA, thus both the theory that results from PA by adding $\boldsymbol{G}$ and that which result from PA by adding $\neg\boldsymbol{G}$ are consistent and hence have a model. Both the models are models of PA

2. As regards inference, we saw that we can take generalization as a rule ($P(x) \vdash \forall xP(x)$), or else merely metarule ($\vdash P(x)$ // $\vdash \forall xP(x)$). Similarly, we can distinguish between two varieties of consequence: the so-called *local* consequence, according to which B is a consequence of A if B is satisfied by every interpretation and valuation that satisfy A and *global* consequence, according to which B is a consequence of A if every interpretation that satisfies A together with every valuation also satisfies B (together with every valuation). The first construal of inference then goes with the first construal of consequence, while the second one goes with the second.

and they are not isomorphic. It turns out that while both models contain the infinite sequence of natural numbers, ascending from zero to infinity, one of them has some more "numbers" that are bigger than all the standard natural numbers – that is, that come "after infinity."[3]

Now the point is that in axiomatizing PA in second-order logic we can articulate the axiom of induction so that it excludes all the non-standard models. The axiom says that every set that contains zero and is closed to the successor operation contains all the elements of the universe; hence, that in effect nothing that may come "after them" can be a natural number. Thus, if we have the conjunction of all second-order axioms of PA, call the formula ***PA2***, we can form the formula $\boldsymbol{PA2 \rightarrow G}$, which is a tautology, for ***G*** holds in every model of ***PA2*** (for there is only the standard model, the non-standard ones were excluded by the second-order induction).[4] On the other hand, Gödel's proof of incompleteness keeps holding for second-order PA, and hence ***G*** is not inferable from ***PA2***. Thus, $\boldsymbol{PA2 \rightarrow G}$ cannot be a theorem.

In this way we see the price that we must pay for the higher "expressive power" of second-order logic: we cannot have a calculus!

15.4 Henkinian Semantics for Higher-Order Logic

We have seen that probably for every conceivable calculus there is a semantics, or at least something that could be called a semantics. Now that we have seen that the semantics which comes naturally for second-order logic (the semantics is usually called "standard") does not fit with the calculus in the sense that the calculus is not complete$_F$ with respect to it (and no alternative calculus can do better), what might come to mind is that there should be a *fitting* semantics for the calculus. Is this the case?

Indeed it is. Return to the point where we extended the semantics for first-order logic to that for the second-order one. What was clear was that the domain of denotation for predicates should consist of elements of $B^{U\times\cdots\times U}$ – after all, this already holds for first-order logic. However, what we must now specify (and what need not have been specified for first-order logic) is exactly which functions from $B^{U\times\cdots\times U}$ form this domain – for this is relevant for the truth values of the sentences quantifying over predicates.

Of course, the answer that sounds most natural is: all of them. According to this answer, the domain of the denotation of *n*-ary predicates is the whole set $B^{U\times\cdots\times U}$. This is what leads to the standard semantics, w.r.t. which no calculus is sound$_F$ and complete$_F$.

3. See Gaifman (2004).
4. It follows that there are consistent second-order theories that have no model.

However, there is also an answer that is perhaps only slightly less natural. Recall that the model theory for first-order logic converged to treating the universe of discourse as *arbitrary* – the only truths that interest logic are those that are true independently of what the universe looks like. Now the universe of individuals also partly determines the denotations of predicates: the denotations must be the functions from the universe to truth values. However, it does not seem to determine the whole domain of the denotations uniquely: The domain must contain those functions that are used for the interpretation of the actual predicates (and pseudopredicates) of the language,[5] but which others should it contain? Should this not again be left arbitrary? Should we not, instead of saying that the domains corresponding to the categories of predicates contain *all* functions of the respective types, say that each of the domains contains an *arbitrary* assortment of such functions (over and above those that are needed for the interpretation of actual predicates)?

Clearly we can do this; and this was also how Henkin (1950) constructed the semantics for higher-order logic. This let him prove the completeness$_F$ of the calculus w.r.t. this semantics, for this is the semantics which fits with the calculus in such a way that the calculus is sound$_F$ and complete$_F$ with respect to it.

Now we can return to the problem we tabled when we investigated the expressive power of the predicate calculus (§13.7) – namely, that we can translate the language of second-order logic into a first-order one, but that we lose the specific kind of expressive power that, for instance, allows us to get hold of the concept of infinity. The language of second-order logic with a Henkinean semantics turns out to be equivalent to the first-order *simulacrum*.

Recall that the problem with emulating a second-order logic within a first-order one was with the stipulation that we have the counterparts of *all* relations over U. There was no way to force this by the means of a calculus of first-order logic; the only way to do this was to use a second-order quantifier of which we simply stipulate that it quantifies over all of them. Thus, in effect, the notion of *all functions* was taken as a primitive one, stipulatively connected with the corresponding quantifier.

15.5 Categoricity

A *formal* language, as we defined it, will have a lot of models, as it is not really a language but rather a language *form*. The situation is different in the case of a *formalized* language, such as the language of PA. Typically,

5. This can be secured by the a comprehension axiom scheme stating that $\exists p \forall x_1 \ldots x_n$ $(p(x_1, \ldots, x_n) \leftrightarrow \phi[x_1, \ldots, x_n])$ for every formula $\phi[x_1, \ldots, x_n]$ with the free variables $x_1, \ldots, x_n$.

it aims at having only one model, and indeed the ambition of PA is to pick up one specific model – the sequence of natural numbers. A formalized language (theory) that manages to specify a unique model is called *categorical*.

Of course when we say "unique" here, what we mean is "unique up to isomorphism," for we saw that if there is a model for a language of logic, then anything that is isomorphic to it is obviously also a model. Thus, when we say that we want PA to pick up a single model, what we literally mean is one *structure* shared by any number of concrete isomorphic models. But anyway, we have seen that PA within the framework of first-order logic did not manage to do this. As Gödel revealed, this language has more than one model: besides the standard one it has also nonstandard models.

Aside from this, the Löwenheim-Skolem theorem tells us that if a first-order theory has an infinite model, it also has models of all the possible infinite cardinalities (see, for example, Mendelson, 1964, §2.13). Thus, as the model of PA cannot have a finite universe, there must be an infinity of non-isomorphic models of all possible infinite cardinalities. (This, of course, is the case if we accept the – currently mainstream – mathematical assumption that there are different infinite cardinalities.)

Thus, we cannot uniquely characterize natural numbers within first-order logic. We can do it, we saw, within second-order logic, but the price we have to pay for this is that we cannot have any calculus for it.

Categoricity is especially important from the viewpoint of model theory (see §13.5). It is in its context that we often face the task of finding a theory which manages to single out a pre-given model structure or a set of model structures.

15.6 Did Semantics Help?

The reason why many logicians embraced the Tarskian project of formal semantics was, we saw, the result of the frustration of the Hilbertian project of reconstructing following-from in terms of inferability within a calculus. Formal semantics offered an alternative reconstruction, in terms of truth – namely, truth-preservation or satisfaction-preservation, which promised to do better. Hence, the question is whether this move paid for itself, whether it allowed us to get a better grip on *following-from*, and if so, if the complication of introducing the semantic apparatus was not too high a price for this.

Consider the special case pointed out by Tarski. Does semantics make the sentence $\forall x P(x)$ of PA into a consequence of P(n) for every natural number n? Well, it does, insofar as the universe of the language of PA contains all and only natural numbers. But this looks somewhat vacuous. If we set up the universe, stipulatively, so that it contains all natural numbers, then that something holds for every member of the universe trivially entails that it holds for all the natural numbers. It is true that the apparatus of model theory makes straightforward room for the possibility of non-compact relations of consequence; however, by itself it does not allow us to handle them in a very nontrivial way.

We saw that the most basic problem for the proof-theoretical approach was that following-from does not seem to be generally compact, while the rules of derivation were bound to have a finite number of premises. The semantic approach did away with this limitation; there is no reason why there cannot be a formula, say $\forall xPx$, which is true only in case an infinite number of cases obtains (for example if $\|P\|$ yields 1 when applied to every of the infinite number of elements of the universe). However, there is a price to be paid for this: switching from inference (defined in terms of a calculus) to consequence (defined in terms of a semantic system), we lose the "determinatedness" of derivations, in the sense that unlike inference, consequence need not be determinate even in the sense in which inference is, all its instances being generatable from a finite basis. Hence, from the point of capturing *following-from* embracing semantics, this is a trade-off: we gain something for losing something.[6]

6. It is worth noticing that a similar kind of effect could be achieved without going (explicitly) semantic, *via* releasing the restriction that the rules of inference can only have a finite number of premises. And obviously this would again lead to the same loss of "determinatedness." See Peregrin (2014, Chapter 7).

16 Through the Looking Glass . . .

16.1 The "Natural" and the "Artificial"

In previous chapters, we have distinguished between natural language, equipped with the relation of following-from, which is the natural vehicle of our reasoning, and various artificial languages, calculi, and systems which we build to account for it (and perhaps also to enrich it or to replace it in certain contexts). The artificial tools are ubiquitous in modern logic, and sometimes they play roles that it is difficult to duly reflect. Therefore, it is essential to be clear about the possibilities and limitations of their employment.

It is often assumed, generally without reflection, that there is no substantial difference between what we called the *material* interpretation of a logical language (interpretation$_M$) and its interpretation within a *formal semantics* (interpretation$_F$). More generally, the distinction is between the concepts that are internal to the machinery of logic (which it has introduced to account for following-from) and those that are external to it, for they interlink the machinery to the following-from, which is the *raison d'être* of logic. Interpreting$_M$ a formal or a formalized language, we map its expressions on those of a natural language, so the story goes, and thereby on that which is denoted by the natural language expressions. The interpretation$_F$, on this picture, only avoids the detour via natural language and maps them directly on the denotations, as in Figure 16.1.

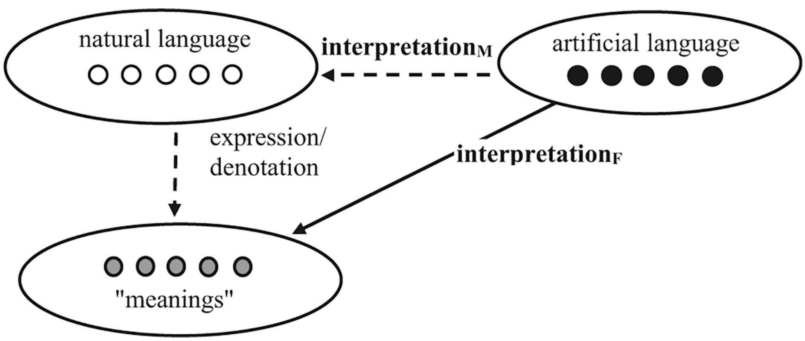

Figure 16.1

Therefore, there is no need to make any categorical distinction between the two concepts.

I am convinced that this line of thought ignores a crucial boundary that is being crossed when we move from interpretation$_M$ to interpretation$_F$, and this is the boundary between the denizens of the "natural" world and that of the world of "ideal" entities we build stipulatively. Note that the languages of logic are completely created by our definitions, hence they are, as I call them, artificial.[1] Studying properties of such languages amounts to studying the consequences of our definitions. But reasoning is something we do, in the real world, and we can do it better or worse; and logic is to help us do it better. How can any investigation of the consequences of our definitions offer us any help in this?

I have suggested that the artificial languages of logic should be seen as mathematical models of (parts of) natural language, and especially those of its rules that are constitutive of one of the crucial "games" we play with it – the "game of giving and asking for reasons," also known as argumentation. (We must keep in mind that "natural" in "natural language" does not mean anything like "everyday," but rather "not created by stipulation"; thus, even mathematics is carried out almost exclusively in a natural language, though often in one that is in some respects enhanced.) Thus, the artificial languages do a similar job as mathematical models of physical phenomena, which capture other aspects of the real world (and so help us deal with it). Thus, it is often crucial to look into how far the model is adequate to what it models.

We must sharply distinguish between the questions that concern the model as such and those that concern the way it fits with what it is meant to model. The former questions allow for a straightforwardly mathematical treatment: the way to find out that the model has some properties is to prove that this follows from its definition. The latter questions, on the other hand, cannot be subjected to a mathematical treatment.

Consider, for the sake of illustration, the concept of computability. This is a natural concept in the sense that it is a fact of the matter that something can be computed whereas other things cannot. Various mathematical models of computability were put forward: Turing machines, recursive functions, computability in lambda calculus, and so on (see, for example, Boolos et al., 2002). All these models were shown to be equivalent. However, there is not, and there cannot be, a proof that any of these models is equivalent to the pre-theoretical phenomenon, simply because the pre-theoretical phenomenon is not something that could figure in a proof. That the independently proposed explications come to one and the same mathematical concept is a strong indication that it

1. In my previous writings, I used the terms *natural* and *formal* (see especially Peregrin, 2000c). However, here I want to preserve the formal for artificial languages that are to capture forms (in contrast to formalized ones); hence, I must use a different term.

is adequate, and it is probably the only "proof" we need to take it for adequate; the Church's thesis stating this equivalence is very different from a thesis stating the equivalence of two mathematical – or artificial, in our terms – concepts.

The same thing holds about properties of the natural language and of its artificial regimentations and formalizations. We can prove that two properties inside one model, or in different models, are equivalent; we cannot prove that a property from a model is equivalent to a property that is from the world outside of the model. And while the concepts with the index F concern the artificial entities within the models, those with the index M concern the relationship of the artificial properties to the natural ones.

16.2 Model and Reality

There are entities that are "natural" in the sense that we *find* them in the nature which surrounds us (where the term "nature" should be understood in the broadest sense, as encompassing everything that we can encounter and explore); and there are things which are artificial in the sense that they are wholly created by us, especially that we produced them in terms of definitions and stipulations.

These considerations lead us to a complete redrawing of the picture of the relationship between a natural and an artificial language presented previously. A crucial boundary is the boundary of the logical system as an artificial entity, which we create in terms of our definitions. (This system may incorporate not only the bare language and possibly a calculus but also the formal semantics.)

This whole system acts as a model for a certain part of external reality, namely for a natural language with its pre-theoretical following-from

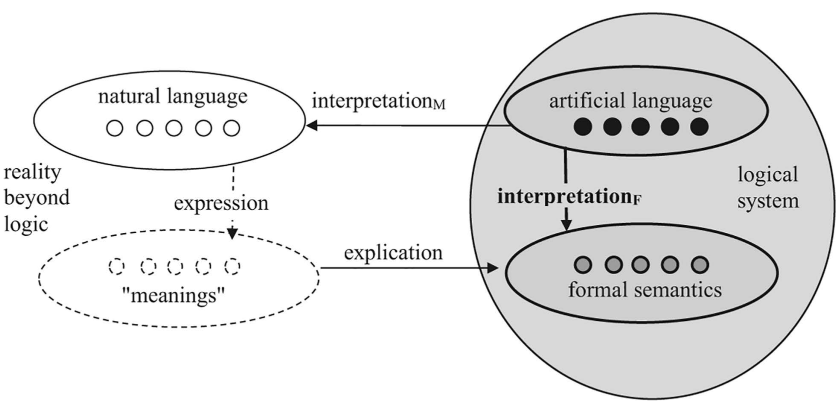

Figure 16.2

and its pre-theoretical semantics. However, it is not true that the bare artificial language with the calculus models the natural one with its pre-theoretical following-from while the formal semantics models the pre-theoretical meanings of expressions of natural language.

The meanings of expressions of natural language, by themselves, are blurry matters that could hardly be directly modeled – it is not so that the meanings would be some clear-cut objects which would only be labeled by expressions and which, in principle, could be accessed directly. It is hardly possible to point out the meaning of, say, the word *run* and then try to conjure its simplified simulacrum to be incorporated into the model. What we instead do is inspect the behavior of the corresponding expression, its role in language, and then use this as a basis for conjuring an artificial object embodying its meaning within the model.[2] Thus, the basic point of contact between the logical model and the reality it models is the natural language and its rules. True, the artificial meanings or denotations which become explicit as a part of the logical model can then be seen as explications of the meanings of the expressions of the natural language; the explication, however, should not be seen as leading from a fuzzy or indeterminate object to a crisp and determinate one but rather as leading to a determinate object (constructed by us) from something that is not yet an object at all.

16.3 Logic as a Natural Science?

The view that our logical theories are akin to models used by natural scientists was spelled out by a few authors, perhaps most clearly by Shapiro (1998, p. 137):

> A formal language is a *mathematical model* of a natural language (of mathematics), in roughly the same sense as, say, a Turing machine is a model of calculation, a collection of point masses is a model of a system of physical objects, and the Bohr construction is a model of an atom. In other words, a formal language displays certain features of natural languages, or idealizations thereof, while ignoring or simplifying other features.

Similarly Cook (2010, p. 500):

> [A] logic is not meant to be a perfect representation of the linguistic phenomenon being studied. Instead, a logic is a model of that linguistic phenomenon, and as a result, all of the advantages and limitations

2. In the sense of Lewis (1972, p. 173): "In order to say what a meaning *is*, we may first ask what a meaning *does* and then find something which does that."

> that are present in modeling elsewhere (such as in the empirical sciences) should be expected to reappear in the study of logic.

If you look at logic as a theory of human reasoning, then this view comes naturally. There are human practices of (overt) reasoning and argumentation, which, I would hasten to add, incorporate certain rules (in the sense that we not only draw inferences or give reasons but also readily assess each other's arguments and reasons as correct or incorrect[3]). Then we can see logical theories as capturing regularities of these practices (and of the assessments, that is their rules) in a sense not very different from that in which physical theories capture the swarming of things in space-time. (There are, to be sure, the dissimilarities concerning normativity spelled out in Chapter 1.)[4]

Things are, of course, quite different if you view logic as a discipline the task of which is to capture something covert, beyond language or empirically investigable thought. Dutilh Novaes (2012), for instance, protests against the understanding of the artificial languages of logic as models. Her reasons are twofold. First, she has a positive story about what we use artificial languages for and claims that this is not the usual role of models. Her main hypothesis is that "formal languages are best viewed as a cognitive technology, developed in order to facilitate a range of cognitive processes" (p. 66); that artificial languages are not means of *representing* natural language but rather a tool to think and calculate with, to calculate things that need to be calculated during reasoning and that are hard to calculate in terms of natural language signs encumbered with their often convoluted meanings, while they can be much more easily calculated with signs unburdened with meanings.

I think that this indeed is true (and I think that Dutilh Novaes does an excellent job explaining the role of formal languages as tools of calculation). Only I also think that she exaggerates the difference between the means of representing and the tools of calculating. Models in natural science, to which we liken the artificial languages of logic, are often both. True, these roles may sometimes pull in opposite directions (as Dutilh Novaes, p. 95, puts it, "there is a sense in which expressive and calculative motivations are in tension with each other"), but so it is with models in natural sciences. (A model like Bohr's, of course, puts a premium on perspicuity; while, say, a system of differential equations as a model of a flowing fluid puts it on computational tractability.)

However, Dutilh Novaes also has another, negative, motivation for rejecting the notion that the artificial languages be seen as models of

3. The way in which such "normative attitudes" constitute implicit rules was anatomized by Peregrin (2014, Chapter 4).
4. See Peregrin (2019) for a more thorough discussion.

natural language. The problem, she thinks, is that "to describe a given target phenomenon, a formal language or formalism makes a detour via ordinary language discourse about the target phenomenon in question," while "a formal language will be much like any mathematical formalism, a 'language' in itself which can characterize directly the target phenomenon without the mediation of ordinary languages" (p. 99). Hence, the view is that the artificial languages are not models of natural ones, but rather deal with "target phenomena"; namely, the ones which are dealt with (not so well) by the natural languages. And this is precisely the view I reject in this book.

What are the "target phenomena" the languages of logic are to deal with? Presumably, in the case of formalized languages it can be anything with which the parts of natural languages formalized by them deal (like natural numbers in case of PA). However, what about *formal* languages, which are the most important items from the viewpoint of logic? What are the "target phenomena" they are to deal with?

Here an answer might be perhaps extracted from an example given by Dutilh Novaes (p. 100):

> For example, in my own work of formalizing medieval logical theories . . . the procedure was never to engage specifically with the textual formulation of these theories (in a 'translation enterprise', as it were), but rather to absorb for as much as possible the conceptual structure of these theories through the text, and then to formulate it in a given formalism – what I have described . . . as 'conceptual translation.'

Hence, it seems that in cases of logical languages, the "target phenomena" are some conceptual structures beyond languages (perhaps something like "logical forms"?), which are expressed, only imprecisely and tentatively, by natural languages. (A version of this construal may be that artificial languages capture the objective, ideal forms of the entities of the real world;[5] or, alternatively, that they are objectively present within the human mind, perhaps as some tokens of a language of thought.[6]) If this is so, should the artificial languages not take over and leave natural language behind?

Let us, for a moment, disregard the qualms regarding the evidence of the objective existence of entities such as conceptual structures or logical forms. Independently of this, it seems that the arguments the correctness of which we prove in the artificial languages are useful only insofar as

5. See, for example, Sher (2011).
6. See Fodor (2008).

they help us decide which of the arguments that can be articulated in natural language are correct. Natural language (in the very broad sense of the word, in which it is contrasted with artificial ones) is the *lingua franca* of different minds; but it is not only that arguments that cannot be translated into it cannot be *shared* by individual minds; in fact, they do not count as real arguments. (Imagine someone were to say: 'I have an ingenious argument in my mind; however, I am not able to articulate either it or its conclusion.')

The key point, I think, is very well expressed by Mercier and Sperber (2017, p. 172):

> Unlike verbal arithmetic, which uses words to pursue its own business according to its own rules, argumentation is not logical business borrowing verbal tools; it fits seamlessly in the fabric of ordinary verbal exchanges. In no way does it depart from usual expressive and interpretive linguistic practices.

And similarly Shapiro (1998, p. 137):

> As such, reasoning involves the use or apprehension of linguistic items by the reasoning subject. So we must consider the role of a *natural language*, like English, or perhaps a natural language augmented with the special terminology and symbols of a branch of mathematics.

Could the problem be that the natural languages we have are deficient and that much of what we are not able to articulate *now* will yield to articulation when we have a better one? This idea, propagated by the "ideal language" movement,[7] resonated throughout the first half of the twentieth century; some of its proponents were adamant that the regimented languages will soon take over – at least, perhaps, in the realm of science. But nothing like this has happened, indicating that natural languages, as they stand, have something to them.

Consider the English sentence *Birds fly*. Is it not hopelessly vague, and does it not become precise only when we get to its logical form and find out whether it says that *all* birds *always* fly, or *all* birds are *capable of flying*, or that *almost all* birds fly, etc.? Well, true, the sentence is indeed ambiguous. But so are almost all the sentences of a natural language: what we do when using a natural language is stick to the minimal level of precision and unambiguity that is needed in the current context, being prepared to make it more precise when it becomes necessary.[8]

7. Rorty (1967) explains this in detail.
8. This feature of natural language was famously scrutinized by Waismann (1968) under the heading of "open texture."

But, those who see logic as the pursuit of "true" logical forms may object that this is precisely the point: when we do science or engage in rational argumentation we need a language that pushes this to the limit and does away with *any* ambiguity, a language that captures the logical forms perfectly and is therefore not ambiguous. However, the problem is that we can hardly do away with every conceivable ambiguity. When I clarify that, by *Birds fly*, I mean *All birds always fly*, do I dispense with every conceivable ambiguity? Surely not. Do I mean that all *actual* birds fly, or that also all past (and perhaps future?) birds fly? And what exactly counts as flying?

True, talking about conceptual structures or logical forms may help us be, on the whole, less ambiguous with respect to logically correct arguments. But it is just the kind of disambiguation we perform when we make the meaning of *Birds fly* more precise by reformulating it as *All birds always fly*; it is not penetrating to a level of meaning where ambiguity is completely absent. There is no meaning free from ambiguities and other imperfections which affect ordinary expressions[9] – the aim of logic therefore cannot consist in penetrating to a level of completely disambiguated meanings or logical forms, for there is no such level.

16.4 Looking at a Model vs. Looking Through It

The upshot of the previous sections is that if we consider the role of a logical model within an investigation, we should always carefully distinguish whether it is an investigation *of* the model or rather an investigation of something else *in terms* of the model. That is to say: whether the investigation exclusively concerns the model or whether it concerns the phenomenon modeled and uses the model only as its proxy. If we do not distinguish between these two perspectives (akin to the distinction between looking *on* a microscope – or a window – and looking *through* it), grave conceptual confusions are likely to arise.

Most of the artificial languages and calculi were introduced as tools to help us analyze, regiment, and improve our reasoning, which we normally carry out in natural languages. We saw that logical constants were introduced as exactly delimited counterparts of certain constituents of natural language. Thus, material implication, as well as other, more sophisticated versions of implication, were introduced as counterparts of such connectives as *if-then*. Now the question is what can we achieve by introducing, besides our fuzzy natural language, another, exactly defined language, or a skeleton of such a language.

The answer to this question that I have put forth is that we can use an artificial language as a model of a natural one (especially in its capacity of a vehicle of reasoning), in a similar sense in which physicists use certain

9. See Peregrin (2010c) for more details.

models of natural phenomena. The model is simplified and idealized and therefore much more perspicuous than the original phenomenon; insofar as it is, despite this, adequate to the phenomenon in its relevant respects, we can replace the study of the latter by the much easier study of the former (and, needless to say, it may also help us deal with the phenomenon). When we do this, however, we must use the results gained by studying the model to deal with the original phenomenon.

This is not what usually happens in contemporary logic. Shapiro again (1998, p. 133):

> In a typical book or class on symbolic logic, one or two of the preformal characterizations of logical consequence are given at the outset, with or without a brief discussion. Very quickly, however, the reader or student finds himself proving theorems about the relation of satisfaction between certain strings on a finite alphabet and certain sets of n-tuples. The question before us here is the relationship between this model-theoretic notion of consequence and the ones given at the start of logic texts and courses. If we have truly succeeded, what is it that we have accomplished?

Logicians often stay within the formal languages and appear to be quite content with the results concerning these languages. This may be for several different reasons. One is that the artificial languages originally built via the regimentation of the natural ones sometimes represent relatively complex mathematical structures posing nontrivial purely mathematical problems. Hence, solving them is an achievement in its own right. Most of what goes on today under the heading of "mathematical logic" is of this kind. Mathematics is certainly a respectable science, so such results, if they are nontrivial, are certainly unobjectionable – only they no longer *directly* relate to the agenda of logic.

Often, however, those who also do not claim to do mathematics settle within the models and do not seem to be much interested in what is going on outside of them. What might be behind this is the tacit conviction, mentioned previously, that, as our natural language is a mere medium, reasoning does not happen directly in it but in something behind it and is therefore merely presented by it, and artificial languages are the more direct presentations of this something. Thus, once we have direct access to a phenomenon mediated by an artificial language, we can simply forget about the natural language.

A mundane objection to this is that though artificial languages have been in play for at least one and a half centuries, our reasoning has not vacated natural language and moved over to them to any significant extent. If reasoning is carried out in something that is only mediated by natural language, and if we have a language that is much better in this mediation, then the question is why this language did not quickly displace natural language, at least in the context of science.

A deeper objection concerns the very existence of the "something" that is only mediated by natural language and that can be accessed directly, bypassing the language. Unless we accept some naively oversimplified representationalist conception of language, we must realize that language is not only a set of labels stuck on entities of the extralinguistic world (which can be easily replaced by better labels), but that it takes part in the very structuring of the world in which we live (and in this sense in the constitution of the entities of which it consists). Thus though we may certainly imagine expressions as representing something "behind" them, this "something" is not so easily detachable from them to make the replacement of a natural language with a formal one into a method of logic.

True, artificial languages may have some properties that appear to make them more suitable vehicles of reasoning – such as a large reduction of ambiguity, fuzziness, and so on. But the fact that we have managed to develop a new tool that may, in some respects, be better than the original one certainly does not mean that the original one was, from the very beginning, an imperfect representation of something that the new one represents perfectly.

16.5 The External and the Internal

It may be helpful to consider some examples of neglecting the distinction between "the natural" and "the artificial." Remember what is in question. Some concepts and some questions merely concern the apparatus logic developed to do its job; they are *internal* to it. Other concepts, on the other hand, concern the (*external*) relationship between the apparatus and reality. And if the apparatus incorporates a formal semantics, then there is a certain duplication: the same term is sometimes used to refer to an internal concept concerning formal semantics and to an external one concerning reality.

Take a sentence of a formal language containing parameters such as $\phi\vee\neg\phi$. It may be called *valid* iff all its instances are true. But what are the instances? One possibility is that they are English sentences of the form *ϕ or not-ϕ*, such as *It rains or it does not rain*. (These are our interpretations$_{\mathrm{M}}$ and hence the resulting concept of validity is our validity$_{\mathrm{M}}$.) Another possibility is that they are instances of an artificial language, such as those of the language of propositional logic we introduced previously (such as $A\vee\neg A$) or whatever they are mapped on by some interpretations$_{\mathrm{F}}$ (in which case we have validity$_{\mathrm{F}}$). In the first case, the concept of validity in question is an external one, interconnecting the apparatus of logic with what it is devised to account for; in the second case, it is internal to the apparatus.

However, logicians sometimes fluctuate between these senses with an "unbearable lightness." Consider a paper of Gillian Russell (2015). Its main topic is the validity of the basic logical laws (and their justification). Laws of the kind of $\phi\vee\neg\phi$, are discussed, but we never learn what kind of validity we consider. If it is the external concept of validity, then it would seem the question turns out to be an empirical one – English is an empirical entity (if an entity at all), and we would have to inspect it to find out which of its sentences are true. If, on the other hand, it is an internal concept of validity, then the answer depends on the logic that we are talking about and it will be very probably quite easy to give. And as it does not seem to be the case that Russell poses the question of the validity of logical law as either an empirical or a trivial one, it is likely to still be some different kind of validity she has in mind. But she does not explain this.

Can we cut this Gordian knot by saying that the instances are neither sentences of English nor those of an artificial language (expressed in a less perfect way by the former and perhaps in the more perfect ones by the latter)? Can they be some non-linguistic propositions or some logical forms as considered previously? Considerations presented in the previous section indicate that the answer would be – not really. For there is no way to present a proposition or a logical form save by means of a sentence. And if we say that the sentence, say, *It rains or it does not rain* is a mere proxy for a proposition, then the proposition may be either *the proposition expressed by "It rains or it does not rain," whatever it is*, or it can be a proposition identified somehow independently of the contingent meaning of *It rains or it does not rain* (perhaps by a sentence of an artificial language that we have devised to express the propositions rigidly and unambiguously). However, in the former case we are back at an empirical investigation while in the latter one we embrace the picture of the artificial language presenting perfect conceptual or logical structures free from any fuzziness brought in by the empirical languages rejected previously.[10]

Consider a paper by Gila Sher (2011). She writes: "An inference/consequence, **I**, is a pair, <X,Y>, where X is a sentence (set of sentences) and Y is a sentence, such that if X is true (all the sentences in X are true), Y is true" (p. 253), giving the example <"New York City is in the United States," "London is in Britain">, indicating that what X and Y are supposed to range over in the previous definition are English sentences. Then she continues (p. 255):

> [A logic] should not claim that an inference <X,Y> is logically valid unless this claim is true, i.e., unless <X,Y> is in fact logically valid, that is, unless <X,Y> in fact transmits truth from X to Y with an especially strong modal force.

Again, which concept of validity is in play here? Due to the example, it would seem that it is the validity with respect to English. But again,

10. See Peregrin (2010b) for a more detailed discussion of the options.

English sentences are empirical items, established as such in the course of the evolution of certain groups of speakers and meaning what they do as a result of this purely contingent evolution. Hence, to find out whether <X,Y> is an "inference," let alone a "logically valid" one, is to carry out an empirical investigation of English. And, again, this is not mere nitpicking, resulting from neglecting the fact that sentences like "New York City is in the United States" can be taken as merely proxies for corresponding propositions; because propositions are accessible only via (such or another) sentences and the problems with the sentences carry over to them.

This is not to say that ambiguities of the kind that I have just pointed out can never be seen as the result of a certain tolerable sloppiness, or that they can never be explained away; however, at least in some cases, they effectively blur the problems they set out to elucidate.

16.6 Logic Between Mathematics and Empirical Research

The considerations in this chapter are based on the observation that concepts employed by logical theories are either internal to the apparatus of logic (these are the concepts with the subscript F we used in previous chapters) or external to it – concerning the relationship between the apparatus and our reasoning and our language as its vehicle (the concepts with the subscript M). The problem is that the internal concepts are just products of our definitions (and investigating the consequences of our definitions can be, in the best case, pure mathematics), while the external ones are something concerning the empirical world. Hence, we seem to be caught in a dilemma: we can either do mathematics or empirical research. But logic seems to be neither a matter of pure mathematics nor an empirical theory. The laws of logic do not seem to be either pure artifacts of our definitions or a matter of empirical generalizations.

The trouble, as I see it, is that this dilemma is often "overcome" by the tacit assumption that they are both, in the sense that they concern a specific kind of reality that is neither purely mathematical, nor purely empirical – that it is an artifact of definitions but is at the same time somehow inherently "about" the real world, namely about our *de facto* reasoning in the world. In this way logic appears to be a matter of the Kantian "synthetic *a priori*": it concerns a realm where we can make *discoveries*, though not as a result of empirical investigations and not by deriving them from our stipulations. We discover them by some peculiar expeditions into a landscape of our reason, a landscape which is not accessible to our science and which thus remains mysterious.

This is not to say that most logicians would state their views in this way. However, when they treat a logical law like the excluded middle, $\phi\vee\neg\phi$, they do not seem to think that this is either just something concerning a particular mathematical structure (like the structure of classical logic) or something concerning an empirical entity, like a natural language. They think this is something that is not discovered by empirical generalization, which, nevertheless, holds of the empirical items in terms of which we *de facto* reason.

I suggest a different way of overcoming the dilemma. I suggest that logic does not concern any specific reality that is both mathematical and empirical; I am convinced that it concerns a superimposition of these two realities; a superimposition that we know so well from our ordinary science sketched previously: logic is largely a study of models (produced by our definitions) that, however, must be arguably adequate to the ultimate subject matter of logic – our reasoning and the languages we use to argue and reason.[11] Therefore, there is the "mathematical" dimension, consisting in devising, building, maintaining, and upgrading the models, and there is the "empirical" dimension that consists in taking care of the models being adequate to what they model.

One consequence of this view of logic is that there is nothing like an "a priori logic." We can do a lot of logic by studying various mathematical structures (and perhaps even call this investigation "a priori"); but this is only because these structures already have turned out to be useful for the study of reasoning. We can contribute to logic by studying, say, Boolean algebras, but this is only because the study of the actual logical connectives we use to reason were discovered to display something close to this structure. Hence, we should not mistake the fact of having an adequate model – we can do a lot of useful work staying inside the model – for the fact that these kinds of structures, the study of which amounts to doing logic, are delivered to us somehow by our reason alone.

11. It is perhaps worth noticing that there is no unique concept of adequateness – it is the purpose of the specific analysis in question that matters.

17 Conclusion

Philosophical Problems of Logical Systems

I think that the way in which logic, in the course of the twentieth century, shifted its attention from natural languages to artificial languages completely changed its nature, and that this change has not been reflected in all of its complexity yet. The new philosophical questions created by this shift are numerous and to a large extent unprecedented, so their exploration is no easy task.

The naive view is that we have got rid of the messy and blurry natural language in favor of the orderly and transparent formal languages of logic. The aim of this book was to indicate that things are not, by far, that simple. While natural languages are natural repositories of meanings, tried and tested by the millennia that they have been serving us, our artificial languages are mere *simulacra*, which mostly depend, for their usefulness, on their proximity to their natural counterparts. (This is not to say that an artificial language could not become fully meaningful without the support of the natural one as a matter of principle; but practically speaking this would require a long and complicated development.)

True, the emergence of the artificial languages of logic boosted the enterprise of logic in an unprecedented way – it helped us see, clearly and distinctly, many important facts regarding reasoning and argumentation in a way that had never been possible before. However, the accelerated enterprise of logic also took a new, unprecedented direction: it not only offered new answers to old questions, but also posed brand new questions that, sometimes, had little to do with the old ones. And unfortunately, due to a lack of philosophical reflection, it was sometimes mistakenly taken for granted that the new answers belong to the old questions.

I think, and I argued for this throughout the book, that the most satisfactory view of the role of the artificial languages within modern logics takes these languages to be a sort of models, not dissimilar to scientific models of empirical phenomena. True, they are not literally scientific models, especially because of their normativity, but what they share with these models is the asymmetric relation between the investigated phenomenon (which in the case of logic is the "natural" following-from with natural language as its essential vehicle) and the model, which is a tool of the investigation of the phenomenon, not an incarnation of its true essence.

References

Ajdukiewicz, K. (1935): "Die syntaktische Konexität." *Studia Philosophica* 1, 1–27.

Anderson, A. R. & Belnap, N. D. (1975): *Entailment: The Logic of Relevance and Necessity, vol. 1*. Princeton, NJ: Princeton University Press.

Anderson, A. R., Belnap, N. D. & Dunn, J. M. (1992): *Entailment: The Logic of Relevance and Necessity, vol. 2*. Princeton, NJ: Princeton University Press.

Arazim, P. (2017): "Expressivist Perspective on Logicality." *Logica Universalis* 11, 409–419.

Aristotle: *Prior Analytics*; quoted from the English translation (Jenkinson, A. J.) available online at http://classics.mit.edu//Aristotle/prior.html

Arnauld, A. & Nicole, P. (1662): *La logique, ou l art de penser (Port Royal Logic)*. Paris.

Bar-Hillel, Y. (1950): "On Syntactical Categories." *The Journal of Symbolic Logic* 15, 1–16.

Barcan, R. C. (1946): "A Functional Calculus of First Order Based on Strict Implication." *The Journal of Symbolic Logic* 11, 1–16.

Barendregt, H. (1980): *The Lambda Calculus*. Amsterdam: North-Holland.

Barendregt, H. (1992): "Lambda Calculi with Types." *Handbook of Logic in Computer Science, vol. 2* (ed. Abramsky, S. et al.), pp. 117–309. Oxford: Clarendon Press.

Barwise, J. & Cooper, R. (1981): "Generalized Quantifiers and Natural Language." *Linguistics and Philosophy* 4, 159–219.

Beaney, M. (2004): "Carnap's Conception of Explication: From Frege to Husserl?" *Carnap Brought Home: The View from Jena* (ed. Awodey, S. & Klein, C.), pp. 117–150. Chicago: Open Court.

Belnap, N. D. (1962): "Tonk, Plonk and Plink." *Analysis* 22, 130–134.

Béziau, J.-Y. (1994): "Universal Logic." *Logica '94 - Proceedings of the 8th International Symposium* (ed. Childers, T. & Majer, O.), pp. 73–93. Prague: FLÚ AV ČR.

Béziau, J.-Y. & Carnelli, W. A., eds. (2006): *Paraconsistent Logic with No Frontiers*. Amsterdam: Elsevier.

Birkhoff, G. (1935): "On the Structure of Abstract Algebras." *Proceedings of The Cambridge Philosophical Society* 31, 433–454.

Boole, G. (1848): "The Calculus of Logic." *Cambridge and Dublin Mathematical Journal* 3, 183–198.

Boolos, G. S., Burgess, J. P. & Jeffrey, R. C. (2002): *Computability and Logic* (4th ed.). Cambridge: Cambridge University Press.

Brandom, R. (1994): *Making It Explicit: Reasoning, Representing, and Discursive Commitment*. Cambridge, MA: Harvard University Press.
Brandom, R. (2000): *Articulating Reasons*. Cambridge, MA: Harvard University Press.
Brun, G. (2003): *Die richtige Formel: Philosophische Probleme der logischen Formalisierung*. Frankfurt am Main: Ontos.
Brun, G. (2014): "Reconstructing Arguments: Formalization and Reflective Equilibrium." *Logical Analysis and History of Philosophy* 17, 94–129.
Brun, G. (2016): "Explication as a Method of Conceptual Re-engineering." *Erkenntnis* 81, 1211–1241.
Brun, G. (forthcoming): "Conceptual Re-engineering: From Explication to Reflective Equilibrium." *Synthese*.
Button, T. & Walsh, S. (2018): *Philosophy and Model Theory*. Oxford: Oxford University Press.
Carnap, R. (1934): *Logische Syntax der Sprache*. Vienna: Springer.
Carnap, R. (1942): *Introduction to Semantics*. Cambridge, MA: Harvard University Press.
Carnap, R. (1947): *Meaning and Necessity*. Chicago, IL: University of Chicago Press.
Chang, C. & Keisler, H. (1973): *Model Theory*. Amsterdam: North-Holland.
Church, A. (1941): *The Calculi of Lambda-conversion*. Princeton, NJ: Princeton University Press.
Coffa, A. (1991): *The Semantic Tradition from Kant to Carnap*. Cambridge: Cambridge University Press.
Cohn, P. M. (1981): *Universal Algebra*. Dordrecht: Reidel.
Cook, R. T. (2005): "What's Wrong with Tonk (?)." *Journal of Philosophical Logic* 34, 217–226.
Cook, R. T. (2010): "Let a Thousand Flowers Bloom: A Tour of Logical Pluralism." *Philosophy Compass* 5, 492–504.
Copeland, B. J. (2002): "The Genesis of Possible Worlds Semantics." *Journal of Philosophical Logic* 31, 99–137.
Cresswell, M. J. (1973): *Logic and Languages*. London: Methuen.
Curry, H. B., Feys, R. & Craig, W. (1958): *Combinatory Logic, vol. 1*. Amsterdam: North-Holland.
Davidson, D. (1967): "The Logical Form of Action Sentences." *The Logic of Decision and Action* (ed. Rescher, N.). Pittsburgh: University of Pittsburgh; reprinted in Davidson: *Essays on Actions and Events*, pp. 105–122. Oxford: Clarendon Press, 1980.
Davidson, D. (1979): "The Inscrutability of Reference." *Southwestern Journal of Philosophy 10*; reprinted in Davidson: *Inquiries into Truth and Interpretation*, pp. 227–241. Oxford: Clarendon Press, 1984.
Dotterer, R. H. (1918): "The Definition of Infinity." *The Journal of Philosophy, Psychology and Scientific Methods* 15, 294–301.
Du, D.-Z. & Ko, K.-I. (2014): *Theory of Computational Complexity* (2nd ed.). Hoboken: Wiley.
Dunn, J. M. & Hardegree, G. M. (2001): *Algebraic Methods in Philosophical Logic*. Oxford: Oxford University Press.
Dutilh Novaes, C. (2012). *Formal Languages in Logic: A Philosophical and Cognitive Analysis*. Cambridge: Cambridge University Press.

Fitch, F. B. (1952): *Symbolic Logic*. New York: Ronald.

Fodor, J. A. (2008): *LOT2: The Language of Thought Revisited*. Oxford: Oxford University Press.

Font, J. M. & Jansana, R. (1996): *A General Algebraic Semantics for Sentential Logics*. Dordrecht: Springer.

Font, J. M., Jansana, R. & Pigozzi, D. (2003): "A Survey of Abstract Algebraic Logic." *Studia Logica* 74, 13–97.

Francez, N. (2015): *Proof-theoretic Semantics*. London: College Publications.

Frege, G. (1879): *Begriffsschrift*, Halle: Nebert; quoted from the English translation (Bauer-Mengeberg, S.). "Begriffsschrift." *From Frege to Gödel: A Source Book from Mathematical Logic* (ed. van Heijenoort, J.), pp. 1–82. Cambridge, MA: Harvard University Press, 1967

Frege, G. (1892a): "Über Begriff und Gegenstand." *Vierteljahrschrift für Wissentschaftliche Philosophie* 16, 192–205.

Frege, G. (1892b): "Über Sinn und Bedeutung." *Zeitschrift für Philosophie und philosophische Kritik* 100, 25–50.

Frege, G. (1918): "Der Gedanke." *Beiträge zur Philosophie des deutschen Idealismus* 2, 58–77.

Gaifman, C. (2004): "Non-standard Models in a Broader Perspective." *Non-standard Models of Arithmetic and Set Theory* (ed. Enayat, E. & Kossak, R.), pp. 1–22. Providence: American Mathematical Society.

Gallin, D. (1975): *Intensional and Higher-order Modal Logic*. Amsterdam: North-Holland.

Garson, J. W. (2013): *What Logics Mean*. Cambridge: Cambridge University Press.

Gentzen, G. (1935a): "Untersuchungen über das logische Schließen. I." *Mathematische Zeitschrift* 39, 176–210.

Gentzen, G. (1935b): "Untersuchungen über das logische Schließen. II." *Mathematische Zeitschrift* 39, 405–431.

Girard, J.-Y. (1987): "Linear Logic." *Theoretical Computer Science* 50, 1–101.

Gödel, K. (1930): "Die Vollständigkeit der Axiome des logischen Funktionenkalküls." *Monatshefte für Mathematik und Physik* 37, 349–360.

Gödel, K. (1931): "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I." *Monatshefte für Mathematik und Physik* 38, 173–198.

Goguen, J. A., Thatcher, J. W., Wagner, E. G. & Wright, J. B. (1977): "Initial Algebra Semantics and Continuous Algebras." *Journal of the ACM* 24, 68–95.

Goldblatt, R. (1984): *Topoi: The Categorial Analysis of Logic* (Rev. ed.). Amsterdam: North-Holland.

Goldfarb, W. D. (1979): "Logic in the Twenties: The Nature of the Quantifier." *The Journal of Symbolic Logic* 44, 351–368.

Goodman, N. (1983): *Fact, Fiction, and Forecast*. Cambridge, MA: Harvard University Press.

Grattan-Guinness, I. (2000): *The Search for Mathematical Roots, 1870-1940: Logics. Set Theories and the Foundations of Mathematics from Cantor through Russell to Gödel*. Princeton, NJ: Princeton University Press.

Grim, P. (1991): *The Incomplete Universe*. Cambridge, MA: MIT Press.

Haaparanta, L., ed. (2009): *The Development of Modern Logic*. Oxford: Oxford University Press.

Hacking, I. (1979): "What Is Logic?" *The Journal of Philosophy* 76, 285–319.
Hardegree, G. M. (2005): "Completeness and Super-Valuations." *Journal of Philosophical Logic* 34, 81–95.
Henkin, L. (1950): "Completeness in the Theory of Types." *The Journal of Symbolic Logic* 15, 81–91.
Henkin, L. & Monk, J. D. (1974): "Cylindric Algebras and Related Structures." *Proceedings of the Tarski Symposium* (ed. Henkin, L. et al.), pp. 105–152. Providence: American Mathematical Society.
Hilbert, D. & Ackermann, W. (1928): *Grundzüge der theoretischen Logik*. Berlin: Springer.
Hodges, W. (1993): *Model Theory*. Cambridge: Cambridge University Press.
Hodges, W. (2016): "A Strongly Differing Opinion on Proof-theoretic Semantics?" *Advances in Proof-theoretic Semantics* (ed. Piecha, T. & Schroeder-Heister, P.). Cham: Springer.
Janssen, T. M. (1986): *Foundations and Applications of Montague Grammar*. Amsterdam: Mathematisch Centrum.
Jönsson, M. L. (2017): "Interpersonal Sameness of Meaning for Inferential Role Semantics." *Journal of Philosophical Logic* 46, 269–297.
Kalmár, L. (1935): "Über die Axiomatisierbarkeit des Aussagenkalküls." *Acta Scientiarum Mathematicarum* 7, 222–243.
Kripke, S. (1963a): "Semantical Analysis of Modal Logic I (Normal Modal Propositional Calculi)." *Zeitschift Für Mathematische Logik Und Grundlagen Der Mathematik* 9, 67–96.
Kripke, S. (1963b): "Semantical Considerations on Modal Logic." *Acta Philosophica Fennica* 16, 83–94.
Kripke, S. (1965a): "Semantical Analysis of Intuitionistic Logic." *Formal Systems and Recursive Functions* (ed. Crossley, J. & Dummett, M.), pp. 92–130. Amsterdam: North-Holland.
Kripke, S. (1965b): "Semantical Analysis of Modal Logic II (Non-Normal Modal Propositional Calculi)." *The Theory of Models* (ed. Addison, J. W. et al.), pp. 206–220. Amsterdam: North-Holland.
Kripke, S. (1972): "Naming and Necessity." *Semantics of Natural Language* (ed. Davidson, D. & Harman, G.), pp. 253–355. Dordrecht: Reidel; later published as a book.
Lewis, C. I. (1912): "Implication and the Algebra of Logic." *Mind* 21, 522–531.
Lewis, C. I. (1917): "The Issues Concerning Material Implication." *The Journal of Philosophy, Psychology and Scientific Methods* 14, 350–356.
Lewis, C. I. & Langford, C. H. (1932): *Symbolic Logic*. New York: Century.
Lewis, D. (1972): "General Semantics." *Semantics of Natural Language* (ed. Davidson, D. & Harman, G.), pp. 169–218. Dordrecht: Reidel.
Lloyd, G. E. R. (1978): "Saving the Appearances." *The Classical Quarterly* 28, 202–222.
Loebner, S. (1987): "Natural Language and Generalized Quantifier Theory." *Generalized Quantifiers* (ed. Gärdenfors, P.), pp. 181–201. Dordrecht: Reidel.
Lotze, H. (1874): *Logik; drei Bücher, vom Denken, vom Untersuchen und vom Erkennen*. Leipzig: Hirzel.
Mal'cev, A. (1941): "Об одном общем методе получения локальных теорем теории групп (On a general method for obtaining local theorems in group theory)." Учен. Зап. Ивановск. Пед. Ин-Та 1, 3–9.

Mendelson, E. (1964): *Introduction to Mathematical Logic*. Pacific Grove: Wadsworth & Brooks.

Mercier, H. & Sperber, D. (2017): *The Enigma of Reason*. Cambridge, MA: Harvard University Press.

Montague, R. (1973): "The Proper Treatment of Quantification in Ordinary English." *Approaches to Natural Language* (ed. Hintikka, J. et.al.), pp. 221–242. Dordrecht: Reidel.

Montague, R. (1974): *Formal Philosophy: Selected Papers of R. Montague*. New Haven: Yale University Press.

Moore, G. H. (1980): "Beyond First-order Logic: The Historical Interplay between Mathematical Logic and Axiomatic Set Theory." *History and Philosophy of Logic* 1, 95–137.

Mueller, I. (1969): "Euclid's Elements and the Axiomatic Method." *The British Journal for the Philosophy of Science* 20, 289–309.

Oehrle, R. T., Bach, E. & Wheeler, D., ed. (1988): *Categorial Grammars and Natural Language Structures*. Dordrecht: Reidel.

Paoli, F. (2013): *Substructural Logics: A Primer*. Dordrecht: Springer.

Peano, G. (1897): "Logique mathématique." *Formulaire de mathématiques, vol. 2*. Torino: Bocca.

Peregrin, J. (1995): *Doing Worlds with Words: Formal Semantics without Formal Metaphysics*. Dordrecht: Kluwer.

Peregrin, J. (1999): "The Pragmatization of Semantics." *The Semantics/Pragmatics Interface from Different Points of View* (ed. Turner, K.), pp. 419–442. North-Hollard: Elsevier.

Peregrin, J. (2000a): "Reference and Inference." *Reference and Anaphoric Relations* (ed. von Heusinger, K. & Egli, U.), pp. 269–286. Dordrecht: Kluwer.

Peregrin, J. (2000b): "The Logic of Anaphora." *The Logica Yearbook 1999* (ed. Childers, T.), pp. 191–205. Praha: Filosofia.

Peregrin, J. (2000c): "The 'Natural' and the 'Formal.'" *Journal of Philosophical Logic* 29, 75–101.

Peregrin, J. (2006a): "Extensional vs. Intensional Logic." *Handbook of the Philosophy of Science, vol. 5: Philosophy of Logic* (ed. Jacquette, D.), pp. 831–860. Amsterdam: Elsevier.

Peregrin, J. (2006b): "Semantics as Based on Inference." *The Age of Alternative Logics* (ed. van Benthem, J. et al.), pp. 25–36. Dordrecht: Kluwer; corrected version available from http://jarda.peregrin.cz/mybibl/PDFTxt/498.pdf.

Peregrin, J. (2008): "What Is *the* Logic of Inference?" *Studia Logica* 88, 263–294.

Peregrin, J. (2010a): "Inferentializing Semantics." *Journal of Philosophical Logic* 39, 255–274.

Peregrin, J. (2010b): "Logic and Natural Selection." *Logica Universalis* 4, 207–223.

Peregrin, J. (2010c): "The Myth of Semantic Structure." *Philosophy of Language and Linguistics, vol. I: The Formal Turn* (ed. Stalmaszczyk, P.), pp. 183–197. Frankfurt am Main: Ontos.

Peregrin, J. (2014): *Inferentialism: Why Rules Matter*. Basingstoke: Palgrave Macmillan.

Peregrin, J. (2015): "Logic Reduced to Bare (Proof-theoretical) Bones." *Journal of Logic, Language and Information* 24, 193–209.

Peregrin, J. (2019): "Logic as a (Natural) Science." *The Logica Yearbook 2018* (ed. Sedlár, I. & Blicha, M.), pp. 177–196. London: College Publications.

Peregrin, J. & Svoboda, V. (2013): "Criteria for Logical Formalization." *Synthese* 190, 2897–2924.

Peregrin, J. & Svoboda, V. (2016): "Logical Formalization and the Formation of Logic(s)." *Logique et Analyse* 233, 55–80.

Peregrin, J. & Svoboda, V. (2017): *Reflective Equilibrium and the Principles of Logical Analysis: Understanding the Laws of Logic*. New York: Routledge.

Prawitz, D. (1965): *Natural Deduction: A Proof-theoretical Study*. Strockholm: Almqvist & Wiksell.

Prawitz, D. (1974): "On the Idea of a General Proof Theory." *Synthese* 27, 63–77.

Priest, G. (1979): "The Logic of Paradox." *Journal of Philosophical Logic* 8, 219–241.

Prior, A. N. (1960): "The Runabout Inference-Ticket." *Analysis* 21, 38–39.

Punčochář, V. (2012): "Some Modifications of Carnap's Modal Logic." *Studia Logica* 100, 517–543.

Putnam, H. (1975): "The Meaning of 'Meaning.'" *Studies in the Philosophy of Science, vol. 7: Language, Mind and Knowledge* (ed. Gunderson, K.). Minneapolis: University of Minnesota Press; reprinted in Putnam: *Mind, Language and Reality*, pp. 215–271. Cambridge: Cambridge University Press, 1979.

Quine, W. V. O. (1953): *From a Logical Point of View*. New York: Harper & Row.

Quine, W. V. O. (1960): "Variables Explained Away." *Proceedings of the American Philosophical Society* 104, 343–347.

Quine, W. V. O. (1968): "Ontological Relativity." *The Journal of Philosophy* 65, 185–212.

Quine, W. V. O. (1986): *Philosophy of Logic*. Cambridge, MA: Harvard University Press.

Raatikainen, P. (2008): "On Rules of Inference and the Meanings of Logical Constants." *Analysis* 68, 282–287.

Rawls, J. (1971): *A Theory of Justice*. Cambridge, MA: Harvard University Press.

Rayo, A. & Uzquiano, G., eds. (2006): *Absolute Generality*. Oxford: Oxford University Press.

Restall, G. (2000): *Introduction to Substructural Logics*. London: Routledge.

Restall, G. (2013): "Assertion, Denial and Non-Classical Theories." *Paraconsistency: Logic and Applications* (ed. Tanaka, K. et al.), pp. 81–99. Dordrecht: Springer.

Ripley, D. (2017): "Bilateralism, Coherence, Warrant." *Act-Based Conceptions of Propositional Content* (ed. Moltmann, F. & Textor. M.), pp. 307–324. Oxford: Oxford University Press.

Rodriguez-Pereyra, G. (1999): "Leibniz's Argument for the Identity of Indiscernibles in his Correspondence with Clarke." *Australasian Journal of Philosophy* 77, 429–38.

Rorty, R., ed. (1967): *The Linguistic Turn*. Chicago: University of Chicago Press.

Rosser, B. (1936). "Extensions of Some Theorems of Gödel and Church." *Journal of Symbolic Logic* 1, 87–91.

Russell, B. (1905): "On Denoting." *Mind* 14, 479–493.

Russell, G. (2015): "The Justification of the Basic Laws of Logic." *Journal of Philosophical Logic* 44, 793–803.

Saussure, F. de (1931): *Cours de linguistique générale*. Paris: Payot; English translation *Course in General Linguistics*. New York: Philosophical Library, 1959.

Scanlan, M. (2010): "American Postulate Theorists and Alfred Tarski." *History and Philosophy of Logic* 24, 307–325.

Schönfinkel, M. (1924): "Über die Bausteine der mathematischen Logik." *Mathematische Annalen* 92, 305–316.

Stalnaker, R. (1986): "Possible Worlds and Situations." *Journal of Philosophical Logic* 15, 109–123.

Shapiro, S. (1991): *Foundations without Foundationalism*. Oxford: Clarendon Press.

Shapiro, S. (1998): "Logical Consequence: Models and Modality." *Philosophy of Mathematics Today* (ed. Schirn, M.), pp. 131–156. Oxford: Oxford University Press.

Sher, G. (2011): "Is Logic in the Mind or in the World?" *Synthese* 181, 353–365.

Shoenfield, J. R. (1967): *Mathematical Logic*. Reading: Addison-Wesley.

Schurz, G. (2001): "Rudolf Carnap's Modal Logic." *Zwischen traditioneller und moderner Logik. Nichtklassische Anstze* (ed. Stelzner, W. & Stoeckler, M.), pp. 365–380. Paderborn: Mentis.

Smith, P. (2013): *An Introduction to Gödel's Theorems* (2nd ed.). Cambridge: Cambridge University Press.

Stekeler-Weithofer, P. (1986): *Grundprobleme der Logik*. Berlin: de Gruyter.

Stone, M. H. (1936): "The Theory of Representation for Boolean Algebras." *Transactions of the American Mathematical Society* 40, 37–111.

Svoboda, V. & Peregrin, J. (2009): *Od jazyka k logice*. Praha: Academia.

Svoboda, V. & Peregrin, J. (2016): "Logically Incorrect Arguments." *Argumentation* 30, 263–287.

Tarski, A. (1932): "Der Wahrheitsbegriff in den Sprachen der deduktiven Disziplinen." *Anzeiger Der Osterreichischen Akademie Der Wissenschaften, Mathematisch-Naturwissenschaftliche Klasse* 69, 23–25; English translation *The Concept of Truth in Formalized Languages* in Tarski: *Logic, Semantics, Metamathematics*, pp. 15–278. Oxford: Clarendon Press, 1956.

Tarski, A. (1935): "Der Wahrheitsbegriff in den formalisierten Sprachen." *Studia Philosophica* 1, 261–405.

Tarski, A. (1936): "O pojeciu wynikania logicznego." *Przeglad Filozoficzny* 39, 58–68; English translation "On the Concept of Following Logically." *History and Philosophy of Logic* 23, 2000, 155–196.

Tarski, A. (1944): "The Semantic Conception of Truth." *Philosophy and Phenomenological Research* 4, 341–375.

Tichý, P. (1988): *The Foundations of Frege's Logic*. Berlin: de Gruyter.

Toulmin, S. (2003): *The Uses of Argument* (updated ed.). Cambridge: Cambridge University Press.

van Benthem, J. (1984): "Questions about Quantifiers." *Journal of Symbolic Logic* 49, 443–466.

Waismann, F. (1968): "Verifiability." Waismann: *How I See Philosophy*, pp. 39–66. Berlin: Springer.

Wansing, H. (2006): "Connectives Stronger than Tonk." *Journal of Philosophical Logic* 35, 653–660.

Whitehead, A. N. & Russell, B. (1910–13): *Principia Mathematica*. Cambridge: Cambridge University Press.

Zach, R. (2003): "The Practice of Finitism: Epsilon Calculus and Consistency Proofs in Hilbert's Program." *Synthese* 137, 211–259.

Name Index

Subject Index