Acknowledgments

It has been my great fortune to have had many chances to present different parts of this work to various kinds of audience. I received very useful comments and encouragement from participants in the Indiana University Logic Group seminar, in a workshop at Logic in Computer Science, in the Thinking with Diagrams workshop (University of Wales), in the University of Virginia Philosophy Department colloquium, and at the First International Conference on the Theory and Application of Diagrams (University of Edinburgh). This work especially benefited from discussions with Gerry Allwein, Peter Cheng, George Furnas, John Howse, Robert Lindsay, Patrick Scotto di Luzio, Bernd Meyer, Arthur Miller, Fernando Molina, Jesse Norman, Peter Ochs, Nik Swoboda, John Taylor, and Dirk Van Gucht.

Jim Cargile's and Paul Humphreys' helpful comments and strong support for my work were absolutely crucial at the final stage of the work while I was spending a sabbatical at the University of Virginia. I truly appreciate the careful reading and extremely valuable suggestions by Keith Stenning, Len Olsen, Richard Shedenhelm, and anonymous reviewers of the MIT Press. Carolyn Anderson, Michael Sims, and Alan Thwaits at the MIT Press have been a great help.

I acknowledge financial support from an NEH Summer Fellowship, a Faculty Research Award from the University of Notre Dame, and financial assistance from the Philosophy Department of the University of Notre Dame. I also greatly appreciate the kind assistance I received from Houghton Library, Harvard University, during numerous visits I paid while working on the manuscript.

Throughout every stage of the work, Jon Barwise's, John Etchemendy's, Jaegwon Kim's, and John Perry's encouragement and moral support have been very important to me. As always, my special thanks go to my mother, who has been praying for me in Korea, and to my husband, Henry Smith, whose love, help, and support have been beyond description.

1 Introduction

A fresh approach to reasoning and representation has been emerging at the crossroads of the philosophy of mind, cognitive science, logic, and computer science.[1] Our ordinary reasoning typically involves information obtained through more than one medium—sentences, diagrams, smells, sounds, and so on. Recognizing the actual practice of this *multi-modal* reasoning, researchers have started focusing on multi-modal, or *heterogeneous*, representation systems, which employ both symbolic and diagrammatic elements. This is a clear departure from the major direction taken by logicians and mathematicians since the development of modern logic: For more than a century, symbolic representation systems have been the exclusive subject for formal logic.

This book analyzes a well-known, but much-criticized, non-symbolic representation system, Peirce's system of Existential Graphs (henceforth, 'EG'), and presents a new approach to EG based on the discovery of its unique nature and on the reconstruction of Peirce's theory of representation. I will explore specific differences between symbolic and diagrammatic systems both in reading-algorithms and in the formulation of inference rules, and I will draw out the implications of these results for several long-standing debates in various disciplines that study multi-modal systems. Before locating this work within the overall picture of research into heterogeneous reasoning, it will be useful to re-examine two related major assumptions which underlie the general heterogenous reasoning project. One is the shared view that there has until quite recently been a long-standing prejudice against non-symbolic representation in logic, mathematics, and computer science. The other is the assumed distinction between symbolic and diagrammatic systems. I do share the view that there is a prevailing bias against non-symbolic systems, and I do assume that symbols are different from diagrams. However, I believe that in order to achieve further fruitful results the time has come to raise more fundamental questions about the main assumptions behind this fast-growing body of research into heterogeneous systems.

In the present work, I would like to emphasize the following circular relation between these two points—the strong preference for symbolic over diagrammatic systems and the distinction between these two types of systems. Without a solid theoretical background for the distinction between symbolic and diagrammatic systems, we easily overlook different kinds of strength and weaknesses that each type of system possesses. Then, given the general predominance of symbolic systems in the study of logic, we tend to try to understand and evaluate diagrammatic systems against the criteria of a symbolic system. As a result, without their own strengths being discovered, diagrammatic systems have been criticized mainly because they lack the properties of a symbolic system. In turn, this unfortunate result—stemming partly from a prejudice against diagrams and partly from an unclear distinction between symbols and diagrams—only reinforces the existing prejudice against non-symbolic systems. I claim that these two main phenomena reinforce each other in a vicious circle.

Where do we need to step in to break this circle? First of all, we should stop evaluating diagrammatic systems in terms of symbolic systems. This would be possible only if we could develop an independent way of approaching diagrams. Again, an independent method cannot become available without identifying the positive properties that diagrammatic systems uniquely possess.

In this book, I examine Charles S. Peirce's "Existential Graphs" as a case study to support my claim that a prejudice against diagrammatic systems reinforced a superficial distinction between symbols and diagrams and vice versa, and to show how this vicious circle can be broken. Let me briefly explain below why Peirce's graphical system provides us with an exceptionally fitting case study for our present inquiry.

At the dawn of modern logic, Charles S. Peirce invented two different types of logical systems, one symbolic and the other graphical. Why has the symbolic system absolutely dominated the other in the subsequent development of logic, despite the fact that Peirce himself considered EG superior to his symbolic logical system?[2] We should be careful not to answer this question by citing the existing prejudice against diagrammatic systems. That would not explain much about the nature of the prejudice we are interested in investigating. More importantly, complaints against EG are different from complaints against the use of diagrams in proofs. Diagrams, critics say, are not rigorous enough to be used in a proof or tend to mislead us in a proof. Accordingly, diagrams are used with caution and only as heuristic aids. However, in the case of EG, logicians accept that EG is sound and complete. In spite of its rigor, EG has not been adopted for deductive reasoning. In this case, the criticism is more specific: The graphs of EG are difficult to read off, and the rules of EG difficult to use.

Why does a translation of a graph of EG almost always result in a complicated-looking sentence? Why is EG harder to use than natural deductive systems in spite of the apparent similarity between how these two systems are set up? Surprisingly, I find the same source responsible for these two puzzles: Despite the fact that EG is known to be a graphical system, the iconic features of EG have not been fully implemented either in any existing reading algorithm or in how the transformation rules are stated. This deficiency has yielded complicated and unintuitive reading methods and produced transformation rules that are not fine-grained enough to be used conveniently. A somewhat ironic, trivial-sounding but unobserved lesson follows: When a system is heterogeneous or diagrammatic, its iconic features should be fully utilized both in its syntax and semantics. Otherwise, many kinds of inefficiency will arise. Therefore, the existing criticisms against EG are the necessary product of the traditional, less then fully iconic way of approaching EG, and I argue that my new ways of understanding EG disarm the main criticisms of EG.

What, then, is the main difference between the traditional method and the new method I will present in this book? I will show that a strong dominance of symbolic over diagrammatic systems is present in how we traditionally have understood EG: Graphs of EG are read off very much as symbolic sentences are, and the inference rules of EG are understood as being like the rules of natural deductive systems. This misguided assimilation of EG into a symbolic system, I claim, is the most fundamental source for the existing complaints against EG, which, in turn, supports a bias against non-symbolic systems. Here we confirm the existence of the vicious circle pattern. In order to correct this situation, I will rely on my discovery of the fundamental differences between EG and a logically equivalent symbolic system in two major respects.

First, I demonstrate how differently a meaningful unit of each system can be read off. In order to prevent ambiguity, the semantic interpretation of a symbolic sentence requires its unique readability, and hence no possibility of multiple readings. On the other hand, in the case of EG, multiple readings do not generate ambiguity. Moreover, multiple readings for one and the same graph are the most *natural* since we perceive a graph differently depending on how we carve it up. Importantly, we may obtain a comprehensive algorithm of multiple readings when we utilize more fine-grained visual features present in graphs. Second, I claim that the *naturalness* of the inference rules of a system is a relative concept. In the case of a natural deductive system, inference rules are stated so as to reflect the derivational history of a formula. However, in the case of EG, since so many different derivations are available for a graph, we should free the transformation rules of EG from any derivational history of a graph and look for a type of naturalness appropriate to a graphical system.

To find this type of naturalness both in the reading of graphs and in how inference rules are set up, the current work presents a new reading method of graphs and a newly formulated system of EG. Hence, we may meet the traditional objections to EG—that the system is hard to read off and hard to manipulate. Furthermore, this case study offers a new approach to non-symbolic systems in general. It urges heterogenous-system researchers to ask whether they fully utilize visual aspects of a non-symbolic system for a direct and natural reading algorithm, for an efficient formulation of inference rules, and for an intuitive interpretation of these rules. If we do not take advantage of visual distinctions already present in a system, we can only expect a graphical system to be less useful or less intuitive than the symbolic system whose criteria have been used to interpret the graphical system. Unless we adopt independent methods or criteria for a graphical system, we cannot challenge the long-standing prejudice against non-symbolic systems, but only reinforce it.

This project draws on results from various disciplines concerned with diagrams. A distinct role that diagrams or pictures—as opposed to traditional linguistic forms—play in our cognition has been extensively discussed in several different areas, including philosophy, cognitive science, logic, artificial intelligence, and design theory. After outlining how various disciplines pursue this topic from slightly different points of view and how the relation between diagrams and representation (or reasoning) has been emerging as an interdisciplinary topic out of these different approaches, I explain how the project of this book fits in this overall picture.

Among many different approaches to the relation between diagrams (or pictures) and our cognition, I would like to draw attention to an interesting and useful distinction between those approaches that focus on diagrams as internal representations and those that treat them as external representations. In the Introduction to one of the most comprehensive anthologies on this subject, Chandrasekaran, Glasgow, and Narayanan (1995) make the following distinction between internal versus external diagrammatic representations:

- *External diagrammatic representations* These are constructed by the agent in a medium in the external world (paper, etc), but are meant as representations by the agent.
- *Internal diagrams or images* These comprise the (controversial) internal representations that are posited to have some pictorial properties.[3]

From now on, unless I specify otherwise, I will use the words 'diagrams' or 'pictures' to refer to external representations, and 'images' to refer to internal representations. Of course, many do not doubt that these two different levels of representation are closely related to each other.[4] Moreover, this distinction is not needed at all for certain projects, and hence some researchers conflate internal and external representations.[5] However, this distinction will nicely serve as a useful framework which I would like to use to show how different areas falling under the category of research on diagrammatic reasoning are related to one another.

The imagery debate between pictorialists and descriptionalists, one of the most time-honored controversies in psychology, focuses on our internal mental representation: It is about whether picture-like images exist as mental representations.[6] Kosslyn and other pictorialists[7] present a series of experimental data to support their position that some of our mental images are more like pictures than a linear form of language (for example, natural languages or artificial symbolic languages) in some important aspects, even though not all visual mental images and pictures are exactly of the same kind. By contrast, Pylyshyn and other descriptionalists[8] raise questions about the status of picture-like mental images and argue that mental images are formed out of structured descriptions. To them, mental images represent in the manner of language rather than pictures, and hence no picture-like images play an important role in our cognition.[9]

At this point we are not far from philosophical territory—the philosophy of mind. Philosophers' deep interest in mental representation easily goes back to ancient times.[10] Nobody would be surprised to realize that mental images were heatedly discussed during the heyday of ideas.[11] As we know, Hobbes', Locke's, Berkeley's, and Hume's writings concern themselves in large part with mental discourse, the meaning of words, mental images, particular ideas, abstract ideas, impressions, etc. Descartes' well-known distinction between imagining and conceiving something has generated much discussion about the unique role of visual images in mental representations. In the twentieth century, pictorialists in the imagery debate found the modern sense-datum theory in philosophy quite close to their point of view. By the same token, the critics of the sense-datum theory argued that the mistaken pictorial view of mental images arises mainly from our confusion about ordinary language. Not surprisingly, they are sympathetic to the view that mental images are epiphenomena.[12] Contemporary philosophers, mainly in the philosophy of mind,[13] have participated in a recent imagery debate among cognitive scientists.[14]

Being slightly distant from the imagery debate itself, some cognitive scientists have concentrated on the functions of mental images or diagrams in our various cognitive activities, for example, memory,[15] imagination, perception, navigation,[16] inference, problem-solving,[17] etc., instead of exploring the ontological status of internal visual images.[18] Here the distinct nature of "visual information," which is obtained either through internal mental images or through externally drawn diagrams, has become a major topic of research. In particular, research on heterogeneous reasoning focuses on how visual information plays a unique role in our reasoning process. In the following, let me outline three important aspects of this research, all of which are crucial to understand the background of my project.

First, when the distinct role of visual information in inference or reasoning becomes a main question, we find an interesting shift of focus among researchers from internal to external representations, while we do not find a similar tendency in other subareas of research on images and cognition. For example, the study on the relation between memory and visual information focuses on internal as well as external visual information. However, in studies of inference, 'images' or 'diagrams' mainly refers to external representations, i.e., drawn pictures, graphs or diagrams, rather than picture-like mental representations.[19] In Larkin and Simon's classic paper "Why a diagram is (sometimes) worth ten thousand words" (1987), this change was made very clear: "Although our discussion [about the unique nature of diagrams in problem solving] may be relevant to this current controversy about the distinguishability of different internal representations, our analysis explicitly concerns external representations."[20] One benefit from this shift is that although consensus is lacking on whether there are different kinds of internal mental representations (as the long history of the imagery debate shows), everyone agrees that there are different forms of external representations.[21]

Second, research on heterogenous reasoning has attracted much more attention from various different disciplines than has research on the relation between imagery and other cognitive activities—for example, memory or perception—mainly for two reasons. One is that human reasoning is a common topic among cognitive science, logic, mathematics, and artificial intelligence. The other reason for interdisciplinary interest in imagery and reasoning is that externally drawn diagrams or graphs, on which researchers have focused can, be a subject of all of these disciplines, unlike mental images.

Third, research on multi-modal representation has led researchers to explore the differences among different forms of (external) representations, but mainly between diagrammatic and symbolic representations.[22] A strong dominance of symbolic languages in the study of representation systems (since the dawn of modern logic) compels anyone who is working on the relation between diagrams and inference to compare two different kinds of languages, that is, graphical and symbolic systems. In spite of a common interest in heterogeneous representation, various disciplines have pursued the same topics—here the relation between diagrams and reasoning, and the comparison between symbolic and graphical systems—from different points of view. After briefly summarizing the slightly different agendas of various disciplines, I will show how the work presented in this book serves to bridge the gap among them.

It has been a while since cognitive scientists started paying attention to how different forms of representation vary in their cognitive effects on human inference. Many important results have been produced along these lines. Based on Simon's distinction between informational and computational equivalence among representations,[23] Larkin and Simon (1987) present a case study in which two informationally equivalent systems, one sentential and the other diagrammatic, are shown to be computationally non-equivalent. Lindsay (1988) makes a related point by specifying where this computational difference lies. Claiming that an important role of diagrammatic representation[24] in inference is not its expressive power, but its efficiency, he showed that this efficiency is obtained through the special properties that diagrams possess. Constraints built into the diagram-construction processes rule out many trivial cases, and, after the construction is completed, conclusions are directly read off from a diagram.[25] Shimojima (1996) uses term 'free ride' to refer to an inference in which the conclusion seems to be read off almost automatically from the representation of premises. Gurr, Lee, and Stenning (1998) argue that the semantics of a diagrammatic system is more "direct" than the semantics of a symbolic system and that this crucial difference explains their characteristic low cost of reading off a conclusion. They also correctly point out that directness is relative, and hence, some rides are cheaper than others. Having a distinct role of graphs in mind, Wang and Lee (1993) present a formal framework as a guideline for correct visual languages. This impressive work provides design theorists and AI researchers with computational support for visual reasoning.

Not surprisingly, AI researchers, one of whose main concerns is the heuristic power of a representation system in addition to its expressive power, have been debating for decades about different forms of representation.[26] Hence, they have welcomed discussions of the distinct role of visual reasoning and have recently hosted interdisciplinary symposiums on diagrammatic reasoning at AI conferences.[27] At the same time, realizing that human beings adopt different representation forms depending on the kinds of problems they face, some AI researchers and design theorists have practiced domain-specific approaches to bringing in problem-tailored representation forms.[28] Harel's invention of higraphs (1988) is an excellent example of obtaining practical results without being bogged down in a more abstract and theoretical controversy. However, it is likely that both top-down and bottom-up approaches to the distinct role of diagrams in reasoning are necessary for the project to accomplish its goal.

Heterogeneous reasoning has also been taken up by logicians. It is important to note that logicians bring slightly different concerns to this project from those of cognitive scientists or of AI researchers. First, logicians' main interest is exclusively in externally drawn representation systems, as opposed to internal mental representations. Second, differences in cognitive effects or in heuristic power among different forms of representation are not at the top of logicians' agenda. The first and necessary test for any representation system is to prove that the system is correct. The next important question is the expressive power of the system. If a language fails to justify its logical status or if its expressive power is too limited, logicians' interest in that language will fade. Accordingly, facing a strong prejudice against diagrams, which is more deeply rooted in the modern history of logic than in any other area, some logicians put their priority in examining whether there is any intrinsic reason why symbolic systems, but not diagrams, could provide us with a rigorous proof. I took up this question with Venn diagrams and showed that this system is not only sound and complete, but a slightly modified version of the Venn system is logically equivalent to a monadic language.[29] Hammer and Shin (1998) modified Euler diagrams to show that their modified version can be used in rigorous proof as well. Earlier works on Peirce's Existential Graphs[30] (in which graphical reasoning itself was not the main topic) can be re-evaluated to find significant contributions to the understanding of heterogeneous reasoning from logicians' point of view. Based on existing case studies, Barwise and Etchemendy, the first two logicians who launched the inquiry into diagrammatic proofs in logic, conclude that "there is no principled distinction between inference formalisms that use text and those that use diagrams. One can have rigorous, logically sound (and complete) formal systems based on diagrams."[31] This conviction was necessary for the birth of their innovative computer program Hyperproof, which adopts both first-order languages and diagrams (in a multi-modal system) to teach elementary logic courses.[32]

Each of these various research agendas is important to understanding heterogenous reasoning and needs to be taken seriously. With many concrete results in diagrammatic reasoning in hand, now is a good time to bridge the gap among the different strands of the research, which is one of the main goals of this work. First of all, EG is a sound and complete diagrammatic system, which makes logicians' main worry about its logical status disappear. And EG is logically equivalent to a first-order language, which meets the criticism among logicians and mathematicians that a diagrammatic system cannot express as much as a symbolic system does. Also, unlike with domain-specific diagrams or pictures,[33] this system is comprehensive and general enough to have the wider application required by AI researchers and design theorists.[34] Moreover, this book will show how to make the existing system more efficacious and more natural by a new way of understanding graphs on their own terms, which is directly related to AI researchers' and design theorists' agendas. At the same time, abundant comparisons between EG and a logically equivalent symbolic language[35] provide useful material to satisfy cognitive scientists' main interests—cognitive differences between symbolic and diagrammatic systems.[36]

The study of Peirce's EG could not only bridge a gap among multiple agendas in contemporary research on multi-modal representation, but also provide us with a chance to explore a rather forgotten historical fact that a diagrammatic representation system was invented at almost the same time as a modern symbolic system was. Moreover, the first comprehensive non-symbolic system, EG, was devised by Peirce, who is one of the founders of modern symbolic logic. This interesting historical fact encourages us to inquire into the philosophical motivation behind Peirce's invention of a graphical system.

Chapter 2 will reconstruct Peirce's view of different logical notations and different representations systems, which I claim to be the first comprehensive theory of heterogeneous reasoning. The evaluation of Peirce's work as a theory of heterogenous reasoning is bound to be new for at least two reasons. First of all, while many important works are available about Peirce's theory of signs and logical notation,[37] the topic of heterogeneous reasoning has not been explored in the context of logical theory, of formalization, or of the philosophical motivation behind EG. More importantly, the area in which I claim Peirce's contribution should be recognized, i.e., heterogeneous reasoning, is a new interdisciplinary research area, as explained above. I will show in chapter 3 that Peirce's view—the advocacy of a system with more than one kind of sign—is implemented in his own graphical system. That is, chapter 3 presents the thesis that EG is a heterogeneous representation system, by examining both symbolic and iconic elements of the system.

Chapters 4 and 5 turn to more specific aspects of EG. Granting the legitimacy of logicians' criticisms against EG, in these two chapters I present new methods of understanding the Alpha and Beta systems, respectively. We can meet existing complaints against EG by discovering more visual features present in graphs and by identifying the unique nature of graphs. The fundamental differences between symbolic and diagrammatic systems are discussed in detail in these two chapters.

Chapter 6 raises a question as to the relation between Peirce's theory and his own practice: How did Peirce himself, who laid out the philosophical groundwork for heterogeneous systems as shown in chapters 2 and 3, fail to obtain better ways of understanding his own graphical system?[38] For this curious historical question, I explore assumptions behind Peirce's distinction between logical systems and calculi and his intention to make EG a logical system, not a calculus. Chapter 7 concludes with a short summary of the book and suggestions for future research on heterogeneous reasoning.

2

The Birth of Graphical Systems

The history of modern logic has been one of intense investigation of symbolic systems. Interestingly, however, Charles Sanders Peirce, a founder of modern logic, developed two equivalent logical systems, one symbolic but the other graphical. The question I raise and aim to answer in this chapter is "Why did one of the founders of modern symbolic logic invent an elaborate diagrammatic representation system?" As is well known, the notations Frege adopted for his formalized language are more iconic than Peirce's corresponding symbolic language.[1] However, it was Peirce who presented a graphical system equivalent to a quantified language. I will argue that Peirce's invention of a different kind of representation system is not just an accidental product of a logician's mind, but a clear reflection of his philosophy of logic, which differed from that of contemporary logicians. In this chapter, I explore the philosophical roots of the birth of Peirce's EG in his theory of representation and logical notation, which is the heart of Peirce's philosophy of logic.

The first section introduces two insightful explanations related to our inquiry and concludes that more detailed explanations are needed to track down the philosophical motivation behind EG. The second section examines Peirce's well-known phrase 'diagrammatic reasoning'. This leads us to the investigation of Peirce's theory of signs in the third section. After extending my analysis to some of the controversial contemporary topics in research on multi-modal reasoning, I claim in the last section that Peirce is the first person who presented a comprehensive theoretical groundwork for a heterogeneous formal system.

2.1 Preliminaries

Hintikka's and Dipert's evaluations of Peirce's contribution to the history of modern logic allowed us not only to appreciate Peirce's unique position in the history of logic but also to gain a more coherent understanding of Peirce's own philosophy of logic. The latter achievement, as I will show below, advances us toward the goal of the current chapter, that is, to explore Peirce's philosophical motivation behind his graphical system. As I point out at the end of each subsection, while both Hintikka's and Dipert's interpretations are consistent with Peirce's invention of a graphical representation system, neither of them provides us with a philosophically sufficient account of Peirce's invention of EG.

2.1.1 Graphs and the model-theoretic tradition

As the title suggests, van Heijenoort's article "Logic as calculus and logic as language" (1967) draws a distinction between Boole's *calculus ratiocinator* and Frege's *lingua characterica*[2] and makes an interesting connection between the *universality* of Frege's *lingua characterica* and Frege's lack of metatheorectical concepts.[3] According to van Heijenoort, while Boole's universe for a logical calculus does not have any ontological significance, and hence easily allows changes, "[f]or Frege it cannot be a question of changing universes,"[4] since Frege's logic presupposes that we have only one universe, i.e., *the* universe.[5] The domain of the language includes everything, and there is no way for us to step back from this language to express ourselves outside of it. "[N]othing can be, or has to be, said outside of the system."[6] Therefore, van Heijenoort argues that there is no room for metalogic in Frege's concept of logic.

When Goldfarb examines the nature of quantifiers in modern logic, he makes the following observation about logicists' interpretation of quantifiers, which is consistent with van Heijenoort's emphasis on Frege's '*the universe*':

> The range of the quantifiers—as we would say—are fixed in advance once and for all. The universe of discourse is always the universe, appropriately striated. . . . For Frege and Russell . . . [e]very logical formula has a fixed meaning; there is no question of reinterpreting any sign. . . . Similarly, for Frege and Russell (or Russell at least until he became influenced by the doctrines of Wittgenstein), logic is about something, namely, everything.[7]

The inflexibility of the interpretation of quantifiers prevents us from taking logic as subject-neutral or from applying logic to any particular domain in which we are interested. This, as Goldfarb says, is very distant from *our* view of logic. Thus, he arrives at a very similar evaluation of Frege's and Russell's project as does van Heijenoort—that a metalogical framework is not possible in Frege's and Russell's logicism:[8]

> To arrive at metamathematics from Russell's approach we must add the "meta," that is, the possibility of examining logical systems from an external standpoint.[9]

At the same time, Goldfarb draws our attention to the other tradition in modern logic, which allows for the concept of 'meta':

> During roughly the same period [as Frege and Russell were working] a completely different approach to logic was being explored by Schröder and his followers. This tradition of the algebra of logic dates back to Boole's work on the calculus of classes, a calculus also interpretable as a calculus of propositions. *Building on earlier work of Peirce*, . . . Schröder develops the calculus of relatives (that is, relations).[10]

The different approach Goldfarb describes here coincides with the tradition of a *calculus ratiocinator* that van Heijenoort discussed.[11] The view of logic as a calculus permits the language to be *re*interpreted so that we can apply it to particular domains or we can talk about the language from outside of it, in a metalanguage. This is why the concept of 'meta' can play a crucial role in the tradition of a *calculus ratiocinator*.

Based on a contrast between these two traditions, Hintikka draws a sharper distinction among modern logicians. Accepting van Heijenoort's terminology, Hintikka calls Frege's *lingua characterica* view the tradition of the universality of logic, but with an important modification. The universality of logic implies not only one universe, but only one language. Therefore, we are imprisoned both in this one world and in this one language. By contrast, Boole's *calculus ratiocinator* view allows multiple domains and multiple interpretations, which is why Hintikka calls this position the model-theoretic tradition.[12] Frege, Russell, Whitehead, Wittgenstein, and Quine are the main figures in the former, unitary language, tradition, and Boole, Schröder, Löwenheim, Gödel, the later Carnap, and Tarski the principals in the latter model-theoretic one.

When Hintikka makes important contrasts between Frege versus Peirce and places Peirce in the model-theoretic tradition, Peirce's own passages about EG are quoted as evidence for Hintikka's claim—Peirce's acceptance of Boole's *calculus ratiocinator*:

> Peirce himself identifies with perfect clarity the "very serious purpose" of his language of graphs, by saying that "this system is not intended to serve as a universal language for mathematicians or other reasoners like that of Peano" (4.424).... [T]his statement shows that Peirce was dealing with interpreted logic.[13]

Without discussing Peirce's EG in any detail, Hintikka draws our attention to Peirce's adoption of graphs or icons in a language of logic to support his main thesis that Peirce belongs to the model-theoretic tradition: "Hence Peirce's willingness to theorize about icons and to use them in his actual work in logic is but another facet of his model-theoretical approach to language and logic."[14] The tradition to which Frege belongs does not allow us even to consider any other kind of language except *the* language we now have.

Hence, the following conclusion easily follows from Hintikka's position: If Peirce had held the view of the universality of logic, he could not have invented a graphical system. Taking a logical language as a re-interpretable calculus was a necessary condition for Peirce to come up with a graphical system. This also explains why Frege, one of the two founders of modern logic, did not develop a graphical system, in spite of his iconic-looking notation for quantifiers. Frege's view on universality left no room to consider any other logical system than the symbolic one that he had, and in particular no room for a non-symbolic language.

Thus, Hintikka's evaluation of Peirce's location in the history of logic is consistent with the birth of EG from Peirce's philosophy of logic. However, it does not provide us with sufficient explanation of the birth of EG, since the model-theoretic view of logic does not necessitate a logical system other than symbolic ones. For example, Peirce could have invented more than one symbolic system to show that there is no one single universal language for logicians. Therefore, the differences Hintikka highlights between the two different logical theories do not complete the puzzle we are interested in solving in this chapter—"Why a *graphical* system (rather than more than one symbolic system)?" and "Why *this* graphical system?"

2.1.2 Graphs as a formal system

When Dipert examines Peirce's place in the history of logic, he makes an interesting observation that the meaning of 'formal' has shifted in a wrong direction from the nineteenth to the twentieth century. While the formal logic of the nineteenth century is not limited only to symbolic logic, the more recent development of logic concentrates on symbolic aspects of logic only:

> We should also give some hard thought to the difficult question of how much conceptual progress is made by symbolization and symbolic rigor alone. Although Frege and Peirce are largely beyond suspicion, it is rather clear that the recent history of logic has appeared to value any, and sometimes quite shallow and unenlightening, symbolisms and axiomatizations and tended to dismiss any non-symbolic, historical account (for example, those of Aristotle or Ockham) as so much empty verbiage.[15]

While he correctly criticizes the predominance of symbolization in the development of twentieth-century formal logic, Dipert does not give any substantial account of what non-symbolization is in formal logic. From the following passage, it is rather clear that for Dipert two concepts—the formal and the symbolic—are distinct from each other: "At root is a twentieth-century misconception of what it is for a logic to be 'formal'. Namely, it need not be symbolic; nor is symbolization a guarantee of a helpful formal analysis."[16]

Then, what is non-symbolic formalism? In the previous quotation, his phrase 'any non-symbolic, historical account' seems to suggest that a historical account is one of the items that modern logic sacrifices for the sake of symbolization. It is true that a historical account, such as Aristotle's or Ockham's, is not symbolic, but it is not formal either. So this cannot be an example for non-symbolic formalism, but is rather non-symbolic non-formalism. Even when Dipert gives credit to nineteenth-century logicians for their correct use of 'formal' logic in the following quotation, it is not clear what it means to be 'formal':

> The nineteenth-century logicians, beginning with De Morgan's *Formal Logic*, kept in mind better than we what it is to be usefully 'formal' (that is, in attending to logical *form*) rather than merely symbolic at all costs.[17]

The phrase 'usefully formal' is ambiguous, since it could mean *either* formal but not exclusively symbolic *or* not completely formal but with a certain appropriate degree of formalism.[18]

Dipert correctly emphasizes that in the recent history of logic, symbolization has been almost the only trademark for formal logic. More importantly, Dipert aimed to convince the reader that nineteenth-century logic was not tied to symbolization as much as twentieth-century logic is. However, when Dipert relates these important historical data to Peirce's contribution to modern logic, all he says is that Peirce, like Frege, made a substantial contribution to modern logic through rigorous and useful symbolization, unlike some modern logicians who adopt symbolism for the sake of symbolism. Here Peirce's merit seems to lie in using symbolism not in a shallow but in an enlightening manner, so that it becomes usefully formal.

While Peirce's appropriate use of symbolism cannot fully explain the invention of a non-symbolic system, Dipert's vague idea that symbolism is different from formalism, I argue, is directly related to a better understanding of the root of EG in Peirce's philosophy of logic. To formalize our reasoning is consistent with, but is not limited to, symbolization. Hence, non-symbolic formalization should be possible. Peirce's invention of EG, a non-symbolic formal system, is the best piece of evidence to show that Peirce did not identify formalization and symbolization. Without a distinction between formalization and symbolization, Peirce could not have come up with two different kinds of logical systems, one symbolic and the other graphical. Dipert praised Peirce for being "the first great pioneer of the substantive (symbolic) *theory* of the logic of relations."[19] He should also have praised Peirce for not confusing the formal and the symbolic, unlike many modern logicians, and as evidence we can point to EG.

The recognition of the difference between symbolization and formalization was a necessary condition for the birth of EG, but not, however, a sufficient one. We need to know more about how Peirce differentiates symbolization and formalization and Peirce's view of what it is for a language of logic to be formal.[20] Therefore, understanding Peirce's concept of a formal language (whether symbolic or not) is crucial to revealing the philosophical foundation for the birth of EG. Peirce, rather than any other previous logician, can give us a better understanding of what it is for a logical notation to be formal, since he himself presented two different kinds of formal systems, one symbolic and the other non-symbolic.

2.2 Diagrammatic reasoning

Peirce's writings about deductive reasoning place great emphasis on diagrammatic reasoning. This, I believe, is one of the main reasons why Peirce's EG has been taken for granted without his philosophical motivation behind the system being explored. Since Peirce takes diagrammatic reasoning seriously, it is no wonder that he himself invented a diagrammatic representation system, EG, or so many have concluded. However, this naive attitude cannot be sustained when we look into the matter more carefully. Peirce called his system not 'diagrammatic' but 'graphical', and many passages from Peirce indicate that he did not use the two concepts—'graphical' and 'diagrammatic'—in a synonymous way.

One of Peirce's often-quoted passages has attracted much attention but has puzzled many philosophers: "All necessary reasoning is diagrammatic; and the assurance furnished by all other reasoning must be based upon necessary reasoning. In this sense, all reasoning depends directly or indirectly upon diagrams."[21] The puzzle is not solved, even when Peirce gives us a clear explanation of what he means by 'diagrammatic reasoning':

> By diagrammatic reasoning, I mean reasoning which constructs a diagram according to a precept expressed in general terms, performs experiments upon this diagram, notes their results, assures itself that similar experiments performed upon any diagram constructed according to the same precept would have the same results, and expresses this in general terms.[22]

This is an important passage, and we will come back to it for further analysis. For now, I would like to find out why this explanation of diagrammatic reasoning has not fully satisfied those who try to understand the sentence "All necessary reasoning is diagrammatic."

It is not difficult to find a similarity between the model of the reasoning process described in the above quotation and the inference steps adopted in Euclidean geometry proofs.[23] At the same time, nobody wants to conclude that Peirce was mainly concerned with the necessary reasoning only as displayed in Euclidean geometry. This conclusion is unacceptable not only because areas of mathematics or logic other than Euclidean geometry should be covered under the topic of necessary reasoning but also because Euclidean-geometry proofs with figures lost their status as rigorous for several reasons. Hilbert's axiomatization showed us that Euclidean figures are dispensable in proofs. More importantly, as is well-known, a user is easily misled by certain accidental features of a figure. Peirce himself presented Euclid's fallacy as an illustration of incorrect reasoning, and made the following diagnosis of the fallacy: "[The (false) axiom that the whole is greater than its part] tempts him to draw a figure and judge by the looks of it what is part and what whole."[24] Since Peirce was fully aware of the problem of relying on figures, most probably Peirce's explanation above of diagrammatic reasoning must not have directly referred to Euclidean-geometry proofs with figures.

This discussion reveals that our puzzlement about Peirce's 'diagrammatic reasoning' originates from the unclarified meaning of Peirce's term 'diagram'. One consensus reached among Peircean scholars is that Peirce's meaning of 'diagrams' is much broader than our naive use of 'diagram'.[25] But substantially informative answers to the question "What does Peirce mean by 'diagram'?" have not been forthcoming in the history and philosophy of logic in spite of their importance.

Peirce talks about diagrams quite often in the context of necessary reasoning,[26] but without any attempt to define what a diagram is. The following sentence has been presented by Peircean scholars as evidence for a broader meaning for Peirce's term 'diagrams', mainly because an auditory diagram is included:[27]

> Such a diagram has got to be either auditory or visual, the parts being separated in the one case in time, in the other in space.[28]

It is true that a diagram in our naive sense cannot be auditory. However, a more interesting point we should not overlook in this sentence is that Peirce's visual diagrams too cannot be the same as diagrams in our sense. According to Peirce, as long as the basic vocabulary of a representation is separated in space, it is a visual diagram. Thus, a symbolic language whose vocabulary is spatially separated is also a visual diagram in Peirce's sense. Again, the sentence quoted above does not say much about what a diagram is, except that a diagram can be heard or seen.

However, we can infer Peirce's ideas about what could or should be a diagram from the following sentences scattered throughout the same paragraphs as the previously quoted sentence:

> . . . the parts of each series [of a diagram] being evidently related as those provinces[29] are said to be, ... [I]t will be necessary to use signs or symbols repeated in different places and in different juxtapositions, these signs being subject to certain "rules," that is, certain general relations associated with them by the mind.[30]

What is suggested here can be rearranged into three important aspects of diagrams. First, signs or symbols are used in a diagram. Second, signs or symbols are related to the reality that the diagram is about. That is, signs or symbols in a diagram represent something which we aim to reason about. Third, we have rules for relations among the signs or symbols, which give us permission how to manipulate some relations among signs or symbols to obtain some other relations among them.

Peirce himself did not use the phrases 'representation', 'system', or 'inference rules', but what is stated here is very similar to our concept of a deductive representation system. The three points in the above quotation address the syntactic and semantic conditions of a representation system.[31] Signs or symbols are the basic vocabulary of a system, that is, a syntactic aspect. A syntactic unit may represent facts other than itself, according to a semantic condition. The third point, about inference rules, is quite interesting, since the rules tell us how to manipulate syntactic objects, but on the other hand the ultimate justification for transformations can be found at a semantic level.

It is obvious that Peirce's 'diagram' is not restricted to a figure (such as those in Euclidean proofs) but is rather closer to a unit of a system equipped with representational import and its own transformation rules. Hence, Peirce's 'diagrammatic reasoning' refers to the reasoning carried out through a representation system. One of his well-known papers, "Prolegomena to an apology for pragmaticism," starts with a paragraph which is consistent with and quite close to my reconstruction of 'diagram' in the Peircean sense: "[L]et us construct a diagram to illustrate the general course of thought; I mean a System of diagrammatization by means of which any course of thought can be represented with exactitude."[32] Therefore, according to Peirce, a diagrammatic system does not have to consist of circles, lines, or dots, as we often assume, but means a representation system in our sense. If so, we should not be surprised to encounter Peirce's statement "[A]lgebra is but a sort of diagram"[33] or the phrase "the diagrams of algebra."[34] Algebra, which has been considered a typical symbolic system, is also a visual diagram, according to Peirce, since algebraic symbols could represent certain facts and are manipulated according to their own rules.

Interestingly enough, when Peirce presents his theory of signs, his discussion becomes even closer to our concept of representation system. After stating "A sign, or a representamen, is something which stands to somebody for something in some respect of capacity,"[35] Peirce identifies three important aspects of a sign: (i) a sign itself, (ii) the object the sign represents, and (iii) a basis for the representation.[36] For each aspect, one branch of semiotics arises, as Peirce says:

> The first [the first branch of semiotics] ... [w]e may term ... *pure grammar*. It has for its task to ascertain what must be true of the representamen used by every scientific intelligence in order that they may embody any *meaning*.
>
> The second is logic proper.... [L]ogic proper is the formal science of the conditions of the truth of representations.
>
> The third, ... I call *pure rhetoric*. Its task is to ascertain the laws by which in every scientific intelligence one sign gives birth to another, and especially one thought brings forth another.[37]

These three branches are the basic components of a logical system in our sense. Pure grammar is the syntax of a language, logic proper is its semantics, and pure rhetoric consists of the inference rules of a system.

2.3 The theory of signs

Diagrams consist of signs, Peirce says. Therefore, if diagrams are representation systems, as I have outlined above, signs used in diagrams must represent something else. How does a sign of a (diagrammatic) system represent anything else? How is any representation possible?[38] This is what Peirce asks in the question "[H]ow is any diagram ever to perform that identification [that some points in a map represent some points in nature]?"[39]

A quick answer to this question is that how a sign represents something depends on what kind of a sign it is. Now, we know that the story of representation Peirce will give us is bound to be quite complicated, since he examines what a sign is at a much deeper level than any one had done before.[40] "We may distinguish between different kinds of signs according to the relation between their material [representing facts] and their imputed qualities [represented facts]."[41] After examining the three different kinds of signs in the first subsection, I relate Peirce's insights to some of the important contemporary topics in the research on heterogeneous representation systems in the second subsection.

2.3.1 Symbols, indices, and icons

Peirce classifies signs into three groups: *icon*, *index*, and *symbol*.[42] This classification is based on different relations between a sign and what it represents.[43] Roughly speaking, an icon represents something based on the *resemblance* between a sign and its object, an index by being *directly* related to its object, and a symbol by *convention*.[44] As Peirce admits, sometimes one sign might belong to more than one category, or the distinction among them is not clear in some cases.[45] I will emphasize the fundamental differences among signs rather than the ambiguous areas.

Sign *A* is a *symbol* for *B* if *A* is associated with *B* by convention. Most English words function as symbols. They represent some objects or some states of affairs by convention. For example, we understand the meaning of the word 'book' not because the string of these four characters looks like a book but because we have come to learn the meaning of the word 'book'. However, knowing the meaning of this word does not help us to pick out a particular book. That is, a word acting as a symbol is associated with a general meaning of the word, not a particular object or a particular state of affairs. We can also find conventional signs which are not words but which are used as symbols. A dove is a symbol for peace, a white flag for surrendering, etc.

A sign that is an *index* is associated with what it represents as if it were pointing out the object. No descriptive meaning needs to be attached. In English, indexicals or pronouns are typical examples of index, according to Peirce.[46] A proper name could be treated as an index.[47] The main function of an index is to refer to a specific object or state of affairs, not to describe it.[48] Sometimes this indexical function is accomplished not by pointing something out but by a law of nature we are familiar with. For example, smoke indicates that there is fire. That is, smoke is an index of fire. There is no conventional descriptive meaning (attached to smoke) to lead us to this representation relation, but the laws of nature almost force us to reach the relation between smoke and fire. Therefore, we do not say that smoke is a symbol of fire. Peirce uses an example of a low barometer with moist air as an index of rain: "We suppose that the forces of nature establish a probable connection between the low barometer with moist air and coming rain."[49] Again, a low barometer cannot be said to be a symbol for rain. There is not much room for our artificial convention in this kind of a representational relation.

A sign that is an *icon* represents a certain object or a certain state of affairs by its likeness to its object or state of affairs.[50] Nobody would confuse a painting of someone, say Tom, with Tom as a person. However, for those who know what Tom looks like, no convention is needed to see this painting as a painting of Tom. Also, no special context needs to be given to pick out the person, Tom, of whom this is a painting, nor is a force of nature necessary in order to understand the connection between the painting and Tom.[51] Hence, a painting of Tom is neither a symbol of nor an index for Tom. The resemblance of the painting to Tom is sufficient to know of whom this is a painting. Some Chinese characters must have started as icons. "In all primitive writings, such as the Egyptian hieroglyphics, there are icons of a non-logical kind, the ideographs."[52] A sign for elevators or escalators in a building is an example of an icon. Figures (e.g., lines, triangles, and circles) used in geometry proofs are also good examples of icons.

The way some signs are drawn immediately prompts us to recognize what they represent: "[The colors of a landscape and the colors of nature] do not make a match but they are sufficiently like them to suggest immediately to the mind the appearance intended to be represented."[53] Neither a general convention nor a specific indication is involved in this recognition process. The key concept is resemblance. As I noted, in the case of a symbol, there is no resemblance between a sign and what it denotes. The word 'rain' is not similar in any sense to rain. The dove, a bird, is not similar to peace, which is abstract.[54] Indices do not resemble what they denote, either. The word 'I' refers to a speaker, but nothing is similar between the word and the person. There is no resemblance between a low barometer and coming rain. Clearly, resemblance or likeness seems to be a unique way for an icon to represent its object.

In his classic work on a theory of symbols, *Languages of Art* (1976), Nelson Goodman argues that resemblance cannot be a necessary or a sufficient condition of representation.[55] Resemblance is reflexive, but nobody is said to represent himself or herself. Resemblance is symmetric, but we do not say Tom represents his picture, while Tom's picture is said to represent Tom. Goodman also points out that a twin brother resembles his twin but does not represent him. I agree that (iconic) representation cannot be defined solely in terms of resemblance. Moreover, in saying that an icon resembles what it represents, we do not mean to give a definition of iconic representation. As Dipert says, "[Goodman's claim that resemblance is neither a necessary nor a sufficient condition for representation] would be a useful contribution to the discussion if anyone had ever seriously proposed that the signification relationship is exactly and only the resemblance relationship. So far as I am aware, no one ever has."[56] On the contrary, when we try to find out how the representing and represented facts are related to each other, it is assumed that we know which is a representing fact and which a represented fact. In the case of an icon, unlike with a symbol or an index, certain aspects of the icon—which constitute a representing fact—resemble the represented fact. The resemblance between representing and represented facts makes sense only when there is no confusion between representing and represented facts.[57]

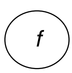
Now, a question that arises is "In what sense or in what aspect does a sign as an icon resemble its denotation?" In the case of photographs, "we know that they are in certain respects exactly like the objects they represent."[58] Physical looks are similar in this case. However, Peirce correctly warns us that resemblance in physical appearance is not the only kind of resemblance between an icon and its denotation: "Many diagrams resemble their objects not at all in looks; it is only in respect to the relations of their parts that their likeness consists."[59] Peirce's example of algebraic equations illustrates this point very well:

> In fact, every algebraical equation is an icon, in so far as it *exhibits*, by means of the algebraic signs (which are not themselves icons), the relation of the quantities concerned.[60]

If we limited resemblance to physical aspects only, the expressive power of icons would be too restricted to be useful as a representation system. In many cases, like the representation of equality, there is no physical look an icon can depict, since the denotation of the icon is abstract. Euler diagrams provide us with a good example to illustrate how sets and relations among sets (all of which are abstract) are iconically represented.

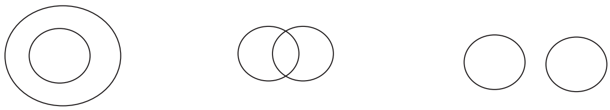
Euler adopts a circle to represent a set. Is Euler's circle a symbol or an icon?[61] This is a classic question that shows that the distinction between a symbol and an icon is sometimes not clear-cut. Importantly, being graphical does not automatically make a sign an icon. Peirce's explanation of icons does not require that being graphical is either a necessary or a sufficient condition for iconicity.[62] Suppose that we represent the set of animals by two different signs, one the letter *A* and the other a circle. Both are conventional in the sense that we do not have any idea about the denotation of each sign unless we learn the stipulation for each sign. However, when we try to add more information to these two signs, interesting differences between them emerge.

For example, we would like to represent the fact that Fido is an animal. Suppose that the symbol *f* denotes Fido. When we use the letter *A* for the set of animals, most probably we need another arbitrary symbol to represent the relation between Fido and the set of animals. For example, the symbol '∈' denotes membership. Hence, '$f \in A$' means that Fido is an animal. On the other hand, in the case of a circle, the most natural way of representing this fact is to write *f* *inside* the circle, rather than adopting another symbol or an icon:



The visually observable relation—*f being inside* the circle—resembles the membership relation between an object and a set: If object *f* is a member of set *A*, then we say *f* is *in* set *A*.[63] Because this homomorphic relation is a quite intuitive one, we do not need any extra convention for it, unlike with the symbol '∈'.[64] Even though it is not clear whether a circle itself is a symbolic or an iconic representation of a set, it is clear that the relation 'being a member of' is iconically (not symbolically) represented when we write '*f*' *inside* a circle.

Similarly, we may iconically represent relations among sets as well. The following diagrams represent subset, intersection, and disjoint relations between two sets, respectively:



Again, the relations between sets are abstract, but these abstract relations are represented not symbolically but iconically: These relations are represented in terms of intuitively homomorphic visual facts, i.e., inclusion, overlapping, and disjointness.[65] It is in the representation of relations that resemblance plays a crucial role in representation by icons, but not in the case of symbols.[66]

For many reasons, we cannot perfectly depict the situation we reason about, and we do not need to, either. In the examples above of Euler diagrams, relations among different classes are what a user intends to represent. Therefore, only those aspects of the reality are depicted by easily observable visual relations among circles. Sometimes we might be interested in representing the locations of objects. In that case, the locations of the signs (which represent the objects) are the only properties whose resemblance matters. Therefore, the icon resembles different aspects of reality, depending on what a user intends to denote by using an icon.[67]

2.3.2 Symbolic versus iconic signs

By extending Peirce's distinction among signs, we can address the following topics which have been heatedly debated in the research on heterogeneous reasoning: What are the strengths and weaknesses of symbolic and iconic representation systems? Is there a theoretical justification for the long-standing prejudice against the use of diagrams in a rigorous proof in logic and mathematics?

Let's examine the case of figures used in proofs in Euclidean geometry. Some might object to the idea that geometric figures are signs in Peirce's sense at all. They might think that we draw a line or a circle in a proof, but none of these figures represents any other thing. In the following, I will show that geometric figures in Euclidean proofs are signs which represent something else and that these signs are icons rather than symbols or indices. The discussion below will also further clarify the concept of resemblance or likeness.

Suppose that a user draws a triangle to prove some property of a triangle. What he draws on a sheet of paper is a *particular* triangle, while the property he is supposed to prove is about triangles *in general*. How does the transition from particularity to generality take place? Peirce's following passage seems to assume that this transition takes place, without explaining how it happens:

> By diagrammatic reasoning, I mean reasoning which constructs a diagram according to a precept expressed in general terms, performs experiments upon this diagram, notes their results, assures itself that similar experiments performed upon any diagram constructed according to the same precept would have the same results, and expresses this in general terms.[68]

A transition *from* the talks about a particular triangle constructed by the user *to* the talks about triangles in general is possible because the former *represents* the latter. Therefore, this particular geometric figure is used as a sign which represents something else.[69] If we assume that an icon represents only a particular thing,[70] we can never cross the bridge *from* the properties that we find in the particular triangle *to* the properties of a triangle in general that we are interested in knowing. If we never allow an icon to represent something in general, then no logical status can be given to icons in a proof.

Let's say that a figure of a triangle represents a triangle in general by its likeness to what it denotes. How can a particular triangle resemble a triangle in general? It seems obvious that this particular triangle resembles a triangle in general much more than the symbol *P* does when we say "Let *P* be a triangle." Certain properties, for example, having three sides and three angles, are shared by both this particular triangle and a triangle in general. However, it should be noted that a specific figure cannot resemble a general triangle in every respect. For example, any particular triangle that we observe has its own size or its own shape, but a general triangle does not. The size of an angle of any specific triangle cannot represent the size of an angle of a triangle in general, since there is no size of an angle of a universal triangle. Therefore, when the user intends to represent a triangle in general by drawing a particular triangle in a geometric proof, his intention is successfully achieved by the recognition that certain properties (for example, having three sides, having three angles, etc.) are representing facts and certain properties (for example, one side being shorter than the other two, no angle being bigger than a right angle, etc.) are not.

Now, we may reach the following general conclusion about the representation of icons: When we adopt an icon to represent something in general (as a symbol does), it is important to classify the observable properties of the particular icon into two different categories: properties that are representing facts and properties that are not. Resemblance or likeness plays a crucial role for the properties that are representing facts. For representing properties, there is an obvious homomorphism between representing facts and represented facts, thanks to the resemblance between them. However, there are properties the user does not intend to use as representing facts. These are accidental properties that the adopted icon has but that the denotation of this icon does not. The existence of accidental properties is directly related to the traditional complaint against the use of diagrams in a proof.[71] While the symbol *P* does not resemble any triangle, a specific triangle figure does resemble a particular triangle in almost every respect.[72] For example, if I wanted to prove some property about a triangle in general but the triangle I drew on the paper happened to be isosceles, then I might easily commit a fallacy in my proof by using some accidental properties of this particular triangle. Therefore, it has been said that a diagram misleads us into fallacies, since it emphasizes wrong aspects of the reality we are reasoning about. We can now specify what these "wrong" aspects are. They are the properties which are not representing facts, and from which, therefore, one should not read off represented facts.

Symbols also have accidental properties. For example, when we represent a triangle by one letter *P*, a token of the letter *P* has many accidental properties, e.g., its own size, its own shape, etc. Then, why isn't a similar problem, i.e., the risk of being misled by wrong aspects, raised against symbolic representation? Accidental properties of a symbol are not confused as representing facts at all, since none of these properties, e.g., the size or the shape of the letter '*P*', is shared by the triangle that this '*P*' represents. Therefore, a symbolic proof system is easily set up so that non-representing facts about symbols do not play any crucial role in any proof step. After the stipulation, say, "Let *P* be a triangle," no step that follows from the stipulation uses the properties of the size or the shape of the token '*P*'.

Symbolic representation adopts arbitrary conventions. Their arbitrariness has two sides. On the one hand, when the conventions are stipulated, we have to learn every convention. What follows from the given premises is not obvious, and we are required to learn how to use permissible manipulations. On the other hand, arbitrary conventions prevent us from exercising our own intuitive way of thinking about their denotations or manipulating these symbols, since these stipulations override our intuitive ideas. Moreover, stipulation, part of the nature of symbols, makes formalization conceptually much easier than for other kinds of signs. As the meanings of symbols are conventionally stipulated, so are the rules of semantics and inference. The concept of formalization thus comes very naturally in the case of symbolic languages, which is why symbolic formal systems have been accepted without any question.

In the case of iconic representation, the relation between representing and represented facts is based on a similarity or resemblance that we may observe at an intuitive level. This is a clear advantage an icon has over a symbol. It can increase the efficiency of a representation system.[73] However, as discussed earlier, since an icon is specific and resembles its denotation in some, but not all, aspects, some properties might be mistakenly thought to be representing facts. This is a drawback of iconic languages.

Both symbols and icons carry out the same function, representation, but there seems to be a trade-off between these two different kinds of signs. The trade-off explains why sometimes symbols are preferred and sometimes icons are. Many logicians, mathematicians, and scientists actually do use icons, but only informally. When it comes down to rigorous formal proofs, the majority of logicians and mathematicians have strongly preferred symbols over icons, since the possibility of fallacy could lead us to disaster in the search of a valid proof. Another important reason for the preference for symbols is that, as explained above, the idea of an iconic formal system has not been considered until very recently. Hence, the concept of formal has been mistakenly identified with the concept of symbolic. In turn, this incorrect identification has intensified the strong prejudice for symbolic systems.

The prejudice against the use of diagrams is thus clarified. No logician or mathematician can afford to take the risk of being misled by icons. But what is the price of this bias? Obviously, we sacrifice the strong points of icons. That is, we lose more intuitive relations, i.e., the visually clear homomorphism, between representing and represented facts. By contrast, in symbolic representation systems, both semantic rules and inference rules need to be arbitrarily stipulated, and therefore, we have to learn them from scratch. The trade-off *seems* to have been made for accuracy at the expense of intuitiveness (or effectiveness).

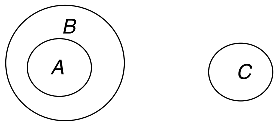
Interestingly, an assumption behind the choice of symbols over icons is that the strength of icons, i.e., the intuitive relation between representing and represented facts, is not compatible with the strength of symbols, i.e., formalization. So when we decide to formalize, we are ready to give up the use of any iconic language. In the next section I challenge this assumption so that we may see that the prejudice against the use of diagrams in proofs was a hasty conclusion after all.

2.4 Heterogeneous formal systems

According to Peirce's theory of signs, none of three kinds of signs—symbol, index, and icon—is intrinsically superior or advantageous to the other two. Each of them carries out the same function, representation, in a different manner. And, as cited in the second section of this chapter, Peirce was perfectly aware of possible wrong inferences caused by figures in geometry proofs.[74] However, a main difference between Peirce and other logicians is that from the possible riskiness of figures Peirce did not hastily conclude that an iconic language should be abandoned in reasoning. On the contrary, Peirce says, "A diagram ought to be as iconic as possible,"[75] and he highlights the advantage of icons over symbols:

> For a distinguishing property of the icon is that by the direct observation of it other truths concerning its object can be discovered than those which suffice to determine its construction.[76]

What does he mean by "direct observation"? An example of Euler diagrams illustrates Peirce's point well. Suppose that we draw the following three circles in order to represent two pieces of information: "All *A* are *B*" and "No *B* is *C*."



Then another proposition, "No *A* is *C*," can be observed in the diagram, since the circle for *A* and the circle for *C* are disjoint from each other. Also, it is *directly* observed, since we do not have to manipulate the original diagram in order to obtain this new piece of information.[77] The benefit of direct observation becomes much clearer when we compare Euler diagrams with symbolic representation. The two sentences, "$A \subset B$" and "$B \cap C = \emptyset$" must be manipulated in order to obtain the sentence "$A \cap C = \emptyset$." Therefore, we obtain the new piece of information that no *A* is *C* more efficiently in iconic representation than in symbolic representation. Even though Peirce did not use the word 'effectiveness', the distinguishing property of an icon he is talking about in the passage quoted above is related to the effectiveness that iconic languages can have over symbolic ones.

At the same time, Peirce makes it clear in other writings that icons have their own shortcomings:

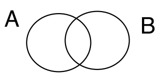
> They [Euler circles] well fulfill the function of icons, but their want of generality and their incompetence to express propositions must have been felt by everybody who has used them. Mr. Venn has, therefore, been led to add shading to them; and this shading is a conventional sign of the nature of a token [symbol].[78]

We need to clarify three points in order to understand this passage better. First, what does Peirce mean by 'want of generality' and what kinds of propositions are not representable in Euler diagrams? Venn's criticism of Euler diagrams directly addresses the same points:

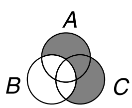
> The weak point in this [an Euler diagram], and in all similar schemes, consists in the fact that they only illustrate in strictness the actual relation of classes to each other, rather than the imperfect knowledge of these relations which we may possess, or may wish to convey by means of the proposition.[79]

That is, Euler diagrams fail to represent partial information about the relationships between the sets represented, and therefore, some propositions, for example, "All *A* are *B*" and "All *C* are *B*," cannot be represented in a single diagram.[80]

Second, Peirce says that the failure of Euler diagrams to represent partial information is why Venn introduced shading. However, it should be noted that shading was introduced to Venn's system after a new concept, that of primary diagrams, was introduced. Therefore, strictly speaking, Venn's primary diagram was the solution to the lack of generality of Euler diagrams. Primary diagrams represent sets without stating any specific relation among them. For example, the following is the primary diagram about sets *A* and *B*:



With these primary diagrams and shading, the two pieces of information, "All *A* are *B*" and "All *C* are *B*," are represented simultaneously as follows:



Third, and most importantly, we need to pay attention to Peirce's insightful observation about Venn's shading: Shading is a symbol, not an icon.[81] That is, shading represents emptiness, and this representation is done by convention, not by resemblance. Hence, Venn diagrams are not purely iconic, but mix icons and symbols. According to contemporary terminology, Venn diagrams are an example of not a homogeneous, but a heterogeneous system. Does the mixture between different kinds of signs cause any problem? Not at all, according to Peirce. On the contrary, after considering the merits and the drawbacks of different kinds of signs, Peirce arrives at the following conclusions:

I have taken pains to make my distinction of icons, indices and tokens clear, in order to enunciate this proposition: in a perfect system of logical notation signs of these several kinds must all be employed.[82]

If symbolic logic be defined as logic—for the present only deductive logic—treated by means of a special system of symbols, either devised for the purpose

> or extended to logical from other uses, it will be convenient not to confine the symbols used to algebraic symbols, but to include some graphical symbols as well.[83]

Peirce, who has explored different kinds of signs, advocates more than one mode of representation in one and the same logical system; he favors a heterogeneous representation system over a homogenous system.

Peirce, a logician known as a founder of modern symbolic logic, was thus also the first person to establish the theoretical grounds for heterogeneous systems. I argue that this theoretical background is the main philosophical driving force behind his graphical system, EG. Peirce's invention of graphical systems was not a coincidental product, but a logical product, of his theory of signs.

My analysis of Peirce's theory of signs as the philosophical explanation for his invention of EG is not only compatible with Hintikka's and Dipert's perspectives on Peirce's logical theory (discussed in the first section of this chapter) but also answers the questions left unanswered by these two perspectives. We can see that Peirce's theory of signs confirms Hintikka's interpretation of Peirce's theory of logic: Peirce, exploring different kinds of signs, must have thought about re-interpreting a language of logic, and would not have insisted on the existence of one unique language. At the same time, Peirce's theory of signs and representation is the philosophical source of EG. Our examination of Peirce's theory of logical notations also supports Dipert's evaluation that Peirce did not wrongly identify formalization and symbolization, as the majority of twentieth-century logicians did. Further, we discovered Peirce's theoretical basis for a distinction between formalization and symbolization. In his theory of notation, symbols do not occupy a special position, but belong to one of the three kinds of signs. Peirce believed all three to be necessary, and thus Peirce supported multi-modal formal representation.

Since Peirce suggested that a logical system should have signs of different kinds, he must have assumed that both symbolic and iconic formalization are possible. This is an interesting and important assumption. Turning to Peirce's concept of formalization brings us back to our discussions of Peirce's diagrammatic reasoning in §2.2. I argued that Peirce's diagrammatic system is close to our deductive representation system: "I mean a System of diagrammatization by means of which any course of thought can be represented with exactitude."[84] "Exactitude" can be achieved by his three branches of semiotics: pure grammar, logic proper, and pure rhetoric. These three branches, respectively, correspond to the syntax of a language, its semantics, and the inference rules of a system, as we understand them. Remarkably, Peirce's concept of diagrammatization corresponds to our concept of formalization. As examined in §2.3, there is no theoretical reason why some kind of signs should have these three branches while some do not. Therefore, iconic diagrammatization, that is, iconic formalization, should be possible.

Some might argue that icons cannot be formalized, since they have, by their nature, misleading aspects, as we have seen in §2.3.2. This is a hasty conclusion. I suggest that we improve iconic languages so that they maintain their strengths but rule out possible sources of fallacy. How do we make sure that a user does not mistake accidental, non-representing facts for representing facts? Recall that the reason why this kind of mistake does not arise in the case of symbolic systems is that every representing fact there is stipulated. For example, the size of a symbol is not stipulated as a representing fact. Hence, no valid conclusion directly follows from the size of a symbol. We may follow a similar procedure for iconic systems as well. We can provide semantic rules so that the user knows which aspects of resemblance are taken to be representing facts of icons. Also, permissible transformations should be stipulated so that no accidental properties of an icon can provide us any substantial piece of information. Nevertheless, we can keep an observable homomorphic relation between representing and represented facts. Then we will have a formal system for iconic language, which we need to avoid the well-rooted prejudice against non-symbolic representations.

I believe that we are very close to what Peirce had in mind when he invented his own graphical system, EG. In the next chapter I show that EG is a standard formal representation system equipped with its own syntax and semantics, and that this system is heterogeneous, as Peirce believed a perfect logical system should be.

3

Existential Graphs as a Heterogeneous System

In the previous chapter, we saw that Peirce's theory of logical notation assigns the same logical status to both symbolic and non-symbolic signs and establishes the theoretical groundwork for heterogeneous formal systems. This, I argue, is the fundamental philosophical motivation behind Peirce's ambitious work, EG, the first comprehensive graphical system.

In what follows, I support this claim by examining EG carefully. After introducing EG in the first section, in the second section I explore how the symbolicity and the iconicity of a logical system are developed throughout the history of Peirce's two graphical systems, Entitative and Existential Graphs. In the final section, I concentrate on the iconicity of EG, focusing especially on Peirce's own discussion.[1]

3.1 Introduction of Existential Graphs

EG consists of three parts, which Peirce calls Alpha, Beta, and Gamma Graphs. These correspond to propositional, predicate, and modal logic, respectively. Throughout this book I focus on the first two, i.e., Alpha and Beta graphs. In this section I will introduce the basic vocabulary of each system and present the meanings of graphs in an intuitive way. More systematic formulations and the mechanics of each system will be the main topics of the next two chapters.

3.1.1 Alpha graphs

The Alpha system has two kinds of primitive vocabulary: sentential symbols and cut. Sentential symbols, $A_1, A_2, \ldots$, represent propositions, and a cut represents negation. Also, when graphs are juxtaposed with each other, we interpret it as conjunction. Examples easily illustrate how to read off Alpha graphs:

Example 3.1 Let S stands for the proposition "It is sunny," and W "It is windy." Graph (1) means "It is sunny," (2) "It is sunny and windy," (3) "It is not the case that it is sunny and windy," (4) "It is neither sunny nor windy," and (5) "It is not the case that it is neither sunny nor windy," which is the same as "It is sunny or windy."

(1) (2) (3) (4) (5)

Let's define well-formed Alpha graphs inductively.

The set of Alpha graphs, $\mathcal{G}_\alpha$, is the smallest set satisfying the following:

1. An empty space is in $\mathcal{G}_\alpha$.
2. A *sentence symbol* is in $\mathcal{G}_\alpha$.
3. *Juxtaposition closure* If G_1 is in $\mathcal{G}_\alpha, \ldots,$ and G_n is in $\mathcal{G}_\alpha$, then the *juxtaposition* of these n graphs, i.e., $G_1, \ldots, G_n$, (we write '$G_1 \ldots G_n$' for juxtaposition) is also in $\mathcal{G}_\alpha$.
4. *Cut closure* If G is in $\mathcal{G}_\alpha$, then *a single cut of* G (we write '$[G]$', following Peirce's linear notation)[2] is also in $\mathcal{G}_\alpha$.

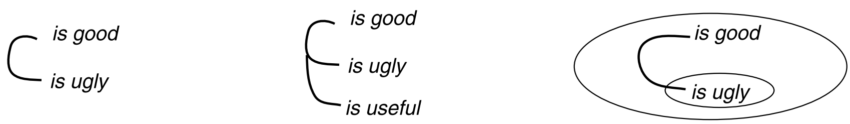
All the graphs in example 1 fit this definition and are, therefore, well-formed.

Intuitively, a cut corresponds to a negation symbol and juxtaposition to a conjunction symbol in a propositional language. Therefore, for now, let's consider this system analogous to a sentential symbolic language with two sentential connectives, conjunction and negation. Hence, the Alpha system, like a symbolic system with negation and conjunction, is truth-functionally complete. However, the analogy between these two systems breaks down in several important respects, and the differences between them will be discussed in detail in the next chapter.

3.1.2 Beta graphs

The Beta system is analogous to a pure symbolic first-order language with an equality symbol, but without a constant symbol. This system has three kinds of primitive vocabulary: predicate symbols, cut, and line. A cut represents the negation of whatever is written inside the cut, just as in the Alpha system. The main difference between Alpha and Beta graphs is the introduction of lines into the Beta system. A line, called 'a line of identity', denotes the existence of an object, and it can have many branches, called 'ligatures'.

Example 3.2 The first graph means "Some good thing is ugly," the second "Some good thing is ugly and useful," and the third "It is not the case that some good thing is not ugly," which is the same as "Every good thing is ugly."[3]



For a better understanding of the Beta system, it is illuminating to compare the extension from sentential to predicate languages on the one hand and the extension from Alpha to Beta graphs on the other hand.

The extension from sentential to predicate languages is quite straightforward when the sentence symbols of a sentential language are replaced by the concatenations of predicates and names. Suppose that an atomic well-formed formula (wff) of a sentential language $\mathscr{L}_s$ is a concatenation of a predicate and a correct number of names depending on the arity of the predicate, for example, $P^n(a_1, \ldots, a_n)$, where P^n is an n-place predicate and $a_1, \ldots,$ and a_n are names in $\mathscr{L}_s$. When this sentential language is expanded to a pure predicate language $\mathscr{L}_p$,[4] we require the following grammatical extensions:

Extension in vocabulary The vocabulary of $\mathscr{L}_p$ is the vocabulary of $\mathscr{L}_s$ *plus* variables and quantifiers

Extension in atomic wffs The definition of atomic wffs is extended in the following way:

$\mathscr{L}_s$: $P^n(a_1, \ldots, a_n)$ is an atomic wff, where P^n is an n-place predicate and $a_1, \ldots,$ and a_n are names.

$\mathscr{L}_p$: $P^n(t_1, \ldots, t_n)$ is an atomic wff, where P^n is an n-place predicate and $t_1, \ldots,$ and t_n are *either* names *or* variables.

Extension in complex wffs The following additional inductive clause is added to the definition of wffs of $\mathscr{L}_s$: If $\mathscr{Q}v_i$ (where $\mathscr{Q}$ is a quantifier and v_i is a variable) is to the left of a wff, then the result is also a wff.

Therefore, every wff of $\mathscr{L}_s$ is a wff of $\mathscr{L}_p$. That is, the language is literally extended.

The extension from the Alpha system to the Beta system cannot be stated as clearly as the above extension of symbolic languages. First of all, EG does not have names. So, unlike the extension from $\mathscr{L}_s$ to $\mathscr{L}_p$, we cannot replace a sentence symbol of the Alpha system with a concatenation of a predicate symbol and names. Another major difference whereby the analogy between symbolic languages and EG breaks down is that there is no unit in EG which corresponds to an open formula of a predicate symbolic language. Every Beta graph corresponds to a sentence of a predicate language. EG does not have a syntactic device corresponding to variables.

Let us compare the basic vocabulary in the Alpha and Beta systems. No sentence symbol is used in the Beta system. Instead, two new syntactic objects are introduced: predicate symbols and lines of identity (henceforth, 'LI'). That is,

Vocabulary in the Alpha system sentence symbols and cuts

Vocabulary in the Beta system predicate symbols, LIs, and cuts

An LI does not have to be straight. All of the following are LIs:

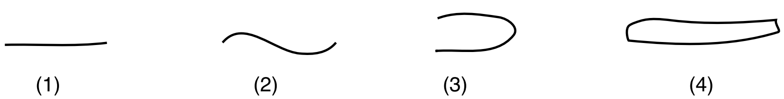
For further discussion, we borrow the term 'loose end' from Zeman[5] and define it as an end of an LI without any attached predicate. None of the lines above has a predicate attached, and therefore, each end of each line is a loose end. That is, each line above has two loose ends.

In the Alpha system, two inductive clauses—one for the juxtaposition of graphs and the other for the enclosure of a graph within a cut—tell us how to build graphs out of atomic graphs. In the Beta system, the juxtaposition clause is kept, the cut clause is modified, and two more clauses are added.

The set of Beta graphs, $\mathcal{G}_\beta$, is the smallest set satisfying the following:

1. An empty space is in $\mathcal{G}_\beta$.
2. A *line of identity* is in $\mathcal{G}_\beta$.
3. *Juxtaposition closure* If G_1 is in $\mathcal{G}_\beta, \ldots,$ and G_n is in $\mathcal{G}_\beta$, then the juxtaposition[6] of these n graphs $G_1, \ldots, G_n$ (we write '$G_1 \ldots G_n$' for juxtaposition) is also in $\mathcal{G}_\beta$.
4. *Predicate closure* If G is in $\mathcal{G}_\beta$, then a graph with an n-ary *predicate* symbol written at the joint of n loose ends in G is also in $\mathcal{G}_\beta$.
5. *Cut closure* If G is in $\mathcal{G}_\beta$, then a graph in which a *single cut* is drawn in any subpart of G without crossing a predicate symbol is also in $\mathcal{G}_\beta$.
6. *Branch closure* If G is in $\mathcal{G}_\beta$, then a graph in which an LI in G *branches* is also in $\mathcal{G}_\beta$.

According to the second clause, each of the following four graphs is a Beta graph:



An LI can be straight or curved. Moreover, the both ends of an LI can meet each other to form a closed curve, like graph (4) above. We call this enclosed curve a *cycle*.[7]

The third clause for $\mathcal{G}_\beta$ is the same as the third clause for $\mathcal{G}_\alpha$ and is straightforward. This clause allows us to juxtapose all of the above lines to make one Beta graph. That is, the graphs above may be one single Beta graph. Note that the non-overlapping condition for juxtaposition needs to be observed. For example, the following graphs (though they are Beta graphs) would not be Beta graphs obtained by the application of the third clause to some of the graphs in the above (that is, we would not interpret graph (5) as the juxtaposition of two graphs (1) and (3), or the graph (6) as the juxtaposition of the graphs (2) and (3)):

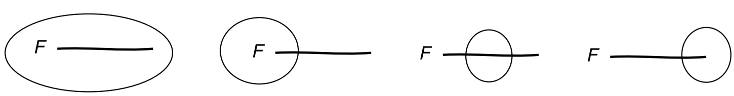
(5)

(6)

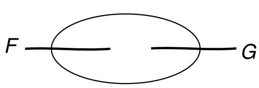
The fourth clause introduces another new syntactic device of the Beta system, a predicate symbol. Suppose *F* is a unary, *G* a binary and *H* a ternary predicate. Then, if *F* is written at one loose end, *G* where two loose ends meet, and *H* at the joint of three loose ends, the result is a Beta graph, as follows:

To put it another way, an *n*-ary predicate has *n* hooks to be filled. As long as the hooks of a predicate symbol are all filled with loose ends of LIs, the number of LIs does not have to equal the number of the hooks of a given predicate. For *F*, one LI is used, for *G* two LIs, but for the ternary predicate *H*, only two LIs. However, the following is not a Beta graph, since *F* has only one hook to fill:

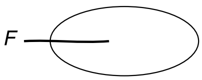
The fifth clause, cut closure for $\mathscr{G}_\beta$, is slightly different from cut closure for $\mathscr{G}_\alpha$. Given graph *G*, the Alpha system allows us to enclose *G* with a cut completely, but not partially. By contrast, the Beta system allows us to draw a cut which encloses *G* partially as well (as long as the cut does not cross a predicate symbol). Therefore, the following graphs are well formed:[8]



The following is a Beta graph by application of clauses 2, 3, 4, and 5:[9]



However, since there is a condition in the juxtaposition closure clause that no part of juxtaposed graphs should overlap with each other, we cannot say that graph above is the result of juxtaposing the following two graphs (7) and (8):

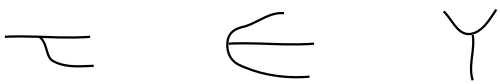


(7)

(8)

Peirce provided a linear notation for the Alpha system:[10] For graph G, a single cut of a graph G may be expressed as '$[G]$'. Since a cut may enclose any subpart of G in the Beta system, and accordingly an LI may cross a cut, Peirce's linear notation for the Beta system allows a line to cross a bracket. Thus, '$-[-G]$' and '$[G'-]-[-G'']$' are examples of the linear notation for Beta graphs.[11] However, we can predict that the multiplicity of ends of LIs and their branches makes Peirce's linear notation for Beta graphs much more complicated than that for Alpha graphs.

The sixth clause of the definition of the set of Beta graphs tells us that an LI may branch. Therefore, all of the following are Beta graphs:



A comparison between my new definition and Zeman's well-known recursive definition of Beta graphs is in order.[12]

Zeman's Definition of Beta Graphs

The set of Beta graphs, $\mathscr{G}^\star$, is the smallest set satisfying the following:[13]

Z1 An empty space is in $\mathscr{G}^\star$.

Z2 A line of identity is in $\mathscr{G}^\star$.

Z3 A branch connecting three LIs is in $\mathscr{G}^\star$.

Z4 Predicate closure An n-ary predicate symbol written at the joint of n separate LIs is in $\mathscr{G}^\star$.

Z5 Complete cut closure If G is in $\mathscr{G}^\star$, then a single cut of G (Zeman writes '$S(G)$') is in $\mathscr{G}^\star$.

Z6 Juxtaposition closure If G_1 is in $\mathscr{G}^\star, \ldots,$ and G_n is in $\mathscr{G}^\star$, then the juxtaposition of these n graphs $G_1, \ldots, G_n$ (Zeman writes '$J(G_1 \ldots G_n)$') is also in $\mathscr{G}^\star$.

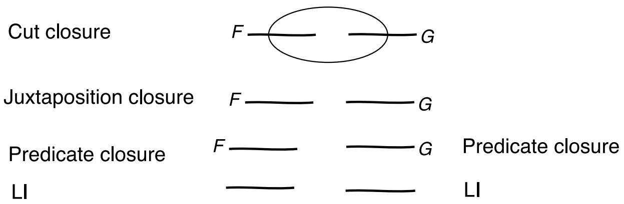


Figure 3.1
Our definition.

Z7 Connecting closure If G is in $\mathscr{G}^{\star}$ and G has n $(2 \leq n)$ loose ends, then a graph in which two loose ends in G are connected without crossing the same cut more than once is also in $\mathscr{G}^{\star}$, and the graph has $n - 2$ loose ends.

An example will illustrate how these two definitions provide different histories for the same Beta graph (see figures 3.1 and 3.2).

Zeman has three basic and four inductive clauses, while I have two basic and four inductive clauses. The first two basic clauses and the juxtaposition-closure clause are the same in both definitions. There are several differences, though, and the question arises about the equivalence. First, I do not have a clause which does the same job as Zeman's connecting closure does. Does my definition generate fewer Beta graphs than Zeman's does? Second, Zeman's third basic clause is covered by my branch closure, but not vice versa. Zeman's predicate closure requires n separate LIs, while mine requires n loose ends. Again, Zeman's complete cut closure is more restricted than my cut closure, since the former only allows one to enclose the entire (not a partial) graph. Does my definition generate more than Zeman's?

Equivalence Zeman's definition and my definition of Beta graphs have the same extension.

Proof We need to show (i) $\mathscr{G}^{\star} \subseteq \mathscr{G}_{\beta}$, and (ii) $\mathscr{G}_{\beta} \subseteq \mathscr{G}^{\star}$.

(i) As mentioned above, the only formulation in Zeman's definition which my definition does not seem to cover is his last clause, connecting closure. There are three cases:

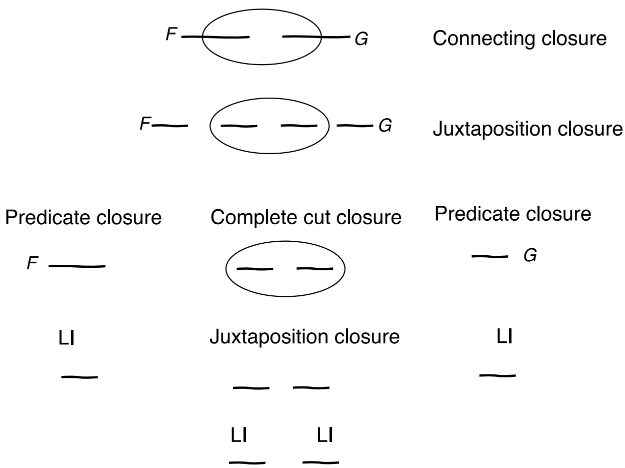
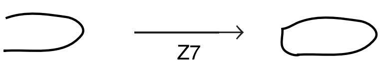


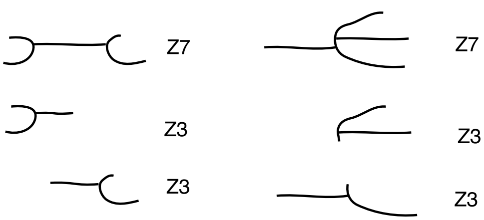
Figure 3.2
Zeman's definition.

(1) Consider the case in which Zeman's connecting-closure clause produces a cycle. For example,



For the definition of $\mathcal{G}_\beta$, this case is covered by one of the basic clauses, the LI clause. Recall that my LI clause covers the case of a cycle.

(2) Consider the case in which the connection does not involve passing through a cut. For example,



For the definition of $\mathcal{G}_\beta$, this case is covered by my branch-closure clause. Without rule Z7, Zeman's definition would produce only a line of identity and a line with three branches.

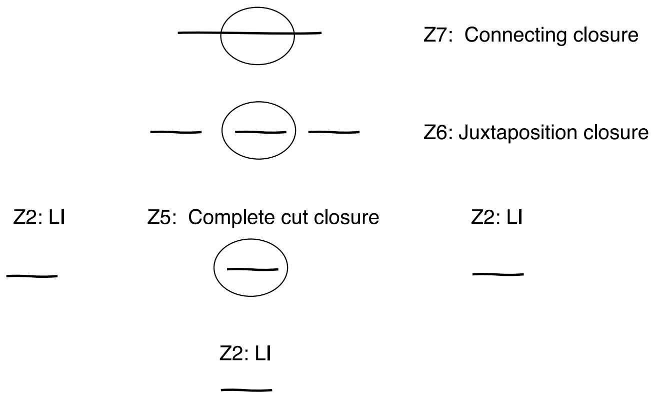


Figure 3.3
Connecting two loose ends through a cut.

(3) Consider a case in which the two loose ends are connected through a cut, for example, that in figure 3.3.

Since my cut closure allows a cut to be drawn around a proper subpart of a graph, Zeman's connecting-closure clause is not needed. The reason why Zeman needed this case of connection is that his complete-cut closure allows us to draw a cut enclosing the whole graph, but not a proper subpart of the graph.

Therefore, whatever is produced by Zeman's definition can be produced by our rules, and so $\mathscr{G}^{\star} \subseteq \mathscr{G}_{\beta}$.

(ii) Now we would like to show that whatever is produced by my definition can be produced by Zeman's rules. My first three rules correspond to Zeman's clauses 1, 2, and 6, respectively. We will examine the other three closures one by one:

(1) Predicate closure: My predicate-closure clause allows us to write an n-ary predicate symbol at the joint of n loose ends, while Zeman's predicate-closure clause requires that an n-ary predicate symbol be written at the joint of n *separate* LIs. However, Zeman has the connecting closure clause, which may connect any loose ends when needed. For example, if P is a binary predicate, then the graph on the left side is in $\mathscr{G}_{\beta}$ by our clauses

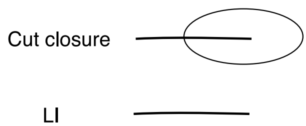


Figure 3.4
Our definition.

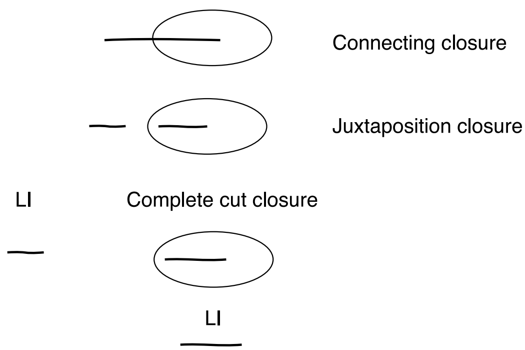


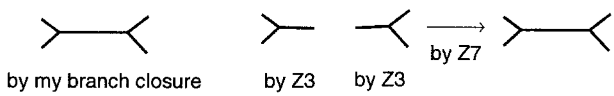
Figure 3.5
Zeman's definition.

2 and 4. Zeman's predicate closure requires a different construction history for this graph, but it is in $\mathscr{G}^{\star}$:

(2) Cut closure: As mentioned before, my cut-closure clause allows us to draw a cut in any subpart of a graph, while Zeman's complete cut closure requires that a cut enclose the whole graph. Again, Zeman's connecting-closure clause permits us to connect loose ends of lines through one and the same cut once. (See the example in figures 3.4 and 3.5.)

(3) Branch closure: This closure clause allows us to draw any number of branches in a graph. In Zeman's definition of $\mathscr{G}^{\star}$, a line and a line with three branches seem to be produced. How can Zeman introduce a line with more than three branches? Peirce's

own theorem Irreducibility of Triads answers this question: The theorem says, "[E]very polyad higher than a triad can be analyzed into triads, though not every triad can be analyzed into dyads."[14] Hence, having a clause for three branches is necessary. When an n branch ($4 \leq n$) may be drawn by my definition, this theorem tells us that we may break this network of LIs into more than one network of LIs with each of them having at most three branches. Now with the help of Zeman's connecting-closure clause, these networks may be connected. The following is an example:



We thus know that a graph produced by my definition is in the set $\mathscr{G}^{\star}$, and so $\mathscr{G}_{\beta} \subseteq \mathscr{G}^{\star}$. □

3.2 The symbolicity of Existential Graphs

As discussed in detail in the second chapter, Peirce made it clear that a logical system is likely to consist of more than one kind of a sign. He pointed out that even an algebraic system has iconic aspects. It will be an interesting inquiry to sort out the elements of symbolicity, if any, and the elements of iconicity in Peirce's own graphical system. As a matter of fact, Peirce admits that EG is not entirely iconic: "The system [EG], of which the slightest possible sketch has been given, is not so iconoidal as the so-called Euler's diagram; but it is by far the best *general* system which has yet been devised."[15]

The history of the development of EG shows us how Peirce struggled with different alternatives for symbolic and iconic elements for the system. We find Peirce's discussion of graphical representation of relatives in his letters of 1882.[16] Peirce's first graphical system, called "Entitative Graphs," was presented in 1897.[17] Some ideas in these earlier discussions survived as features of Entitative Graphs, while others did not. Also, some features of Entitative Graphs have been kept in EG, while others have not.

I will show that both transitions—one from Peirce's preliminary discussions to Entitative Graphs and the other from Entitative Graphs to EG—involve the replacement of some symbolic features with iconic ones. But, strikingly, we do not find the opposite; no iconic features are replaced with symbolic ones. In this section I mainly focus on the symbolic representing features, some of which are replaced in the several stages of Peirce's development of graphical systems, and I will discuss the iconic features of EG in the next section. Interestingly, as we will see soon, even Peirce's final system, EG, is not completely free from symbolic features. After discussing why these symbolic features are retained throughout the process, I will conclude that EG, with both symbolic and iconic features, is a formal *heterogeneous* representation system. If so, it is not surprising to see Peirce refer to EG as "My Chef d'Oeuvre" in light of his support for mixtures of signs in a logical system.

Peirce's original discussions of graphs, which are far from being comprehensive, are centered around the representation of existential and universal statements. The main points of these discussions are summarized by Roberts in the following way:

Note that the lines represent individuals—persons, in this case; and note that the lines, when simply drawn on the sheet, are to be read 'something' or 'someone'.... Figs. 1 and 2 further illustrate the use of the line as a sign of 'something', and Figs. 3 and 4 illustrate Peirce's notational device for expressing 'everything'. The interpretations given below are Peirce's.

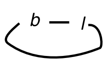


Fig. 1

Fig. 2

Fig. 3

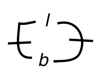


Fig. 4

Fig. 1 means 'something is at once benefactor and loved of something, that is, something is benefactor of a lover of itself'.... Fig. 2 means 'something is a lover of itself'.... Fig. 3 means 'everything is a lover of itself'.... In this system, to draw a bar through a line is to make it a sign of 'everything'.... Fig. 4 means 'everything is either a lover or a benefactor of everything'....[18]

As Roberts correctly pointed out, the use of identity lines is retained in EG,[19] and I will argue in the next section that this is an iconic feature of EG.

Which part, then, of the original ideas in this discussion was abandoned either in Entitative Graphs or in EG? The notation for the representation of 'everything', that is, a bar through a line as in Roberts' figures 3 and 4, is not found in either system. This notational device is symbolic rather than iconic. Given the meaning of an identity line,[20] a bar drawn through an identity line does not resemble the meaning of 'everything', which this bar represents. That is, this notation is not iconic at all. For example, in the above quotation, given the meaning of figure 2, we have no way to tell from inspection alone what figure 3 means, since the meaning of a bar is arbitrarily stipulated, which is the essential nature of symbols. This notational device of a bar is dropped in Peirce's first graphical system, Entitative Graphs.

The letters of 1882 give only a sketch of Peirce's main ideas of graphical systems and are far from being complete. First of all, the representation of negation was not even mentioned at all. Second, it is not at all clear how figure 1 represents conjunctive information while figure 4 represents disjunctive information. We will see how these two questions were answered in Peirce's first graphical system.

Roberts suspects that a notational invention for the representation of negation made it possible for Peirce to turn these preliminary discussions into a graphical system.[21] As explained in the first section of this chapter, a cut represents negation in EG, and this is also the case with his previous system, Entitative Graphs. Is a cut a symbol or an icon for negation?

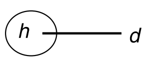
Zeman's following explanation is the closest attempt to find iconicity in this logical notation, the cut:

> The cuts are discontinuities on the sheet of assertion, and they are meant by Peirce to *correspond to* discontinuities in reality. The non-existent, the unactualized, is in a definite sense discontinuous with the existent, the actual insofar as it is not part of the universe of existent individuals. In general, a cut as a break in the continuity of any graphical area *indicates* a certain break in continuity between what is inside and what is outside of it.[22]

Assuming that there is no continuity between actual and non-actual reality, Zeman seems to think that Peirce chooses the cut to represent this discontinuity, by isolating non-actual propositions inside a cut. However, Zeman does not have a firm ground to claim that this representation is iconic, since all he could claim for the relation between the cut and non-existent fact is "corresponds to" or "indicates." Both of these relations are far from being a sufficient condition for a sign to be an icon, since we may say that a symbol too corresponds to or indicates a certain fact or object.

Even though Peirce adopts a broad meaning of ‘resemblance’ as a criterion for iconicity, it is not easy to explain what it means for any sign to resemble a negative fact. Can a negative fact be iconized? How can we visualize a negative fact itself? This is a classic problem with negation. For example, we may visualize the fact that it is sunny, but can we visualize the fact that it is not sunny? We may visualize the fact that it is raining, and this fact implies the fact that it is not sunny. This suggests that any notational device for negation is bound to be a symbol.

How does Peirce handle the other issue, the distinction between conjunctive and disjunctive information, in Entitative Graphs? Even though Peirce did not focus on this issue directly in his paper “The logic of relatives” (1897), where this system was presented for first time, his solution to the problem of how to represent conjunctive and disjunctive information becomes clear in the system. He presents a structure for conditional propositions as one of the most basic graphs:

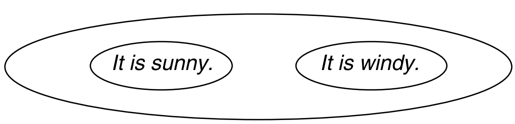


Here, *h* is the monad “____ is a man,” and *d* is the monad “____ is mortal,” The antecedent is completely enclosed, and the meaning is “Anything whatever, if it be a man, is mortal.”[23]

Since this proposition is equivalent to “Anything whatever, it is *either* not a man *or* mortal,” in this system juxtaposition represents disjunctive information. That is, the following Entitative Graph represents the proposition “It is sunny *or* windy.”

It is sunny.   *It is windy.*

Then, the proposition “It is sunny *and* windy” is represented in the following way:[24]



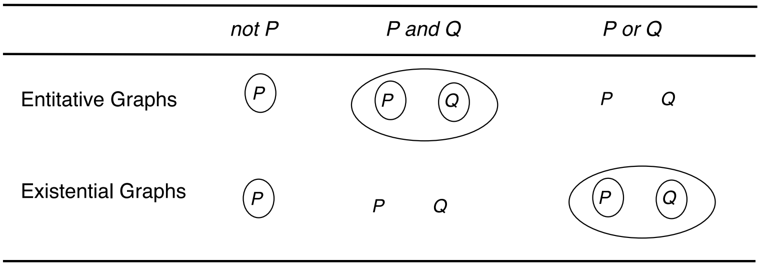
This is opposite from how EG represents conjunctive and disjunctive information. Before we evaluate the symbolicity of the notational devices of these two systems, we need to discuss the differences between conjunctive and disjunctive information.[25]

Suppose that it is sunny and windy. In this situation, we may observe *both* that it is sunny *and* that it is windy, but not conjunction itself. In reality, there is no represented fact corresponding to the English conjunctive word 'and'. Instead, facts are just accumulated, and nothing else is needed to amalgamate these facts. On the other hand, in English we use the word 'and' to convey conjunctive information. This is the case with symbolic logic as well. The connectives '$\wedge$', '&', and '$\cdot$' are used for this purpose.

Suppose that it is sunny or windy and that we take a picture to verify this information. If this information is correct, we will get a picture of a situation conforming to one of the following three possibilities: (i) it is sunny and not windy, (ii) it is windy and not sunny, or (iii) it is sunny and windy. However, we will never get a picture of the situation in which the disjunctive fact that it is sunny or windy is displayed. Interestingly, any one of these three possible pictures conveys more than the original disjunctive information. We can find a situation that supports disjunctive information, but this situation contains more than the disjunctive information.

Reality displays only conjunctive information, not disjunctive information. In the case of conjunctive information, as seen above, facts are accumulated in reality without any "connective fact" itself. We cannot find anything in reality that resembles connectivity. Therefore, any notational device for conjunction is a symbol, not an icon. Thus, if a system is to be iconic in representing conjunctive information, it should not introduce a sign but display pieces of information. This is the closest resemblance we can expect between reality and a representation system as far as conjunctive information goes. On the other hand, in the case of disjunctive information, no reality displays a disjunctive fact itself. Therefore, any form of representation for disjunctive information—whether a sign is introduced or not—is bound to be symbolic, since there is nothing to be resembled in this case.

Now let's summarize Entitative Graphs' and EGs' representation for negative, conjunctive, and disjunctive information.



Entitative Graphs adopt a sign for conjunctive, but not for disjunctive information, while EGs go the other way. Therefore, both conjunctive and disjunctive information are represented with their own conventions in Entitative Graphs, while in EGs conjunctive information is represented (rather) iconically[26] and disjunctive information symbolically. Why did Peirce change from one way to the other? There seems to be no reason other than that Peirce decided to move towards iconic representation as much as possible.[27]

3.3 The iconicity of Existential Graphs

This section discusses Peirce's own stated intuitions about EG, which, I claim, take these representation systems to be iconic and, accordingly, distinct from symbolic predicate languages. On the basis of Peirce's passages, I reconstruct three iconic components in EG, all of which are involved in representing quantified information. These three visual features will become important when I evaluate the existing reading methods of EG and present my new reading algorithms in the following two chapters.

3.3.1 Lines of identity

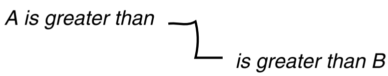
It is well known that relations are more difficult to represent in a graphical system than are properties. For example, Euler or Venn graphs are effective diagrammatic representation systems as long as inferences are limited to properties.[28] In the discussion of Venn diagrams, which precedes his writing on EG, Peirce points out this limitation as one of the problems of Venn's system: "[The Venn system] does not extend to the logic of relatives [i.e., relations]."[29]

Hence, Peirce's first motivation for the Beta system is to represent relations graphically.[30]

> In many reasonings it becomes necessary to write a copulative proposition in which two members relate to the same individual so as to distinguish these members.[31]

His example is the proposition that *A* is greater than something that is greater than *B*. After suggesting unsuccessful graphs, finally he says,

> [I]t is necessary that the signs of them should be connected in fact. No way of doing this can be more perfectly *iconic* than that exemplified in Fig. 78 [the following graph].[32]



This syntactic device, i.e., the line connecting two predicates, represents *one and the same* object, which *A* is greater than and at the same time which is greater than *B*. This is why Peirce calls this line a *line of identity*, which is a clear case of iconic representation of the sameness. This intuition is captured in the following convention:

Convention A heavy line, called a line of identity, shall be a graph asserting the numerical identity of the individuals denoted by its two extremities.[33]

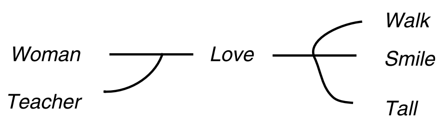
This new vocabulary serves two important functions. First, the introduction of lines adds expressive power to the Beta system by allowing us to represent individuals and relations, while his Alpha system cannot. At the same time, this device, representing sameness *iconically*, makes a clear distinction between Beta graphs as a graphical system and other predicate languages as symbolic systems. While in a symbolic system tokens of the same type of letter represent the same individual that each token denotes, in Peirce's Beta system a network of lines represents the same individual denoted by each branch of the network. That is, Peirce's system *graphically* represents numerical identity with one connected network. Peirce's following passage intends to express these roles that a line of identity plays in the Beta system:

The beta-part of the system of existential graphs is distinguished by its taking cognizance of individual identity and individuality.[34]

For example, in the following first-order sentence,

$$\exists x \exists y[\text{Tall}(y) \wedge \text{Love}(x, y) \wedge \text{Walk}(y) \wedge \text{Teacher}(x) \wedge \text{Smile}(y) \wedge \text{Woman}(x)],$$

we need to keep track of tokens of x and y more carefully than when the identity of each individual is represented (in a visually much clearer way) by graphical lines in the following Beta graph:



Visual clarity in representing identity is more easily obtained in the Beta system than in symbolic languages, thanks to this iconic representation with lines of identity. A contrast between the use of variables and the use of a line is well captured by Zeman:

The line of identity shows that two verbs have the same subject by actually *hooking them together*; the selective [a bound variable Peirce adopted] (and the variable of ordinary quantification theory) performs this function by putting identically appearing symbols in the *two* subject positions. . . . Peirce wants a sign which will not merely be *conventionally* understood as signifying identity, but which will *naturally*, *iconically* represent identity.[35]

For representing identity, writing tokens of the same type of a variable in more than one spot is conventional,[36] while connecting more than one spot by one and the same line is iconic. It is also interesting to notice that Zeman claims that the latter is a *natural* way of representing identity.

Peirce aimed to make the Beta system as iconic as possible,[37] and Peirce's intuition about the use of lines stems from this motivation. Peirce's valuable intuition should be respected when we read off and manipulate Beta graphs. This will be one of the important criteria when we evaluate existing reading methods and transformation rules of the Beta system. As we will see in chapter 5, some existing methods of translation from a graph to a sentence unfortunately ignore this iconic feature to generate more than one type of variable for one and the same network of identity lines.

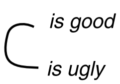
3.3.2 Existential versus universal quantifiers

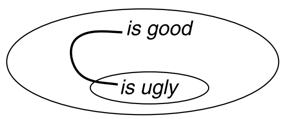
As Roberts says at the beginning of his chapter on the Beta part of EG, this system is "a treatment of the functional or predicate calculus, the logic of quantification."[38] However, the most striking aspect of Beta graphs compared with other predicate languages is that this graphical system does not introduce any syntactic device which corresponds to the quantifiers of symbolic languages. Then, we might ask, how can this system be about the logic of quantifiers? This question leads us to the next iconic feature of EG, to which I now turn.

Rather than adopting one more syntactic device for quantification, Peirce relies on visual features that already exist in a graph:

> [*A*]*ny line of identity whose outermost part is evenly enclosed refers to* **something,** *and any one whose outermost part is oddly enclosed refers to* **anything** *there may be*.[39]

The following exemplify even enclosure and odd enclosure:[40]





Since the outermost part of the line in the left graph is evenly enclosed (in this case, enclosed zero times), this graph says that something good is ugly. On the other hand, the graph on the right side says that everything good is ugly: the outermost part of the line, next to 'is good', is enclosed once, i.e., oddly.

Existential and universal quantifiers are represented by whether the outermost part of an LI is evenly or oddly enclosed, without any additional iconic or symbolic object being introduced. Some might wonder whether this choice is conventional. If so, they might argue, the representation of these two different quantifiers is not intrinsically different from the arbitrary convention for symbolically representing these quantifiers, i.e., '∃' and '∀'. But Peirce's decision for these two quantifiers is not arbitrary at all, for the following reason: As explained in §3.2 and in §3.3.1 respectively, a cut represents a negation and a line of identity represents something. As pointed out in each section, a cut is symbolic representation and a line of identity is iconic. From this heterogeneous representation, the representation for existential and universal quantifiers is easily obtained. When the outermost part of a line lies in an evenly enclosed area, it is interpreted as 'something', since it is an iconic representation of 'one and the same object'. If the outermost part of a line is enclosed by an odd number of cuts, we get the negation of the existential quantifier, that is, 'it is not the case that something is....' Hence, 'everything is not ...' is the interpretation of this line. Quite similarly, the symbol '$\forall$' is adopted as the abbreviation for the symbols '$\neg\exists\neg$'. Without the introduction of the new symbol '$\forall$', expressions sometimes could become too cumbersome to read off easily. But, unlike with symbolic languages, Peirce's graphical system does not need another syntactic device to represent this universal quantifier. All we have to do is to read off a visual aspect of the graph, i.e., the outermost part of an LI being oddly enclosed.

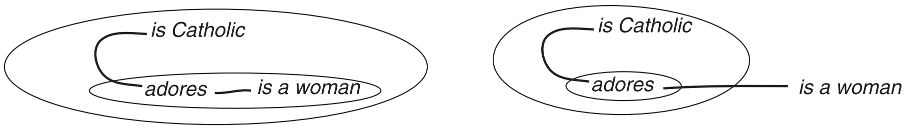
I take this intuition to reflect one of the most important features that makes this system graphic. However, as I will show in chapter 5, disappointingly none of the existing reading methods systematically implements Peirce's intuitive distinction between existential and universal quantifiers. The neglect is quite surprising, since Peirce spelled out this strong point of the system in the quotation above and many authors are not only perfectly aware of Peirce's intuition but even quote Peirce's discussion in their works.[41] I will investigate the cause of this neglect after carefully examining each reading method.

3.3.3 Scope of quantifiers

In linear symbolic systems, the scope of syntactic operations is clarified by the linear order of the symbols, or sometimes by this order along with the use of parentheses. Since Beta graphs' representation of quantifiers is not linear, the next question is how to represent the order among different quantifiers. For example, how does this system make a distinction between the proposition $\forall x \exists y L(x, y)$ and the proposition $\exists y \forall x L(x, y)$? Roberts refers to Peirce's manuscript where Peirce addresses this problem:

> According to Peirce, the difficulty of representing the logic of relatives graphically lies entirely in the circumstance that it is necessary to distinguish between [the above] two propositions. (Ms. 481)[42]

Peirce solves this problem in the Beta system by appealing to a visual feature about where the outermost part of a line is written: The less enclosed the outermost part of a line is, the larger the scope that the line gets. As an example, let us compare the following graphs:[43]



In the first graph, the line whose outermost part is oddly enclosed is less enclosed than the line whose outermost part is evenly enclosed. Therefore, the universal quantifier has larger scope than the existential quantifier. In the second graph, it is the other way around. While $\forall x(\text{Catholic}(x) \rightarrow \exists y[\text{Adores}(x, y) \wedge \text{Woman}(y)])$ is what the first graph expresses, the second says $\exists y(\text{Woman}(y) \wedge \forall x[\text{Catholic}(x) \rightarrow \text{Adores}(x, y)])$.

All the literature on the Beta system emphasizes the importance of scope. However, no literature highlights this feature as a crucial aspect that differentiates this visual system from symbolic languages. Note that Peirce takes advantage of a visually natural homomorphic relation between the position of lines and their scopes: If the outermost part of line l_1 is *less enclosed* than the outermost part of line l_2, then the reading of the LI l_1 has *larger scope* than l_2.

4

The Alpha System Reconsidered

In the previous two chapters, we have explored the theoretical aspects of Peirce's EG. In this and the following chapters, we take up more practical aspects of EG and suggest improvements in the actual use of EG as a deductive system.

EG's Alpha and Beta systems have been proven to be sound and complete deductive systems equivalent to sentential and first-order languages.[1] However, logicians have strongly preferred symbolic systems to these graphical systems. Why? Some might cite a long-standing prejudice against non-symbolic representation to explain logicians' preference for symbolic languages. However, since we are interested in how the prejudice plays a role in this specific case, I investigate more concrete reasons behind this preference.

It is true that Peirce himself did not consider EG a calculus.[2] However, this cannot be the main reason why logicians have strongly preferred symbolic systems over Peirce's graphical system. First, Peirce's refusal to categorize EG as a calculus follows from his own distinction between logical systems and calculi, which not many of us would accept. If this distinction is not assumed, we do not have to accept the narrow role Peirce assigned to EG. More importantly, when Peirce said that good logical systems cannot be good deductive calculi, he meant to include both symbolic and graphical systems under logical systems. Hence, according to Peirce, there is no reason why symbolic logical systems are used as calculi, but graphical systems are not. However, his symbolic first-order system, as modified, has been used by working logicians. Hence, Peirce's own discouragement cannot be a major obstacle for the use of his EG system. Third, there is not much evidence for logicians' even being aware of Peirce's passages where he stipulated the goal of making EG not a calculus but a logical system.

There are much more serious criticisms against the use of EG as a deductive system. Logicians commonly complain that Peirce's EG is too complicated to put to actual use. This complaint comes from the following two specific sources: First, graphs of EG are not easy to read off. These graphs have been considered to be less natural or less intuitive than symbolic sentences. Similarly, another complaint against Peirce's graphical system is that the inference rules of EG are not as intuitive as the inference rules of natural deductive systems.[3] In addition to these specific criticisms, many have also believed that Peirce's graphical system lacks the kind of visual power present in a system like Euler diagrams.[4]

These criticisms are well taken in the following respect: The more complicated the interpretation or the manipulation of a system is, the less efficient the system is. Granting the legitimacy of these criticisms, in chapters 4 and 5, I present new methods to understand the Alpha and Beta systems, respectively. I will show that the superficial relation drawn between the Alpha system and a sentential language with only two connectives, i.e., conjunction and negation,[5] is not only misleading but has prevented us from fully using this graphical system. Therefore, I suggest that one utilize more fine-grained visual aspects of the systems for a more direct and more natural reading algorithm, for a more efficient formulation of inference rules, and for a more intuitive interpretation of these rules. The main idea is that if we take advantage of visual distinctions already present in EG, the system becomes more intuitive and useful. Some of these features were discussed by Peirce and other scholars, but none of the existing works has implemented all of them, nor have they presented a systematic way for understanding the visual features.

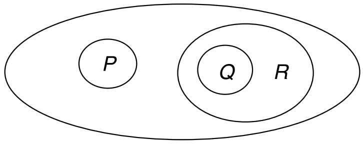
The first section summarizes the traditional reading method for this graphical system to see why reading off an Alpha graph has been considered challenging. In the second section, while uncovering important visual features of the Alpha system, I present a new reading method, one which *directly* gives us a translation in a useful and important form. In the third section, I draw attention to another visual feature of the system, called "scroll," which was discussed by Peirce and mentioned in other literature but has never been fully developed. I then show that we can combine all the visual features discussed in this chapter to create another reading method of the Alpha system. I claim that this method is more *natural* than the other two examined in the first and the second sections. The fourth section then re-states the inference rules of EG in a more natural and intuitive way than they have been by Peirce or commentators. Finally, in the fifth section I make a stronger claim for the Alpha system: This graphical system is more efficient than symbolic languages for *certain* purposes.

4.1 "Endoporeutic" reading

It has long been believed that Peirce's Alpha system has only two kinds of syntactic operation: cut and juxtaposition. By interpreting cuts as negation and juxtaposition as conjunction, the meaning of a graph usually turns out to be a negation of a sentence which consists of several conjuncts, each of which is again a negation of a sentence, etc. These nested negations and conjunctions add only unnecessary complication, just as a formula which uses only negation and conjunction symbols is often more cumbersome than a formula which has the same meaning but uses disjunction and conditional symbols as well. As we know, this is a main reason why we usually use more connectives than '¬' and '∧', even though these two are enough to express every Boolean function.

Hence, the meaning of a graph has been obtained through two stages: (i) translate a cut into a negation and a juxtaposition into a conjunction to get a sentence with nested negation and conjunction symbols, and (ii) if the result looks complicated, simplify it by adopting an additional connective to get an equivalent sentence. Let me illustrate this process through an example.

Example 4.1 The following graph is translated into the sentence '$\neg(\neg P \wedge \neg(\neg Q \wedge R))$':



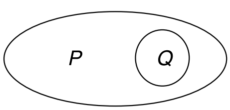
Next, DeMorgan's Law is used to get a simpler-looking sentence, '$(P \vee (\neg Q \wedge R))$'.

The impression has endured that Peirce's system has fewer syntactic devices, which makes the two translation stages ((i) and (ii) above) inevitable. Therefore, in most cases, Peirce's graphs are bound to be read indirectly.

Interestingly, while we have heavily relied on first-order symbolic logic for understanding EG, no effort has been made to find anything analogous in Peirce's system to '$\vee$' and '$\rightarrow$' in symbolic systems. Many, without further inquiry, assume that Alpha graphs are analogous to symbolic sentences with only '$\neg$' and '$\wedge$'. Therefore, just as we prefer sentences using more connectives, Alpha graphs have not been our choice for practical use.

A similar problem arises when we translate a sentential formula into a graph. Given the simple sentence '$(P \vee Q)$', we need to transform this sentence into the more complicated looking sentence '$\neg(\neg P \wedge \neg Q)$', since it is believed that only two kinds of syntactic devices exist in the Alpha system. That is, whenever we need to represent information in this graphical system, we need to manipulate the information in terms of negation and conjunction. Again, this does not seem to be the simplest way to represent given information.

Closely related to these well-known complaints, I would like to raise a more fundamental problem with this traditional reading method, one which I propose to solve in the next section. In his pioneering work *The Existential Graphs of Charles S. Peirce* (1973), Don Roberts emphasizes the importance of the order of negation and conjunction for translating the following Peircean graph (which means 'It is not the case that *both P* is true *and Q* is false', i.e., 'Either *P* is false or *Q* is true'):



Notice that we do *not* read it: 'Q is true and P is false', even though Q is evenly enclosed and P is oddly enclosed. . . . [W]e read the graph from the outside (or least enclosed part) and we proceed inwardly. . . .[6]

Roberts suggests that the following passage from Peirce is evidence that this method, "a method to which Peirce gave the name 'endoporeutic',"[7] was what Peirce originally had in mind:

> The interpretation of Existential graphs is *endoporeutic*, that is proceeds inwardly; so that a nest sucks the meaning from without inwards unto its centre, as a sponge absorbs water.[8]

The interpretation of the above graph as '*Q* is true and *P* is false' is not correct, because this interpretation confuses the scope of negation and the scope of conjunction. When we consider negation and conjunction to be the basic relations of a graph, the question of how to proceed, i.e., whether outwardly or inwardly, is crucial, since the negation of conjunctions is different from the conjunction of negations. No challenge has been made against this *endoporeutic* reading method.

I claim that this reading method has prevented us from benefiting from the visual power of the system. My main criticism is that this method does not reflect visually clear facts in the system. As Roberts points out in the above quotation, in the cited graph it is true that *Q* is evenly enclosed and *P* is oddly enclosed. However, the endoporeutic reading does not directly reflect this visually clear fact at all, and what is worse, it leaves the impression that this visual fact is misleading. To put this criticism in a more general way, the "endoporeutic" method forces us to read a graph in only one way. For example, we are supposed to read off Roberts' example in the following way only: This graph is a cut of [the juxtaposition of *P* and a cut of *Q*]. However, there is another possible reading. We might say that this graph has two cuts, with *Q* enclosed by both cuts and *P* enclosed by the outer cut only. This reading is not directly reflected in the endoporeutic reading.

4.2 "Negation normal form" reading

In this section I identify overlooked visual properties and recognize more syntactic distinctions in EG than on the traditional approach. I do not introduce new syntactic devices into EG, but only observe significant visual differences already present in EG. In particular, the discovery that the system represents conjunction and disjunction *visually* by different kinds of juxtaposition undermines the traditional view that Peirce's

Alpha system uses only one kind of juxtaposition, that is, juxtaposition as conjunction. The reading method presented below will always give us a translation in negation normal form.[9] I call this translation method the "NNF reading."

4.2.1 Conjunctive and disjunctive juxtapositions

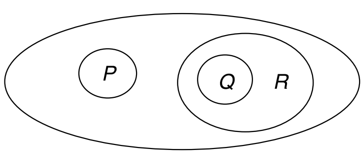
I suggest that the following two features be interpreted *directly*: (i) the visual distinction between when a sentence symbol is written in an area enclosed by an even number of cuts and when it is written in an area enclosed by an odd number of cuts, and (ii) the visual fact that some juxtapositions occur in an evenly enclosed area and some in an oddly enclosed area. Recall that the traditional method reflects these features in the translation not directly but only *indirectly*: Whatever is asserted in an oddly enclosed area is translated as a negation of what is asserted in an evenly enclosed area. In this sense, I claim that the reading method presented in the current section is more *direct* than the endoporeutic method discussed in the previous section.

For the direct interpretation of these features, let us introduce the following definitions:

Definition 4.1 Let X be a subgraph of a Peircean Alpha graph.[10] Then X is in an *E-area* iff X is enclosed by an even number of cuts, and X is in an *O-area* iff X is enclosed by an odd number of cuts.

Definition 4.2 Let X and Y be disjoint subgraphs of a given graph. Then the juxtaposition of X and Y is an *E-jux* iff X and Y are juxtaposed in an E-area, and the juxtaposition of X and Y is an *O-jux* iff X and Y are juxtaposed in an O-area.

Let us return to the graph of example 4.1 in the previous section, repeated here, to see how these definitions are applied:



The letters P, Q, and R are asserted in an E-area, in an O-area, and in an E-area, respectively. R and a cut of Q are juxtaposed in an E-area, and we therefore say that the juxtaposition of these two subgraphs is an E-jux. On the other hand, the juxtaposition between a cut of P and a cut of [a cut of Q and R] is an O-jux.

Using these definitions, I will present a translation algorithm based upon the following two principles. First, for every token of a letter x in a graph, the translation of the graph contains either 'x' or '$\neg x$' as a basic component, depending upon whether the token x is written in an E-area or in an O-area respectively. Second, we translate E-jux and O-jux into '$\wedge$' and '$\vee$', respectively.

Before stating the translation algorithm, some definitions are in order. First, we define the set of Peircean Alpha graphs, $\mathscr{G}'$, in the following way so that our translation algorithm may be defined recursively on this set.

Set $\mathscr{G}'$ is the smallest set satisfying the following:[11]

1. (a) For every sentence symbol A_i, A_i and a single cut of A_i (we write '$[A_i]$', following Peirce's linear notation)[12] are in $\mathscr{G}'$.

   (b) An empty space (we write '$\emptyset_{sp}$') and an empty cut (we write '[ ]') are in $\mathscr{G}'$. (By empty space, I mean a blank sheet of paper. By empty cut, I mean a cut with a blank inside.)
2. If G is in $\mathscr{G}'$, then a double cut of G (we write '$[[G]]$') is also in $\mathscr{G}'$.
3. If G_1 is in $\mathscr{G}', \ldots,$ and G_n is in $\mathscr{G}'$, and none of $G_1, \ldots,$ and G_n is an empty space,[13] then the juxtaposition of the n graphs $G_1, \ldots, G_n$ (we write '$G_1 \ldots G_n$') and a single cut of $G_1 \ldots G_n$ (we write '$[G_1 \ldots G_n]$') are also in $\mathscr{G}'$.

Definition 4.3 Graph G is a *simple graph* iff G is a sentence letter, a single cut of a sentence letter, an empty space, or an empty cut.

Definition 4.4 Formula x is a *simple formula* iff x is a sentence letter, $\neg y$ (for a sentence letter y), $\top$, or $\bot$.

NNF Reading The following function reads off a simple graph into a simple formula.

i. Let A_i be a *token* of a letter in graph G. Then $f(A_i) = A_i$, and $f([A_i]) = \neg A_i$.
ii. $f(\emptyset_{sp}) = \top$
iii. $f([\ ]) = \bot$

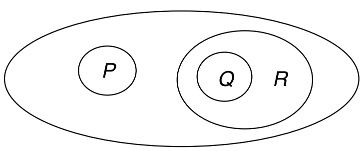
Now we extend this function f to $\bar{f}$ to obtain translations of Alpha graphs.

1. $\bar{f}(G) = f(G)$ if G is a simple graph
2. $\bar{f}([[G]]) = \bar{f}(G)$
3. $\bar{f}(G_1 \ldots G_n) = \bar{f}(G_1) \wedge \cdots \wedge \bar{f}(G_n)$[14]
4. $\bar{f}([G_1 \ldots G_n]) = \bar{f}([G_1]) \vee \cdots \vee \bar{f}([G_n])$

The first clause reads off the visual fact of whether a sentence symbol is in an E-area or in an O-area to obtain a simple formula. The second clause erases any double cut of a graph. The third and the fourth clauses read off E-jux and O-jux to obtain conjunction and disjunction respectively. Unlike the endoporeutic reading, this reading does not yield any nested negation and conjunction. A negation, if any, comes in only as a simple formula by the first clause.

Let me illustrate how these clauses work through several examples.

Example 4.2 At the beginning of the previous section, we discussed the following graph, which is translated into '$\neg(\neg P \wedge \neg(\neg Q \wedge R))$' by the traditional reading method:

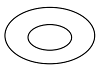


Function $\bar{f}$ defined above is applied as follows:

$$
\begin{aligned}
\bar{f}([[P]\,[[Q]\,R]]) &= \bar{f}([[P]]) \vee \bar{f}([[[Q]\,R]]) && \text{by clause 4}\\
&= \bar{f}(P) \vee \bar{f}([Q]\,R) && \text{by clause 2}\\
&= \bar{f}(P) \vee (\bar{f}([Q]) \wedge \bar{f}(R)) && \text{by clause 3}\\
&= f(P) \vee (f([Q]) \wedge f(R)) && \text{by clause 1}\\
&= P \vee (\neg Q \wedge R) && \text{by def. of } f
\end{aligned}
$$

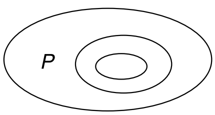
Since both P and R are written in an E-area, and Q in an O-area, the translation has P, R, and $\neg Q$ as its components. Since the juxtaposition of $[P]$ and $[[Q]\,R]$ is an O-jux, it is translated into '$\vee$', while the juxtaposition of $[Q]$ and R, an E-jux, is read off as '$\wedge$'.

Example 4.3 If we have only one double cut as follows, then we will get the empty space by erasing the double cut by the second clause:



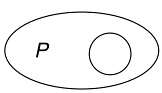
Then the function f translates this empty space into '$\top$'.

Example 4.4 We will see how the following double cut is treated:



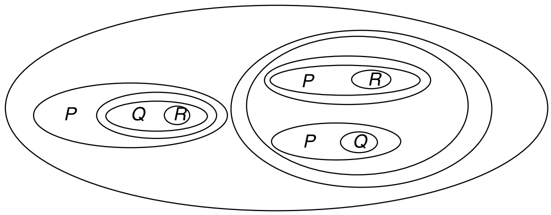
$$\begin{aligned} \bar{f}([P\,[[\ ]]]) &= \bar{f}([P]) \vee \bar{f}([[[\ ]]]) && \text{by clause 4} \\ &= \bar{f}([P]) \vee \bar{f}([\ ]) && \text{by clause 2} \\ &= f([P]) \vee f([\ ]) && \text{by clause 1} \\ &= \neg P \vee \bot && \text{by def. of } f \end{aligned}$$

Example 4.5 Suppose we have an empty cut as follows:

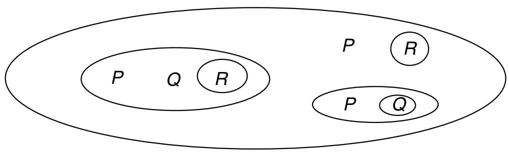


$$\begin{aligned} \bar{f}([P[\ ]]) &= \bar{f}([P]) \vee \bar{f}([[\ ]]) && \text{by clause 4} \\ &= \bar{f}([P]) \vee \bar{f}(\emptyset_{sp}) && \text{by clause 2} \\ &= f([P]) \vee f(\emptyset_{sp}) && \text{by clause 1} \\ &= \neg P \vee \top && \text{by def. of } f \end{aligned}$$

Example 4.6 Let us test the following complicated looking graph chosen from Roberts' examples:[15]



According to our new reading method, a double cut may be erased any time, since nothing could be written between the outside and inside cuts in the reading process. So, let's erase all the double cuts of the above graph to get the following:



$$
\begin{aligned}
&\bar{f}([\,[P\,Q\,[R]]\,P\,[R]\,[P\,[Q]]\,]) \\
&\quad = \bar{f}([[P\,Q\,[R]]]) \vee \bar{f}([P]) \vee \bar{f}([[R]]) \vee \bar{f}([[P\,[Q]]]) && \text{by clause 4} \\
&\quad = \bar{f}(P\,Q\,[R]) \vee \bar{f}([P]) \vee \bar{f}(R) \vee \bar{f}(P\,[Q]) && \text{by clause 2} \\
&\quad = (\bar{f}(P) \wedge \bar{f}(Q) \wedge \bar{f}([R])) \vee \bar{f}([P]) \vee \bar{f}(R) \\
&\qquad \vee (\bar{f}(P) \wedge \bar{f}([Q])) && \text{by clause 3} \\
&\quad = (f(P) \wedge f(Q) \wedge f([R])) \vee f([P]) \vee f(R) \\
&\qquad \vee (f(P) \wedge f([Q])) && \text{by clause 1} \\
&\quad = (P \wedge Q \wedge \neg R) \vee \neg P \vee R \vee (P \wedge \neg Q) && \text{by def. of } f
\end{aligned}
$$

Very importantly, a graph always gets translated into a sentence in *negation normal form*. This is much more advantageous than a sentence with nested negations and conjunctions. In the next subsection, we will check whether this direct reading is legitimate.

4.2.2 Semantics

To prove the legitimacy of this new reading, we need to check the following: Given a truth assignment v, the same truth value is assigned to one and the same graph by both the endoporeutic and the NNF readings. The following definition is in order.

Definition 4.5 Let v be a function which assigns T or F to each sentence symbol. Also, $v(\top) = \text{T}$ and $v(\bot) = \text{F}$. Then $\bar{v}$ is an extension of v which assigns T or F to a formula of a sentential language with $\neg$, $\wedge$, and $\vee$:

$$\begin{aligned}
\bar{v}(\alpha) &= v(\alpha) && \text{if } \alpha \text{ is a sentence symbol, } \top, \text{ or } \bot. \\
\bar{v}(\neg\alpha) &= \text{T} && \text{if } \bar{v}(\alpha) = \text{F} \\
&= \text{F} && \text{otherwise} \\
\bar{v}(\alpha \wedge \beta) &= \text{T} && \text{if } \bar{v}(\alpha) = \text{T and } \bar{v}(\beta) = \text{T} \\
&= \text{F} && \text{otherwise} \\
\bar{v}(\alpha \vee \beta) &= \text{T} && \text{if } \bar{v}(\alpha) = \text{T or } \bar{v}(\beta) = \text{T} \\
&= \text{F} && \text{otherwise}
\end{aligned}$$

To make the comparison clear, I will also define the endoporeutic-reading algorithm in an analogous way to the NNF-reading algorithm $\bar{f}$ presented in the previous subsection.

Endoporeutic Reading The following function h reads off a sentence letter or an empty space:

i. Let A_i be a *token* of a letter in graph G. Then $h(A_i) = A_i$.

ii. $h(\emptyset_{sp}) = \top$

Now we extend this function h to $\bar{h}$ to obtain translations of Alpha graphs.

1. $\bar{h}(G) = h(G)$ if h is a sentence symbol or an empty space
2. $\bar{h}([G]) = \neg\bar{h}(G)$
3. $\bar{h}(G_1 \ldots G_n) = \bar{h}(G_1) \wedge \cdots \wedge \bar{h}(G_n)$

With the two functions $\bar{f}$ and $\bar{h}$ for different reading methods, we define two semantics in the following way:

Definition 4.6 Let G be an Alpha graph, and v be a truth assignment. Then,

$v \models_f G$ iff $\bar{v}(\bar{f}(G)) = \text{T}$

$v \models_h G$ iff $\bar{v}(\bar{h}(G)) = \text{T}$

What we want to prove is that these two semantics, $\models_f$ and $\models_h$, are equivalent to each other. For this proof, we need to clarify the relation between graphs G and $[G]$ with respect to $\models_f$. In the case of $v \models_h G$, obviously $v \models_h G$ if and only if $v \not\models_h [G]$, since $\bar{h}([G]) = \neg\bar{h}(G)$. However, reading algorithm $\bar{f}$ is not defined for graph $[G]$ except when G is a simple graph. Hence, the following lemma needs to be proven:

Lemma 4.1 Given a truth assignment v, for every graph G, $v \models_f G$ iff $v \not\models_f [G]$.

Proof Induction on graphs.

Basis: When G is a simple graph, i.e., a sentence letter, a cut of a sentence letter, an empty space, or an empty cut, it is obviously true.

Inductive step: According to the definition of $\mathscr{G}'$, there are three inductive cases.[16]

(i) IH: Suppose that $v \models_f G$ iff $v \not\models_f [G]$.

We want to show that $v \models_f [[G]]$ iff $v \not\models_f [[[G]]]$.

$v \models_f [[G]]$	iff	$v \models_f G$	since $\bar{f}([[G]]) = \bar{f}(G)$
	iff	$v \not\models_f [G]$	by IH
	iff	$v \not\models_f [[[G]]]$	since $\bar{f}([[[G]]]) = \bar{f}([G])$

(ii) IH: Suppose that $v \models_f G_1$ iff $v \not\models_f [G_1], \ldots,$ and $v \models_f G_n$ iff $v \not\models_f [G_n]$.

We want to show that $v \models_f G_1 \ldots G_n$ iff $v \not\models_f [G_1 \ldots G_n]$.

$v \models_f G_1 \ldots G_n$	iff	$v \models_f G_1, \ldots,$ and $v \models_f G_n$	by def. $\models_f$ and $\bar{f}$
	iff	$v \not\models_f [G_1], \ldots,$ and $v \not\models_f [G_n]$	by IH
	iff	$\bar{v}(\bar{f}([G_1])) = \text{F}, \ldots,$ and $\bar{v}(\bar{f}([G_n])) = \text{F}$	by def. $\models_f$
	iff	$\bar{v}(\bar{f}([G_1]) \vee \cdots \vee \bar{f}([G_n])) = \text{F}$	by def. $\bar{v}$
	iff	$\bar{v}(\bar{f}([G_1 \ldots G_n])) = \text{F}$	by def. $\bar{f}$
	iff	$v \not\models_f [G_1 \ldots G_n]$	by def. $\models_f$

(iii) IH: Suppose that $v \models_f G_1$ iff $v \not\models_f [G_1], \ldots,$ and $v \models_f G_n$ iff $v \not\models_f [G_n]$.

We want to show that $v \models_f [G_1 \ldots G_n]$ iff $v \not\models_f [[G_1 \ldots G_n]]$.

$v \models_f [G_1 \ldots G_n]$	iff	$\bar{v}(\bar{f}([G_1 \ldots G_n])) = \text{T}$	by def. $\models_f$
	iff	$v \models_f [G_1], \ldots,$ or $v \models_f [G_n]$	by def. $\models_f$ and $\bar{f}$
	iff	$v \not\models_f G_1, \ldots,$ or $v \not\models_f G_n$	by IH
	iff	$v \not\models_f G_1 \ldots G_n$	by def. $\models_f$
	iff	$v \not\models_f [[G_1 \ldots G_n]]$	by def. $\bar{f}$ □

We are ready to prove the equivalence of the two readings:

Proposition 4.2 Given a truth assignment v, for every graph G, $v \models_h G$ iff $v \models_f G$.

Proof Induction on graphs.[17]

Basis: When G is a sentence symbol or an empty space, it is obviously true.

Inductive step:

(i) IH: Suppose that $v \models_h G$ iff $v \models_f G$.

We want to show that $v \models_h [G]$ iff $v \models_f [G]$.

$v \models_h [G]$	iff	$v \not\models_h G$	by def. $\models_h$
	iff	$v \not\models_f G$	by IH
	iff	$v \models_f [G]$	by lemma 4.1

(ii) IH: Suppose that $v \models_h G_1$ iff $v \models_f G_1, \ldots,$ and $v \models_h G_n$ iff $v \models_f G_n$.

We want to show that $v \models_h G_1 \ldots G_n$ iff $v \models_f G_1 \ldots G_n$.

$v \models_h G_1 \ldots G_n$	iff	$v \models_h G_1, \ldots,$ and $v \models_h G_n$	by def. $\models_h$
	iff	$v \models_f G_1, \ldots,$ and $v \models_f G_n$	by IH
	iff	$v \models_f G_1 \ldots G_n$	by def. $\models_f$ □

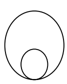
4.3 Multiple readings

In the following, I discuss another important visual feature, called a "scroll" by Peirce. This feature was discussed by Peirce, but subsequent works have not paid much attention to it. In the first subsection, after exploring Peirce's discussion on the visual feature "scroll," I incorporate Peirce's idea to present another reading algorithm. In the second subsection, I claim that the new algorithm gives us more flexibility in reading off Alpha graphs, and I show why this flexibility yields the most *natural* reading method for the Alpha system of EG. The discussion also reveals important differences among different modes of representation.

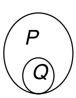
4.3.1 Scroll as conditional

When Peirce presented his EG, he provided conventions that tell us how graphs should be interpreted. His conventions correspond to an informal semantics of the system. How to interpret juxtaposition and cut are included, and these two conventions have been adopted in most studies of EG, as discussed in the first section of the current chapter. However, it is interesting to notice that Peirce had a separate, third convention for material implication. Let me introduce Roberts' presentation of this convention:

C4 [Convention 4][18] concerns the way in which EG is to express the conditional proposition. . . . How shall we graph 'If *P* then *Q*'? In order to assert it we must place it on SA [a sheet of assertion, i.e., a sheet of paper on which a graph is drawn]. But since 'If *P* then *Q*' asserts neither *P* nor *Q*, we must be careful not to scribe them on SA. We get what we want by means of what Peirce called a "scroll"—"two closed lines one inside the other" (Ms 450, p. 14), like this:



. . . Suppose now that we place the graph *Q* in the innermost circle, and the graph *P* in the outermost compartment, obtaining this graph:



Note that we have succeeded in diagram[m]ing both *P* and *Q*, yet not on the surface of SA itself. And we agree to express in this way the conditional proposition *de inesse*: If *P* then *Q*. . . . Here then is C4: *The scroll is the sign of a conditional proposition de inesse* (that is, of material implication) (Ms 450, p. 14).[19]

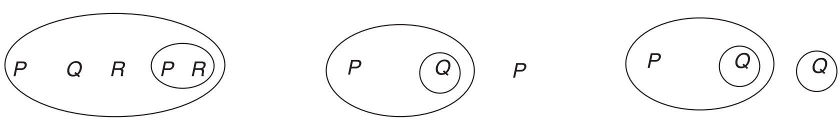
An interesting question is why Peirce had this convention even though the rest of the conventions are enough to get a correct reading of EG's

Alpha graphs. He could have said that the above graph expresses 'It is false that [P is true and Q is false]', and this reading could be obtained by the other conventions.

Clearly, Peirce's intention was to get a direct reading of a scroll, rather than taking a detour through nested cuts and juxtapositions. Quite surprisingly, the importance of the scroll convention has not been discussed. Reading off cut as negation and juxtaposition as conjunction is enough to demonstrate the soundness and completeness of the Alpha system, which satisfies theoretical curiosity about the system. However, when we put EG to practical use as a deductive representation system, the scroll reading adds much more convenience and naturalness not only to the existing reading method but to our understanding of Peirce's transformation rules.

Even though the set of $\neg$ and $\rightarrow$ is truth-functionally complete, we do not want to use only cuts and scrolls in reading an Alpha graph. First of all, many graphs do not have a scroll. In that case, we would have to introduce a step to draw a double cut, which is a legitimate but awkward process. It is much more convenient and natural to *add* a scroll to the existing list of conventions for cut and juxtaposition. The following example will illustrate this point.

Example 4.7 Reading off cut, conjunctive juxtaposition, and scroll, graph (1) represents '$(P \wedge Q \wedge R) \rightarrow (P \wedge R)$', graph (2) '$(P \rightarrow Q) \wedge P$', and graph (3) '$(P \rightarrow Q) \wedge \neg Q$'.

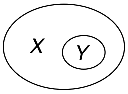


(1) (2) (3)

Therefore, importantly, we know easily that the first graph represents a tautology, the second modus ponens reasoning, and the third modus tollens.

We put Peirce's convention for scroll in a more general way:

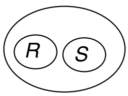
Reading a scroll Let X and Y be graphs. Suppose that each of them is translated into α and β, respectively. Then the following graph with a scroll is translated into $(\alpha \rightarrow \beta)$:



It is clear that reading off a scroll gets us the same result as we get by using only cuts and juxtapositions, since $\neg(\alpha \wedge \neg\beta)$ is logically equivalent to $(\alpha \rightarrow \beta)$.

Now we are interested in adding this re-discovered visual feature to the features discussed in the previous sections so that we may obtain the multiple readings of an Alpha graph. Let us start with a simple example:

Example 4.8 The following graph is translated into the following four logically equivalent formulas:



1. $\neg(\neg R \wedge \neg S)$ (endoporeutic reading)
2. $R \vee S$ (NNF reading)
3. $\neg R \rightarrow S$ (reading off a scroll $[X[S]]$, where $X = [R]$)
4. $\neg S \rightarrow R$ (reading off a scroll $[[R]X]$, where $X = [S]$)

I suggest that different reading methods be combined so that the reader may be given flexibility in reading off a graph. For this flexibility to be implemented, I first define the set of Peircean Alpha graphs with more inductive clauses than we had in the previous sections, and call it $\mathcal{G}''$.

The set $\mathcal{G}''$ is the smallest set satisfying the following:

1. An empty space is in $\mathcal{G}''$.
2. Any letter is in $\mathcal{G}''$.
3. If G is in $\mathcal{G}''$, then a single cut of G ('$[G]$') is in $\mathcal{G}''$.
4. If G_1 is in $\mathcal{G}''$ and G_2 is in $\mathcal{G}''$, then all of the following are also in $\mathcal{G}''$: $G_1\, G_2$, $[G_1\, G_2]$, $[G_1\, [G_2]]$, $[[G_1][G_2]]$.

Clearly, the set $\mathcal{G}''$ is equivalent to the set $\mathcal{G}_\alpha$ in §3.1.1 and to the set $\mathcal{G}'$ in §4.2.1. However, there is an important difference. Recall that how

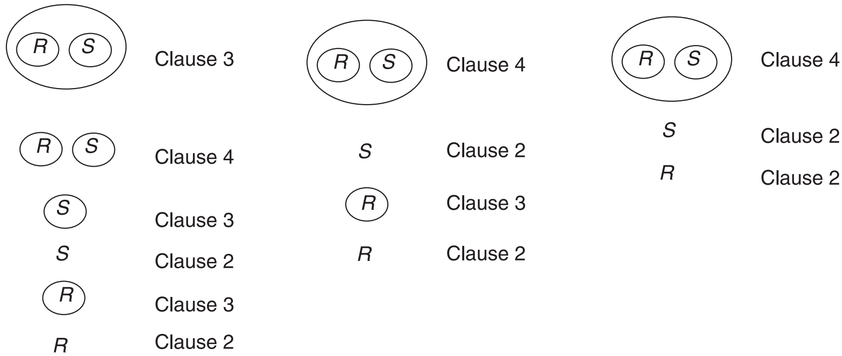


Figure 4.1
Three possible building histories for the graph in example 4.8.

both $\mathscr{G}_\alpha$ and $\mathscr{G}'$ are defined guarantees a unique building tree for each graph. However, this new inductive definition does not. Let me give three possible, but not exhaustive, building histories (among many more) for the graph in example 4.8 (see figure 4.1).

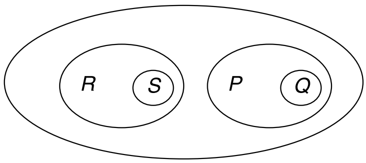
Based on this new definition of Alpha graphs, another reading method can be generated.

Multiple-Readings Algorithm Let X and Y be Alpha graphs.

1. If X is an empty space, its translation is $\top$.
2. If X is a sentence letter, its translation is X.
3. If a translation of X is α, then a translation of $[X]$ is $\neg\alpha$.
4. If a translation of X is α and a translation of Y is β, then
   (a) a translation of $X\,Y$ is $(\alpha \wedge \beta)$,
   (b) a translation of $[X\,Y]$ is $(\neg\alpha \vee \neg\beta)$,
   (c) a translation of $[X\,[Y]]$ (i.e., scroll with X in the outside cut and Y in the inner cut)[20] is $(\alpha \rightarrow \beta)$, and
   (d) a translation of $[[X]\,[Y]]$ is $(\alpha \vee \beta)$.

Let's see how these clauses are applied through the following example:

Example 4.9 Depending upon which *form* we happen to think the graph belongs to, different clauses are applied. The following are some of the possibilities:



(i) This graph has a form of $[[X]\,[Y]]$, where $X = R\,[S]$ and $Y = P\,[Q]$. Then clause 4(d) will be applied. For the translations of X and Y, clauses 4(a), 3, and 2 are applied so that we get $R \wedge \neg S$ as X's translation and $P \wedge \neg Q$ as Y's translation. Therefore, by clause 4(d), $(R \wedge \neg S) \vee (P \wedge \neg Q)$ is a translation of this graph.[21]

(ii) If we see a scroll in the graph, we consider it as a form of $[X\,[Y]]$, where $X = [R\,[S]]$ and $Y = P\,[Q]$. In this case, clause 4(c) is being applied. Y is translated into $P \wedge \neg Q$. For the translation of $[R\,[S]]$, there are several choices:

(1) We may apply clauses 4(c) and 2 to get $R \rightarrow S$. Then, by applying clause 4(c) again, we get $(R \rightarrow S) \rightarrow (P \wedge \neg Q)$.

(2) We may apply clauses 3, 4(a), and 2 to get $\neg(R \wedge \neg S)$. Then, by applying clause 4(c), we get $\neg(R \wedge \neg S) \rightarrow (P \wedge \neg Q)$.

(3) We may apply clauses 4(b), 3, and 2 to get $\neg R \vee S$. Then, by applying clause 4(c), we get $(\neg R \vee S) \rightarrow (P \wedge \neg Q)$.

(iii) Again, we see a scroll, but in a slightly different form, $[[X]\,Y]$, where $X = R\,[S]$ and $Y = [P\,[Q]]$. (I will leave details to the reader.)

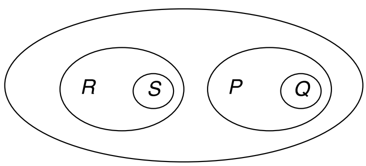
(iv) This graph has a structure of $[X]$, where $X = [R\,[S]]\,[P\,[Q]]$. That is, we start with clause 3. Again, there are several ways to read $[R\,[S]]\,[P\,[Q]]$. I leave the details to the reader.

4.3.2 The Multiple Carving Principle

As its name suggests, the Multiple-Readings Algorithm presented above allows us to translate one and the same graph into more than one sentence. Hence, the method does not guarantee a unique translation. Does this cause any problem? I claim that it does not. On the contrary, this flexibility renders the Multiple-Readings method the most natural reading method among those proposed.

Both the endoporeutic and NNF readings force the reader to read off a graph in one and only one way, which guarantees a unique translation of the graph. On the other hand, in many cases, this is not the natural way of perceiving the graph. If no instruction is given, not everybody would perceive a graph from outside inwards (the only perception that the endoporeutic reading allows us to adopt), and not everybody would pay attention to E-jux or O-jux visual features (the one used by the NNF reading). Sometimes a scroll might catch our eyes first, etc. We sometimes mix these different kinds of perceptions. A graph, unlike a language in a linear system, can be perceived in more than one way, depending on how the reader happens to carve up the given graph. Let me call the non-uniqueness of how to perceive one and the same graph the 'Multiple Carving Principle'.

I will illustrate the Multiple Carving Principle with the following graph, used in example 4.9:



How do you perceive this single graph? As Peirce originally instructed us, some readers might perceive the graph from outside inwards. That is, the outermost cut would catch the reader's eye first, a juxtaposition between two cuts second, etc. Then the endoporeutic reading translates this graph into the formula '$\neg(\neg(R \wedge \neg S) \wedge \neg(P \wedge \neg Q))$'. On the other hand, a reader might pay attention to the following visual features: (i) Both R and P are written in an E-area, while S and Q are in an O-area. (ii) There are two E-juxs and one O-jux. Then the NNF algorithm guides her to directly obtain the formula '$(R \wedge \neg S) \vee (P \wedge \neg Q)$'.

However, these two cases are far from being exhaustive. The graph may be carved up in many more different ways. Let me illustrate only a few of them, by highlighting the cuts which catch the reader's eyes first (see figure 4.2).[22]

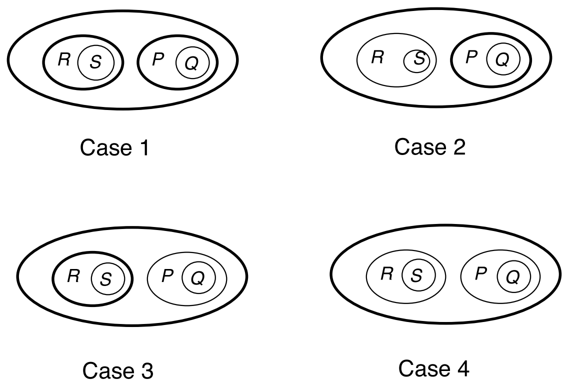


Figure 4.2
A few possible ways of carving up the graph in example 4.9.

In the first case one perceives the whole graph as $[[X]\,[Y]]$, where $X = R\,[S]$ and $Y = P\,[Q]$. On the other hand, one might see a scroll first and perceive this whole graph as $[X\,[Y]]$, where $X = [R\,[S]]$ and $Y = P\,[Q]$, or as $[[X]\,Y]$, where $X = R\,[S]$ and $Y = [P\,[Q]]$. These correspond to cases 2 and 3, respectively. Or one might perceive this graph as $[X]$, where $X = [R\,[S]]\,[P\,[Q]]$. In this case the outermost cut catches the reader's eye to provide a different pattern.

The Multiple-Readings method aims to exploit such differences in perceiving graphs. I argue that this reading method, which reflects the Multiple Carving Principle for Peircean graphs, is more *natural* than any reading method based on a single way of perceiving a graph.

According to the Multiple Carving Principle, we have different ways of perceiving one and the same graph. How the reader percieves a graph determines which way is the most convenient and the most natural reading for him. For example, in the above example, the reader in the first case would find clause 4(d) in the Multiple-Readings Algorithm[23] the easiest to start with. On the other hand, if the reader reads off a scroll first, then clause 4(c) would be the most helpful rule for him.

The Multiple-Readings Algorithm covers all these possible ways of carving up one and the same graph and allows the reader to choose whichever way is the most intuitive, and therefore most natural, to him. This flexibility is lacking both in the traditional method and in the NNF reading method. Accordingly, I claim that the Multiple-Readings method based on the Multiple Carving Principle is more *natural* than either of the previous methods.

It is interesting to notice that this issue of flexibility does not arise in a linear symbolic language. On the contrary, a symbolic system is very careful to prevent multiple readings of a formula, since it would yield ambiguity. Sentential languages are defined so that each sentence may have one and only one way of being read off, and the semantics is built on this unique readability. For example, a string '$P \wedge Q \vee R$' is not well-formed, and accordingly, we do not bother with its semantics. To secure unique readability, parentheses or prefix notations have been adopted so that one and only one way of parceling up a sentence is available.

Neither the endoporeutic nor the NNF reading violates this principle of symbolic languages. Each of these readings defines the well-formed graphs so that each graph may have one and only one way of being composed. Each algorithm is based on its own uniquely defined inductive syntactic history. As examined above, there is no theoretical defect in either method.

However, I have shown that there is no need to keep the unique-readability principle in the Alpha system. On the contrary, the Multiple Carving Principle is found in the actual practice of perceiving a graph. It would be quite unnatural to prevent the reader from carving up a graph in any way other than how a particular reading algorithm can handle the graph. Hence, no reading method of graphs based on unique readability can avoid being needlessly arbitrary, in spite of its theoretical correctness. This is why the Multiple-Readings method is practically more desirable for graphical systems than other reading algorithms.

This improvement of the reading of Alpha graphs is a prime example demonstrating the need to identify fundamental differences among various representation systems and to take different approaches according to the nature of a representation system. Why has existing work on EG failed to reflect the Multiple Carving Principle? I argue that this neglect is directly related to our prejudice for symbolic systems, which has led us to understand a graphical system in terms of symbolic systems. Both the endoporeutic and the NNF readings blindly followed the practice of symbolic languages, i.e., the unique-readability principle, without any need to. Only when we became free from unique readability and found a different principle appropriate to a graphical system did we find a new and more natural way of understanding EG.

4.4 Transformation rules

With more visual features than are traditionally recognized, we made the reading of graphs easier, and we will now see that we can state inference rules with a more specific symmetry and can understand them more intuitively. As stated at the beginning of the chapter, logicians have been doubtful of the practical use of the inference rules of EG. This section examines the main reasons behind this skepticism and reconstitutes Alpha graphs as a more effective deductive system.

4.4.1 A natural deductive system versus the Alpha system

Why have we strongly preferred natural deductive systems to other systems (either symbolic or not)? The inference rules of a natural deductive system are easier to understand and, accordingly, easier to apply than those of other systems. All natural deductive rules of propositional logic are built around connectives. The rules instruct the user how to eliminate and how to introduce each of these syntactic objects. This is the most natural way to manipulate the system's meaningful units, because formulas are inductively built out of sentential symbols and connectives. When inference rules are based on how meaningful units of a system are built, it is easy for the user to choose an appropriate rule in each proof step. For example, when a premise is a conjunctive sentence, we know that the conjunction-elimination rule is needed to use this premise. If the conclusion is a conditional sentence, we know that the conditional-introduction rule is needed somewhere in the proof. This is a main reason why *natural* deductive systems received their name.

What makes the rules of a system natural in this sense is not always the same as what makes a system efficacious. However, in the case of a natural deductive system, the naturalness of the inference rules—that the procedures for manipulating or obtaining a formula is based on how the formula is composed—clearly adds efficacy to the system: A deduction is more easily obtained and, accordingly, the system is more efficacious than if there were no correspondence between inference rules and the structure of the system's units. Interestingly and importantly, in the case of EG, a syntactic history of how a graph is built does *not* provide us with the most efficient way to state the rules of inference. The *naturalness* of this graphical system is not the same as the naturalness of a symbolic system, as we will see soon.

At the same time, the symmetry between elimination and introduction of each connective has been considered to lend a system a certain elegance. The symmetry itself is not directly related to the efficacy of a system, even though one might argue that a symmetric way of stating the rules helps the user to remember them. The crucial issue is what kind of specific symmetry underlies the system. In the existing Alpha system, as we will see below, a symmetry is built in. However, I will show that the existing symmetry does not make EG more efficacious, unlike the symmetry of a natural deductive system. I will then present a different kind of a symmetry based on important visual features of EG in the next subsection.

The inference rules of the Alpha system are summarized as follows:

R1 The rule of erasure Any evenly enclosed graph may be erased.

R2 The rule of insertion Any graph may be scribed on any oddly enclosed area.

R3 The rule of iteration If a graph *P* occurs on SA [the sheet of assertion] or in a nest of cuts, it may be scribed on any area not part of *P*, which is contained by {*P*}.[24]

R4 The rule of deiteration Any graph whose occurrence could be the result of iteration may be erased.

R5 The rule of the double cut The double cut may be inserted around or removed (where it occurs) from any graph on any area.[25]

This list of rules seems to be quite simple—only five rules—and keeps a certain symmetry—erasure versus insertion, and iteration versus deiteration. Why doesn't this simplicity and symmetry increase the efficacy of the Alpha system?

I suggest three main reasons for the mismatch between appearance and reality. First and most importantly, the 'natural' of 'natural deductive system' is misapplied here. Peircean scholars might have found a similarity between the rules of erasure, insertion, iteration, and deiteration (in EG) and the rules of introduction and elimination of connectives (in natural deduction systems). I claim that this superficial similarity does not contribute to the efficacy of a graphical system, since a syntactic history of a graph is fundamentally different from how a formula is composed out of the basic vocabulary of its system.

A propositional symbolic system defines its grammatical formulas inductively by functions introducing connectives. Hence, when the rules are stated in terms of these connectives, it increases the system's efficacy. For example, if the conclusion is $P \wedge Q$, we guess we need to introduce the connective $\wedge$ somewhere in a proof, that is, that the $\wedge$-introduction rule is needed. However, this kind of naturalness cannot be useful in the Alpha system, since a derivational history for an Alpha graph can vary more than the syntactic history for a formula.

Example 4.10 The example in figure 4.3 shows that a graph can have more than one derivational history.[26] Of course, we can define a graph so that there is only one unique construction for each graph. We have seen two different ways of defining well-formed Alpha graphs that produce a unique history of each graph.[27] According to those definitions, record 1 in figure 4.3 is the only legitimate building history for this graph. But unlike with a symbolic sentence, this uniquely defined history does not tell us how to obtain the graph. For example, sometimes we

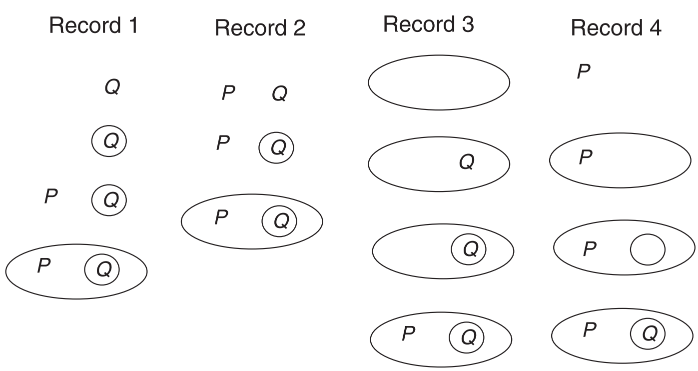


Figure 4.3
Several derivational histories of the same graph.

need to transform the lefthand graph into the righthand graph in the following:

This transformation, which should be permissible, is not reflected in any previously defined unique construction but is a step in the history shown in record 3.

Another reason why Peirce's rules have not been considered efficient is that the symmetries built into Peirce's inference rules are not fine-grained enough to be useful. The only symmetry these rules exhibit is between writing and erasing, which occurs in any kind of manipulations of signs, whether symbols or graphs. Peirce is perfectly aware that this is the only kind of symmetry present in the system: "All our transformation are analyzed into insertions and omissions."[28]

Lastly, the user finds the rules redundant, since the iteration and deiteration rules partially overlap with the insertion and erasure rules. The iteration rule says that an iteration of subgraph *X* is allowed in any area which is enclosed by the area where *X* is written. However, if the area where I need to write *X* happens to be an oddly enclosed area, we do not need the condition that another token of graph *X* is in the same area as, or an area outside of, this O-area, since the rule of insertion allows that directly. A mirror image of this redundancy is to be found between the rule of deiteration and the rule of erasure in the case of an evenly enclosed area.

I claim that these three problems are closely related to the fact that the existing rules did not take full advantage of visual distinctions in graphs. By reading off visual features as much as possible and implementing them in the rules of inference, we may obtain a certain type of efficacy that a graphical system can have over symbolic systems. This is the goal of the next subsection.

4.4.2 The rules reformulated

Replacing Peirce's simple symmetry—erasure versus insertion, and iteration versus deiteration, I re-present Peirce's Alpha inference rules with more specific symmetries built around the visual features of the system.

Unlike symbolic systems, the Alpha system does not introduce new syntactic objects for conjunction or disjunction, which is a strong point of graphical systems over symbolic ones.[29] Instead, these two connectives are represented in terms of a very clear visual feature, that is, whether a juxtaposition between two subparts of a graph takes place in an area enclosed by an even number or by an odd number of cuts. Therefore, we could provide another kind of symmetry by relying on the visual distinction between an E-area and an O-area, which we discussed in the second section.

While a natural deductive system is natural because the rules are presented in terms of the history of the formation of a formula, the Alpha system should not pursue this specific kind of naturalness. To achieve naturalness in a graphical system like EG, visual facts (rather than a history of syntactic composition) should be implemented in the system. A crucial visual feature of EG is whether something is written or erased in an E-area or in an O-area. Therefore, I suggest a new version of the Alpha rules in order to base the inference rules of the system on the following fundamental visual features: what we may *draw* or *erase* in which area, be it an *E-area* or an *O-area*. This will make the rules *natural* for a different reason from why a natural deductive system is natural. This way of naturalness is related to the efficacy of the system.

Reformulated inference rules[30]

1. In an E-area, say area *a*,
   (a) we may *erase* any graph, and
   (b) we may *draw* graph *X* if there is a token of *X*
      (i) in the same area, i.e., area *a*, or
      (ii) in the next-outer area from area *a*.[31]
2. In an O-area, say area *a*,
   (a) we may *erase* graph *X* if there is another token of *X*
      (i) in the same area, i.e., area *a*, or
      (ii) in the next-outer area from area *a*, and
   (b) we may *draw* any graph.
3. A double cut may be erased or drawn around any part of a graph.[32]

Table 4.1
Rules 1 and 2 for Alpha graph *X*

	E-area	O-area
Erase	*X*	*X* if there is another *X* either in the same area or in the next-outer area
Draw	*X* if there is another *X* either in the same area or in the next-outer area	*X*

The symmetries in the first two rules can be summarized as in table 4.1.

Let us compare the reformulated rules with Peirce's original rules and see whether these two sets of rules are equivalent to each other.

My clauses 1(a), 2(b), and 3 correspond to Peirce's R1, R2, and R5, respectively. Semantically, these rules correspond to $\wedge$-elimination, $\vee$-introduction, and the double negation law of symbolic logic, respectively. However, the relations between my 1(b) and Peirce's R3 and between my 2(a) and Peirce's R4 need to be explained.

My rule 1(b) corresponds to only part of Peirce's iteration rule (R3). The iteration rule allows us to redraw any graph occurring on some area either on that area or on any area enclosed by additional cuts. First, notice that Peirce's rule of iteration is somewhat redundant, since it partially overlaps with the insertion rule (R2). The insertion rule allows us to inscribe anything in an O-area. Therefore, we do not need any other condition in order to insert something in an O-area. What matters is what we may insert in an E-area. Rule 1(b) above allows us to draw another token of *X* in an E-area, say area *n*, if a token of *X* exists in the area *n* or in *the next-outer* area from *n*, while Peirce's iteration allows one to copy *X* in *n* if *X* exits in *n* or in *any* outer area. I will show that my reformulated inference rules cover this important aspect of Peirce's iteration rule.

Proposition 4.3 What we may iterate in an E-area by Peirce's iteration rule may also be iterated in the E-area by my inference rules.

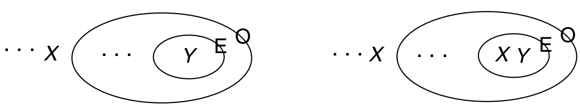
Proof Suppose that we may iterate X in an E-area, say area a, by Peirce's iteration rule. This means that we can have another token of X in an area b such that $a \subseteq b$.[33] We want to show that under the same conditions we may draw a token of X in area a by using my inference rules listed above.

(i) If a is the same area as b, then by the first clause of my rule 1(b), we may draw X in area a.

(ii) If a is not the same area as b but is properly contained by b, we need to consider three different cases. (The other cases are repetitions of these three cases.)

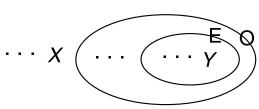
(Case 1) Suppose that area b (where X exists) is the next-outer area from area a. Then the second clause of 1(b) may be applied to draw another token of X in area a.

(Case 2) Suppose that area b is the next-outer area from the area next-outer to area a. That is, Peirce iteration rule says that in the following we may transform the graph on the left side to the graph on the right side:[34]



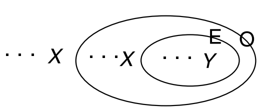
By paying attention to whether it is an E-area or an O-area, we proceed in the transformation by my inference rules as follows:

1.



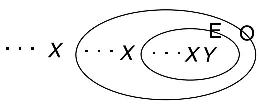
given

2.



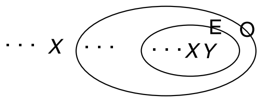
by 2(b)

3.



by 1(b)(ii)

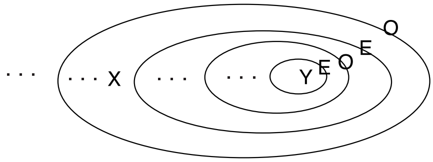
4.



by 2(a)(ii)

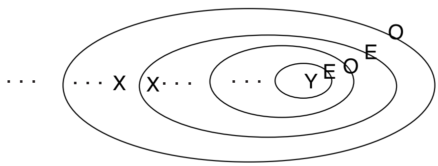
(Case 3) Suppose that X is in the area which is once more further out than in case 2 (i.e., the next-outer area from the next-outer area from the area next-outer to a). Hence, X is in an O-area this time. We apply rule 1(b) first, and the rest of the process is similar to case 2.

1.



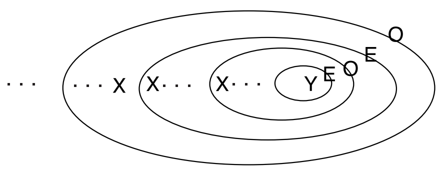
given

2.



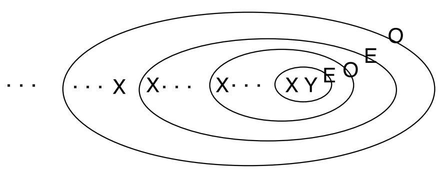
by 1(b)(ii)

3.



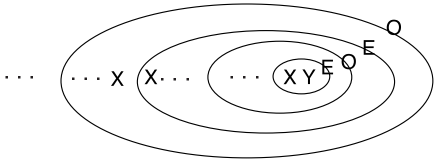
by 2(b)

4.



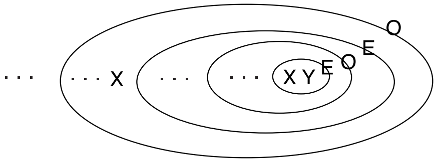
by 1(b)(ii)

5.



by 2(a)(ii)

6.



by 1(a)

□

Rule 2(a) is the mirror image of rule 1(b), just as Peirce's deiteration rule is a counterpart of his iteration rule. Rule 2(a) corresponds to part of Peirce's deiteration rule. Again, Peirce's rule of deiteration is somewhat redundant, since it partially overlaps with the erasure rule.[35] We can show that the important part of Peirce's deiteration rule, i.e., the deiteration in an O-area, is not lost in my list of inference rules.

Proposition 4.4 What we may erase in an O-area by Peirce's deiteration rule may also be erased in the O-area by my inference rules.

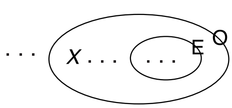
Proof This proof is very similar to the proof of proposition 4.3.

4.4.3 The rules reinterpreted

Many have found the meanings of Peirce's original rules rather obscure. So the soundness of Peirce's Alpha system has not been understood at an intuitive level as the soundness of a natural deductive system has. A main reason for this difference is that the inference rules for a natural deductive system are easily understood through the obvious correspondence with the meaning of each connective, but there has been no attempt at a similar understanding of Peirce's inference rules. With the traditional method of reading graphs by using nested negations and conjunctions, for example, Peirce's iteration rule is hard to understand. I will show that understanding my reformulated inference rules is much easier if we utilize all the visual features discussed in the previous sections.

The first inference rule has two clauses: Clause 1(a) allows us to erase anything in an E-area. As demonstrated in the second section, a juxtaposition in an E-area (we called it "E-jux") corresponds to conjunctive information. Clearly, erasing a conjunct should be valid. Clause 1(b) says we may draw X in an E-area when one of the two conditions is satisfied. This case requires some examination, since adding a conjunct is not a valid step.

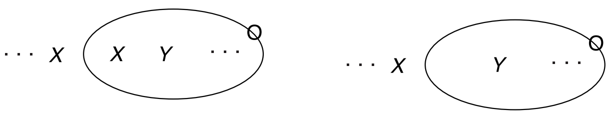
It should be noted that clause 1(b) is quite different from the $\wedge$-introduction rule of symbolic logic. The rule of $\wedge$-introduction allows us to conjoin previously obtained formulas, that is, combining more than one piece of information. However, none of the Alpha rules deals with more than one graph at the same time. Inferences are from one *single* graph to another *single* graph.[36] The first condition of 1(b) corresponds to the following inference in symbolic logic: $\alpha \Longrightarrow (\alpha \wedge \alpha)$. In the case of symbolic logic we do not need this copy rule separately, since the reiteration and $\wedge$-introduction rules together do the job. The second condition —i.e., *if* a token of X is in the next-outer area from the E-area where we want to draw another token of X—is rather complicated. Since a token X is in the next-outer area from an E-area, it must be in an O-area. We thus know that we are dealing with a graph with the following scroll as its subpart:[37]



Since a scroll represents a conditional proposition, this subgraph represents a piece of conditional information with X as an antecedent. Therefore, it is a valid step to infer that X occurs as a consequent as well. This corresponds to the following symbolic derivation: $X \rightarrow Y \Longrightarrow X \rightarrow (Y \wedge X)$. Thus, rule 1(b), i.e., the drawing rule in an E-area, is valid.

There are two main reasons why it is much easier to see the validity of this rule than the validity of Peirce's iteration rule. One is that my rule 1(b) cuts off the redundant part of the iteration rule.[38] The other is that we here make use of a re-discovered visual feature, the scroll.

Rule 2(b) allows us to draw any graph in an O-area. In the second section we saw that O-jux represents disjunctive information. Hence, this rule corresponds to adding a disjunct, which is a valid step. Rule 2(a), however, is not the same as the $\vee$-elimination rule of propositional logic. For the same reason as there is no similar rule in EG to $\wedge$-introduction, there is no single inference rule in this system that has the same function as the $\vee$-elimination rule of symbolic logic: The Alpha system transforms one single graph into another single graph. The first clause of 2(a) is the same as the following inference in symbolic logic: $(\alpha \vee \alpha) \Longrightarrow \alpha$. This inference is obtained by both the reiteration and $\vee$-elimination rules in first-order logic. The second clause says that we may erase a token X in an O-area if another token of X is in the next-outer area to this O-area. That is, in the following, we may transform the graph on left side into the graph on the right side:[39]



According to the reading algorithm of the third section, if α is a translation of X and β a translation of Y, then the part of the graph on the left side will be translated into $(\alpha \wedge (\neg\alpha \vee \neg\beta))$.[40] Then from this we may infer $(\alpha \wedge \neg\beta)$, which is the translation of the part of the graph on the right side and is obtained by erasing the token X of the inner O-area. Therefore, this is a clearly valid rule.

The third rule, erasing and drawing of a double cut, is clearly valid, just as eliminating or introducing a double negation is.

4.4.4 Efficacious graphical systems

Now we will examine in what respect my reformulated rules add to the efficacy of the Alpha system. Through an example, I will illustrate how my rules can be more useful than Peirce's original ones and will show that this efficiency is analogous to the efficiency that the inference rules of symbolic logic have, but with a crucial difference.

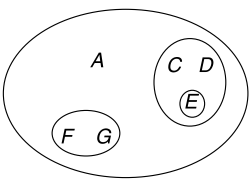
Suppose we would like to obtain sentence 2 from sentence 1:[41]

$$A \rightarrow [(C \wedge D \wedge \neg E) \vee (F \wedge G)] \tag{1}$$

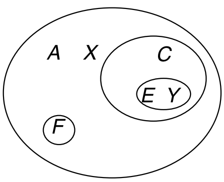
$$(A \wedge X) \rightarrow ([C \wedge \neg(E \wedge Y)] \vee F) \tag{2}$$

A natural deductive system is set up conveniently so that it is quite clear which rules are needed to deduce sentence 2 from sentence 1: Since sentence 2 is a conditional sentence, we need the introduction rule for $\rightarrow$. So, assume the antecedent, $A \wedge X$, and try to obtain the consequent, $([C \wedge \neg(E \wedge Y)] \vee F)$. This can be deduced from the consequent of sentence 1, $((C \wedge D \wedge \neg E) \vee (F \wedge G))$. Since this consequent is a disjunctive sentence, we try to use the $\vee$-elimination rule, and so on. By paying attention to what kinds of sentences we have (that is, a conjunctive, a disjunctive, a negative, or a conditional sentence) and what kind of a sentence we need to obtain, we use either the introduction or the elimination rule for the appropriate connective.

A process that is similar but different in an important respect occurs when we manipulate graphs using our reformulated inference rules. For example, consider these two graphs, which correspond to sentences 1 and 2 above:



(1)



(2)

We compare what we are given, graph (1), and what we aim to obtain, graph (2). The following visual differences are apparent: (i) X is in an O-area in graph 2, while it does not appear in graph 1, (ii) G does not exist in graph 2, while it is in an E-area in graph 1, (iii) Y is in an O-area in graph 2, while it does not appear in graph 1, and (iv) D does not exist in graph 2, while it is in an E-area in graph 1. Therefore, the following manipulations are needed: (i′) X should be drawn in an O-area, (ii′) G in an E-area should be erased, (iii′) Y should be drawn in an O-area, and (iv′) D in an E-area should be erased. All these can be taken care of easily by looking at what is permissible either in an E-area or an O-area. Transformation rule 1(a) is applied for both (ii′) and (iv′), and transformation rule 2(b) for both (i′) and (iii′). By paying attention to which kind of an area erasure or drawing needs to take place in, it becomes clear which rule we need to apply.

Also, this example nicely illustrates another important difference between the transformation of Alpha graphs and the transformation of propositional sentences. While the inference from sentence 1 to sentence 2 requires several intermediate lines of proof, graph (1) may be directly transformed to graph (2) without introducing intermediate graphs; one needs only to write and erase symbols in some of the areas in graph (1) itself. The manipulation from the premises to the conclusion is sometimes easier for a user to observe and operate *at once* in the Alpha system than in a symbolic natural deductive system.

I would like to make two general points before moving on to the next section. The work presented in this section may be considered a case study to illustrate the link between the naturalness and the efficacy of Peirce's graphical system. Not surprisingly, the study shows that when the rules of a system are stated more naturally and intuitively, the efficacy of the system increases.

A more interesting lesson we may draw from the current work is that the naturalness and the intuitiveness of rules depend on the type of representation system to which they belong. In the case of a symbolic system, we easily know how the sentences in the premises and the conclusion are built out of their components. There is no ambiguity in the syntactic history of each sentence. According to which derivational operations are used, we choose correct rules of inference. However, in the case of graphs, as shown in §4.4.1, a building history of each graph is not helpful in finding the correct rule. First, each graph might have more than one way of being constructed. Second, how a graph is constructed is visually not clearly presented in the graph. Third, the Alpha system has very few syntactic objects, i.e., only letters, cuts, and juxtaposition. If we strictly applied the principles behind a natural deductive system to the Alpha system, we would have to limit ourselves to rules formulated around only these three objects.

For all these reasons, transformation rules based on basic syntactic objects and building operations are not natural in graphical representation systems. Instead, naturalness in a graphical system must stem from visual intuitiveness. In the case of an Alpha graph, whether something is written in an E-area or in an O-area is not directly related to the building operations of a graph, but it is visually apparent. When we compare a graph given as the premise and a graph we aim to obtain as the conclusion, visual differences are immediately observed and lead us to select the correct rule *if* the rules are written in the natural way presented here. Hence, Peirce's graphical system becomes natural when we take full advantage of the visual features of the system. This is how to make a graphical system more natural.

4.5 Sentences versus graphs

So far, I have challenged the following main criticisms of EG: (i) that the meaning of each graph is not clear, and (ii) that the function of the inference rules is not transparent either. By presenting alternative reading methods and inference rules, I have undermined the conclusion to which these two criticisms have led: As a deductive system, EG is not as useful as first-order symbolic languages. One last challenge I will take up in this section is to explore whether there is any good reason for us to prefer EG's Alpha system to standard propositional languages.

4.5.1 Translation of sentences into graphs

Much work on EG has concentrated on translating from graphs of EG to symbolic languages. This is the correct direction to take for understanding a new system, in this case the Alpha system. However, when this task is done, if we aim to explore more about the new system, we cannot be satisfied with the mere fact that all the units of the Alpha system are understandable in terms of the medium we already have.

In this subsection, I suggest that we reverse the traditional relationship between formulas and graphs: While we have understood graphs in terms of formulas, we will here attempt to understand formulas in terms of graphs. The following algorithm translates sentences into graphs:

The $\overline{\mathscr{K}}$ algorithm for translating sentences into graphs Let *SS* be a set of sentence symbols, *WFF* be a set of sentences of sentential logic and $\mathscr{G}$ be a set of the Alpha graphs. We define function $\mathscr{K}$ as follows:

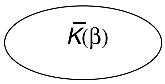
$\mathscr{K} : SS \rightarrow \mathscr{G}, \quad \text{where } \mathscr{K}(A_i) = A_i$

Then, we extend this function to $\bar{\mathscr{K}}$ as follows:

$\bar{\mathscr{K}} : WFF \to \mathscr{G}$, where

K1 If α is a sentence symbol, then $\bar{\mathscr{K}}(\alpha) = \mathscr{K}(\alpha)$.

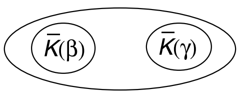
K2 $\bar{\mathscr{K}}((\neg\beta)) =$



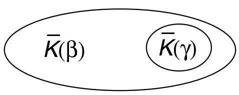
K3 $\bar{\mathscr{K}}((\beta \wedge \gamma)) =$

$\bar{K}(\beta)$ $\bar{K}(\gamma)$

K4 $\bar{\mathscr{K}}((\beta \vee \gamma)) =$

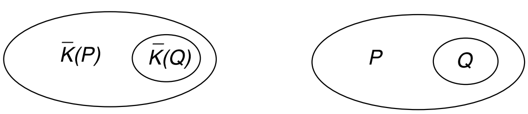


K5 $\bar{\mathscr{K}}((\beta \to \gamma)) =$



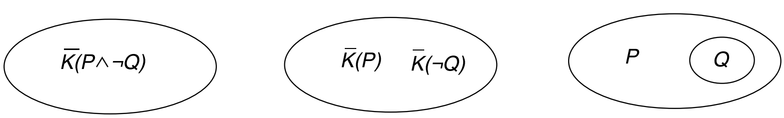
To illustrate how this algorithm works, a few examples are in order.

Example 4.11 Formula $(P \to Q)$ is translated into an Alpha graph in the following process:



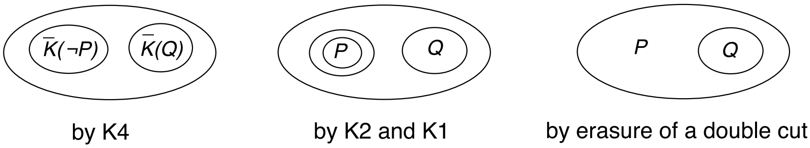
by K5 by K1

Example 4.12 Sentence $\neg(P \wedge \neg Q)$ takes a different process, but winds up translated into the same graph as in example 4.11:



by K2 by K3 by K2 and K1

Example 4.13 Another logically equivalent sentence, $(\neg P \vee Q)$, is translated in the following way:



4.5.2 Applications: logical equivalence and NNF

Let me present two kinds of applications for the algorithm, $\overline{\mathscr{K}}$, plus the double cut rule. One is to find logical equivalence among formulas, and the other is to transform a formula into a negation normal form.

In the examples above, we observed that different formulas can sometimes be translated into one and the same graph,[42] which is consistent with our previous observation that one and the same graph can sometimes be translated into more than one formula. So when multiple syntactically different formulas are translated into one and the same graph, we know that these formulas are logically equivalent to one another. We thus can establish the following simple proposition:

Simple proposition Let α and β be sentences, and let G be the graph of α and G' be the graph of β. That is, $\overline{\mathscr{K}}(\alpha) = G$ and $\overline{\mathscr{K}}(\beta) = G'$. If *either* G and G' are the same graph *or* G and G' are the same except that we may transform one to the other by erasing or drawing a double cut, then α and β are logically equivalent.

There are several uses we can make of this proposition. To test logical equivalence among different sentential formulas, we are accustomed to either setting up a truth table or seeing whether we can deduce one from another and vice versa. We can now add one more method to this list, that is, using Peirce's Alpha system.

Example 4.14 The two sentences $P \rightarrow (Q \rightarrow R)$ and $(P \wedge Q) \rightarrow R$ are logically equivalent, since we get the same graph for each sentence (figure 4.4).

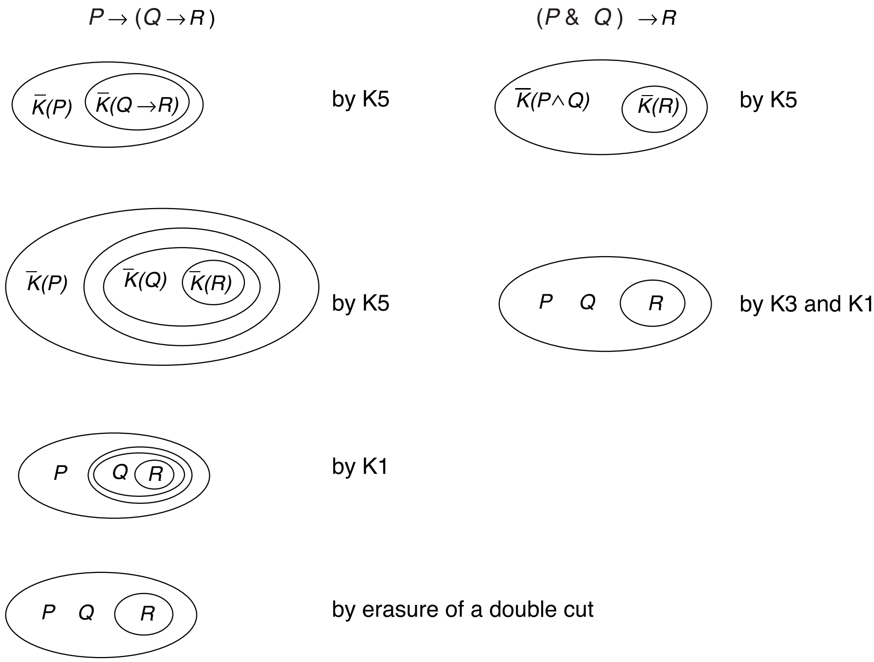
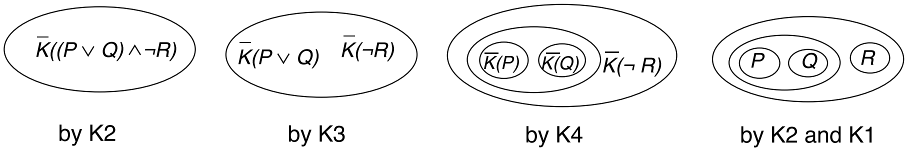


Figure 4.4
Graphs for $P \rightarrow (Q \rightarrow R)$ and $(P \wedge Q) \rightarrow R$.

Of course, our method is not complete since getting the same graph (or the same except for double cuts) is a sufficient but not necessary condition for logical equivalence. We get two different graphs for some pairs of logically equivalent sentences, e.g., $(P \wedge \neg P)$ and $(Q \wedge \neg Q)$.[43]

Combining the simple proposition stated above and the NNF reading method presented in the second section, we can also use Alpha graphs to transform a sentence into a sentence in negation normal form. After translating a given sentence into an Alpha graph by the algorithm $\overline{\mathscr{K}}$, we read the graph by the NNF reading to get a sentence in negation normal form.

Example 4.15 We would like to change the sentence $\neg((P \vee Q) \wedge \neg R)$ into a sentence in negation normal form. By algorithm $\overline{\mathscr{K}}$ we obtain the following graph:



Now we read this Alpha graph according to NNF reading to get the sentence $(\neg P \wedge \neg Q) \vee R$, which is the sentence in negation normal form for $\neg((P \vee Q) \wedge \neg R)$.

4.5.3 Visual efficiency

Importantly, in the Alpha system we *see* logical equivalence. It is more efficient to see that the resulting graphs are the same graph than to find a deduction sequence from one sentence to the other and vice versa. Accordingly, some laws in propositional logic, e.g., DeMorgan's laws and distributive laws, can be presented more efficiently in this graphical system. Also, we obtain a sentence in negation normal form by *reading off* the graph of the given sentence, rather than transforming it by using inference rules. Again, this is more efficient than finding a deduction sequence.

The efficiencies we have discussed so far can be seen as a more general advantage of EG over symbolic systems: EG is efficient in that only two kinds of syntactic devices, i.e., cut and juxtaposition, allow us to express negative, conjunctive, disjunctive, and conditional propositions, while propositional languages adopt different syntactic devices for each connective. The fewer syntactic devices a deductive system has, the fewer inference rules it requires, and accordingly, the simpler the search for a deduction becomes. However, a symbolic system with only two connectives, though complete, is quite inconvenient. By contrast, EG's Alpha system has fewer syntactic devices than propositional languages, but without suffering from the inconvenience of a symbolic system with only two connectives.

5

The Beta System Reconsidered

Extensive research has been undertaken on Peirce's Beta system, which is equivalent to a first-order predicate language. Both Zeman's[1] and Roberts'[2] works are sophisticated treatments of this impressive graphical system. Zeman's definition of the set of Beta graphs[3] and his ingenious method of translating Beta graphs have become a foundation for those who work on EG. And Roberts' classic book on EG has made the system much more accessible to us.

However, as pointed out at the beginning of chapter 4, logicians have doubted that the Beta system, as well as of the Alpha system, have practical use. In the following, I aim to meet this challenge to Peirce's Beta system. To accomplish this, we need to keep three goals in mind: First, as presented in §3.1.2, I treat Peirce's Beta system as an extension of his Alpha system. Hence, the reading method I suggest in this chapter makes a full use of the visual features uncovered and revived for the Alpha system in chapter 4. Second, I pay attention to Peirce's own motivations behind the Beta system, which we have explored in detail in §3.3, so that they may be reflected in the new reading method. Third, I evaluate existing methods carefully with a view to adopting the merits and improving the defects of each method.

In the first section, I examine two different approaches to the Beta system of EG: Zeman's, and Roberts'.[4] In the second section, I offer a new reading algorithm which reflects Peirce's original intuitions for this system. Inference rules are restated in the third section to make the Beta system more efficacious.

5.1 Preliminaries

Two pioneering works have provided methods for reading off Peirce's Beta graphs, one by Zeman[5] and the other by Roberts.[6] It is important to notice that Zeman and Roberts accomplished their goal by taking very different routes, as a result of their different emphases. Zeman aimed to come up with a formal and comprehensive translation algorithm for the system. By contrast, Roberts was interested in presenting Beta graphs at a more informal and intuitive level, which he achieved by a close reading of Peirce's manuscripts. We will see how the difference between these two scholars' intentions is reflected in their proposed reading methods.

Before we examine the existing reading methods, I would like to set up the criteria for a good reading. First of all, as a necessary condition, the reading should yield a correct result. Second, the algorithm should be easy to understand and to use. Third, the reading should respect Peirce's intuitions about his graphical system, discussed in §3.3. Fourth, we want the reading method to contain as few exceptions or special stipulations as possible. At the end of the each subsection, I will evaluate each reading method by applying these criteria.

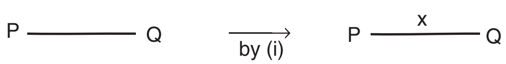
5.1.1 Zeman's reading

It is quite clear that Zeman's method for reading EG is the most comprehensive and systematic one presented so far. Nonetheless, this reading has hardly been put to use. A main reason for this neglect is the impression that Zeman's algorithm introduces a new vocabulary and a rather involved procedure and, as a result, yields a translation that looks more complicated than the original graph. I believe that this superficial understanding of Zeman's reading method has prevented us not only from appreciating Zeman's ambitious project but also from focusing on its merits for a better reading method.

In the following I will reconstruct his entire algorithm to make it clear where each step is heading. My reconstruction of Zeman's method also helps us to understand the motivation behind each step and at the same time to locate the origins of the existing complaints against this reading method. I will show that the root of the problem is the fact that Zeman did not build his method to reflect Peirce's intuitions discussed in §3.3. Furthermore, we will discover why Zeman did not implement Peirce's clear ideas but introduced more vocabulary and obtained a translation that is more difficult to read.

Zeman's basic idea, which is ingenious, is to transform a Beta graph into a quasi-Alpha graph, since an algorithm for Alpha graphs is already available.[7] He notices that a main difference between the Alpha and Beta systems is the use of LIs in the latter, and therefore that transforming a Beta graph into an Alpha graph requires a process to erase all the LIs of a Beta graph in a legitimate way. Erasing LIs involves three steps: (i) Assign a variable to each LI,[8] (ii) write a well-formed formula at the end of each LI, and (iii) erase the LIs. Let's see how these processes take place one by one.

When a simple Beta graph is given, step (i), the assignment of a variable to an LI, works in a straightforward way. For example,



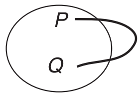
Zeman distinguishes between these simple cases in which a "geodesic" line directly receives a variable and other more complex cases in which "non-geodesic" lines require special treatment. This is Zeman's definition for "geodesic" lines:

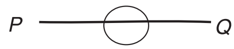
> We shall call a LI "geodesic" between two points on the SA [sheet of assertion, i.e., a paper or a blackboard] iff it connects those points, crossing only the cuts which may be between them, and each of those cuts only once.[9]

That is, if there is no cut that an LI crosses or if an LI crosses any cut only once, then it is geodesic. For example, the LIs in the following graphs are geodesic:

In the third graph, even though the LI crosses two cuts, it crosses each cut only once.

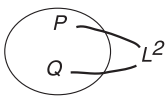
On the other hand, if an LI crosses one and the same cut more than once, the line is non-geodesic. The following are examples of non-geodesic LIs:

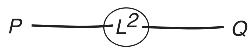




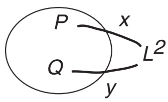
In the case of a non-geodesic line, the part of an LI between the crossings of the cut is called a "loop."[10] In the above example, the loop in the first graph is the part drawn outside the cut, and the loop in the second graph is the part drawn inside the cut. Hence, each non-geodesic line has at least one loop.

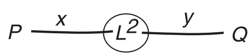
Zeman suggests that each loop be broken into two parts and a temporary predicate, L^2, be inserted at this new joint.[11] The above examples become:



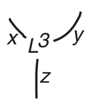


Now all the lines are geodesic, and we assign variables to them, as follows:



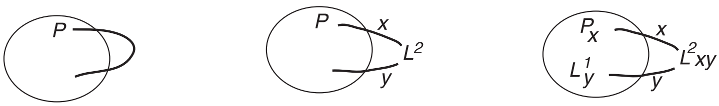


One more special treatment is in order: Where three branches meet, Zeman suggests that we should break this joint as well, use a new predicate, L^3, and assign a variable to each line.[12] For example, the Beta graph on the left side is transformed into the graph on the right side:

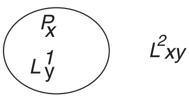


Zeman calls these two newly introduced predicates, L^2 and L^3, "a translation vocabulary."[13] This is the end of process (i) of variable assignment, described above.

The next step is to write a formula by using predicates and the variables assigned in process (i). At each end of an LI, we write a well-formed formula formed out of a predicate written at the end and variables assigned to each line hooked to the predicate. Zeman needs to take care of the special case when no predicate is written at the end of an LI, i.e., a *loose end*. He suggests that we write another temporary predicate, L^1, at a loose end and form a formula with the variable assigned to the line.[14] Let me illustrate these processes with the following example:[15]



Now we are ready for process (iii), the erasure of LIs and temporary variables, which is the final process for transforming a Beta graph into a pseudo-Alpha graph. Let's erase lines from the last graph in the above to get the following:



This graph just looks like an Alpha graph except that sentence letters of an Alpha graph are replaced with first-order formulas. Zeman calls these Alpha-looking graphs X'.[16] One reads off quasi-Alpha graphs similarly to how one reads Alpha graphs (on the endoporeutic reading, according to Zeman): A juxtaposition is translated into conjunction, a cut into negation, and we proceed from outside inwards. So the Alpha-looking graph above is translated into '$\neg(Px \wedge L^1y) \wedge \neg L^2xy$'.

The next step is to transform a formula into a sentence by adding existential quantifiers. For this process, it is crucial to write down an existential quantifier with a variable, i.e., $\exists x_n$, in front of the *shortest* subformula which contains all the occurrences of the variable x_n.[17] In this way we make sure to take care of the scope problem between quantifier and negation, and achieve a correct result. Following this rule, we obtain the sentence '$\neg\exists x\exists y((Px \wedge L^1 y) \wedge \neg L^2xy)$' for the graph above.

However, the sentence obtained by adding existential quantifiers, negation, and conjunction might have some of the temporary vocabulary, i.e., L^1, L^2 and L^3. The final step is to replace L^1x, L^2xy, and L^3xyz with $x = x$, $x = y$, and $(x = y \wedge y = z)$ respectively. Therefore, '$\neg\exists x\exists y((Px \wedge y = y) \wedge \neg x = y)$' is the final translation of our example.

The following is my reconstruction of Zeman's translation, reflecting Zeman's motivations in each step:[18]

1. Transform a Beta graph, say G, into a pseudo-Alpha graph, say G', that is, a graph free of LI's, thus:
   (a) Assign a temporary predicate in the following way:
      (i) At a loose end, write L^1.
      (ii) Break a loop and write L^2 at the broken spot.
      (iii) Break the joint where three branches meet and write L^3 at the broken spot.
   (b) Assign a variable to each single LI.
   (c) Write an atomic wff at the end of an LI: At an end with predicate P where n LIs are hooked[19] with variables $x_1, \ldots,$ and x_n assigned, write $Px_1 \ldots x_n$.
   (d) Erase LIs and the variables assigned to the LIs.
2. Convert G' into a formula in the following way:[20]
   (a) If G' is an empty graph, then it is translated into $x = x$.
   (b) If G' is an atomic wff, say $Px_1 \ldots x_n$, then it is translated into $Px_1 \ldots x_n$.
   (c) If G' is a single cut of graph X and the translation of X is α, then G' is translated into $\neg\alpha$.
   (d) If G' is $G_1 \ldots G_n$ and the translation of G_1 is $\alpha_1, \ldots,$ and the translation of G_n is α_n, then G' is translated into $(\alpha_1 \wedge \cdots \wedge \alpha_n)$.
3. Transform this wff into a first-order sentence: For each variable x in the wff, write $\exists x$ in front of the shortest subformula which contains all the occurrences of x.
4. Remove temporary vocabulary: Replace L^1x with $x = x$, L^2xy with $x = y$, and L^3xyz with $(x = y \wedge y = z)$.

Let us look at one example to see how these clauses work.

Example 5.1 Figure 5.1 shows the process for obtaining the pseudo-Alpha graph G' from the Beta graph G. We then apply the second clause to obtain the following formula:

$\neg(Px \wedge L^3xyz \wedge \neg(Qy \wedge Rz))$

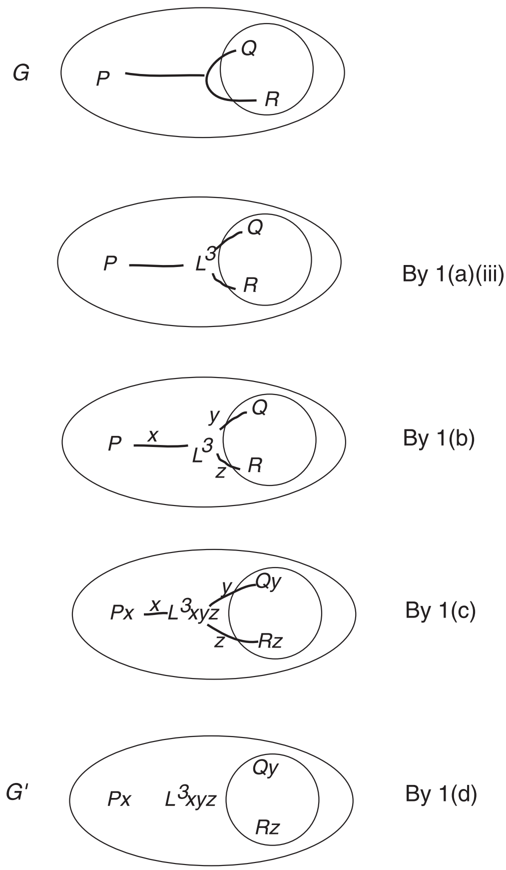


Figure 5.1
Transforming the Beta graph G into the pseudo-Alpha graph G'.

We change this formula into a sentence by adding existential quantifiers in front of the smallest subformulas:

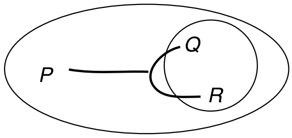
$$\neg\exists x(Px \wedge \exists y\exists z(L^3xyz \wedge \neg(Qy \wedge Rz)))$$

After replacing the temporary predicate L^3, we get the following sentence as the final translation of the Beta graph G by Zeman's reading:

$$\neg\exists x(Px \wedge \exists y\exists z(x = y \wedge y = z \wedge \neg(Qy \wedge Rz)))$$

This quite complicated-looking formula is logically equivalent to '$\forall x(Px \rightarrow (Qx \wedge Rx))$'.

The analysis of this example serves as a starting point for inquiring why, in most cases, Zeman's reading is bound to yield a complicated-looking translation. For the given Beta graph G in the above example, i.e.,



(1)

let's compare Zeman's translation of G, i.e.,

$$\neg\exists x(Px \wedge \exists y \exists z(x = y \wedge y = z \wedge \neg(Qy \wedge Rz))) \tag{2}$$

and the following sentence, which is logically equivalent to (2).

$$\forall x(Px \rightarrow (Qx \wedge Rx)) \tag{3}$$

The question we are interested in is which formula, (2) or (3), looks closer to graph G. Even though this is not a well-clarified question unless the 'looking closer' relation between a graph and a formula is defined, we can try to answer it partially, but intuitively, on the basis of two striking differences between (2) and (3). One is the number of variables used in each sentence, and the other is the kinds of quantifiers used in each sentence. Importantly, these two issues are directly related to the first two of Peirce's intuitions presented in §3.3, one about the use of lines and the other about the distinction between universal and existential quantifiers. Let us examine these two matters more carefully.

For visual clarity Peirce adopts a line (which is a graphical object) instead of a variable (which is a symbolic object) to represent numerical identity among individuals. The connection of the endpoints by one line visually represents the identify of the individuals denoted by those endpoints. Also, one line with branches is more visually perspicuous than several tokens of one type of variable. There is almost no danger of missing the continuity of a line, unlike with the scattered tokens of the same variable type. For example, the line in graph G above has one connected line with three branches, while the first-order translation (3) uses three tokens of variable x. Accordingly, when graph G is translated into sentence (3), this visual clarity is reduced. Even worse, when G is translated into sentence (2), where three different types of variables, x, *y*, and *z*, are used, neither identity nor continuity is directly reflected in the translation. Clearly, Zeman's reading fails to capture Peirce's motivation for the use of lines.

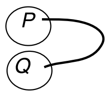
In §3.3, I reviewed another novel aspect of the Beta system. Peirce succeeds in representing *every* individual and *some* individual without using any new additional syntactic object. Universal and existential quantifiers are represented by the visual fact of whether the outermost part of a line is in an E-area or in an O-area. When that idea is followed, since the outermost part of the line in *G* is in an O-area, the graph *G* should be read off as a universal sentence, just as in sentence (3). Unfortunately, this novelty is not directly reflected in sentence (2). Instead, a universal sentence is represented there only indirectly, i.e., as a negation of an existential sentence. When it is noticed that Zeman's reading introduces only existential quantifiers,[21] it is clear that this reading does not adopt Peirce's visual distinction for quantifiers at all.

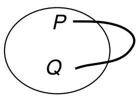
The failure to follow Peirce's two intuitions results in sentences with more variables than needed and with nested negations and existential quantifiers. The more variables and the more nested negations, the more difficult a sentence is to read. Nowhere in Zeman's writing does he explicitly address why he decided not to implement some of Peirce's ideas. For example, Zeman is quite aware of Peirce's representing different kinds of quantifiers by using visual features. After citing Peirce's 4.458 (the passage cited in §3.3.2),[22] Zeman positively evaluates Peirce's visual representation of existential versus universal quantifiers:

> The type of quantification applying to a given line of identify, then, is determined by examining the line and noting how many cuts enclose the least-enclosed part of the line. The very interesting feature here is that no explicit sign for quantification—that is, no quantifier—is required to "get quantification" in Beta.[23]

Curiously enough, however, in his own algorithm, Zeman does not implement different visual features for the two different kinds of quantifiers at all, and there is no explanation for this decision. Below I attempt to provide this part missing from Zeman's work, to reconstruct Zeman's rationale behind his reading method, and to evaluate whether it was worthwhile to ignore the visual distinction for quantifiers that Peirce explicitly set up.

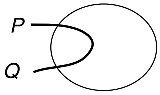
The first problem, i.e., the mismatch between the number of lines in a graph and the number of the variables in its translation, comes from the fact that Zeman's reading requires that one line be broken into two or three lines, depending on whether it has a loop or three branches. By breaking a line, the algorithm loses the link with Peirce's intuition about the use of a line, i.e., that the continuity of a line represents identity. For example, in the following, the first graph gets only one variable, but the second one gets two, since the loop will be broken into two lines:





However, both graphs can be translated into an expression with only one variable: '$\exists x(\neg Px \wedge \neg Qx)$' for the first one and '$\exists x \neg(Px \wedge Qx)$' for the second. As seen at the beginning of this subsection, Zeman treats the line with a loop as a special case, calling it "non-geodesic."

Why does Zeman believe that this non-geodesic line is special and should be broken into two parts? The non-geodesic line in the graph on the right side in the example above does not need to be translated into a sentence with two different variables. On the other hand, Zeman is correct that *some* non-geodesic lines require this kind of special treatment. For example, the line in the following has a loop:



This graph means '$\exists x \exists y(x \neq y \wedge Px \wedge Qy)$', which cannot be expressed with one variable, since the existence of two distinct objects is part of what the graph expresses. If we assign one variable to one line, there is no way to represent two distinct objects, since this graph has only one line.

Therefore, Zeman's intuition that we need a special treatment for non-geodesic LIs is partly correct: Some non-geodesic cases have to be treated differently from geodesic cases, but not all of them have to be. Zeman could have made a more fine-grained distinction among non-geodesic lines. Or instead of a distinction between geodesic and non-geodesic, Zeman could have adopted a different distinction, that between a line to which we may assign one variable and a line to which more than one variable is assigned. His geodesic and non-geodesic distinction was not the most useful one that could be drawn here. This problem will be remedied in the new reading I will suggest in the next section.

The second noticeable problem with Zeman's reading is that a translation usually turns out to be a sentence with nested negations and existential quantifiers. No universal statement is obtained directly. As seen above, Zeman knew that Peirce's system directly represents *everything* by drawing the outermost part of an LI in an O-area. Why, then, didn't Zeman write down a clause for this representation? Two explanations are plausible.

One possible explanation is that Zeman's algorithm requires that LIs be erased in the middle of the reading process. After lines have been erased, it is not easy to use the visual feature of whether the least enclosed part of a line is in an E-area or an O-area. Hence, Zeman chose to obliterate the clear visual representation of universal quantification in the system.

However, there is a more fundamental reason than LIs' being erased. Even if Zeman left lines until the end, he would not have used the visual difference which indicates either *every* or *some* individual, because his algorithm adopts the endoporeutic reading method. After converting a Beta graph into an Alpha-looking graph, Zeman reads off the pseudo-Alpha graph by Peirce's *endoporeutic* reading.[24] That is, when a graph has cuts, the interpretation "proceeds inwardly." Thus, Zeman's failure to read off a clear visual feature of LIs as either a universal or existential quantifier is directly related to the defect of the traditional reading of the Alpha system examined in the previous chapter.

For example, the first graph in the following is interpreted to be the negation of the second graph:

Hence, after we interpret the second one to be '$\exists x(\neg Px \wedge \neg Qx)$', we write a negation in front of this sentence to get the interpretation of the first graph, '$\neg\exists x(\neg Px \wedge \neg Qx)$'. Notice that for the interpretation of the second graph, we use Peirce's idea that an identity line with its outermost part being evenly enclosed represents *some* individual. However, for the interpretation of the first graph, there is no occasion in Zeman's scheme to read off the visual fact that the least enclosed part of the line is in an O-area. Instead, the condition 'the shortest subformula' in the third clause of Zeman's algorithm becomes crucial.[25] If we ignored this condition, we would get a wrong translation '$\exists x\neg(\neg Px \wedge \neg Qx)$', while '$\neg\exists x(\neg Px \wedge \neg Qx)$' is a correct one. Hence, according to the endoporeutic reading built into Zeman's algorithm, we never get the reading '$\forall x(Px \vee Qx)$' directly from the first graph, but only after using DeMorgan's laws on a sentence with nested negations and existential quantifiers.

This problem is not unique to Zeman's reading, since Peirce's *endoporeutic* reading for Alpha graphs has been accepted by all the literature on this subject. As long as a cut of a graph is interpreted as the negation of the interpretation of the graph, Peirce's insight about the representation of existential versus universal statements has no room to be applied. While Peirce's insight leads one to a direct reading of *every* or *some*, the *endoporeutic* reading leads one to read off quantification indirectly. Therefore, Peirce's discussion of direct representation of different quantifiers can be easily implemented in my direct reading method presented in §4.2 (one which uses two kinds of juxtaposition) or my multiple reading method in §4.3 (one which uses scrolls in addition to two juxtapositions) but not in the traditional indirect readings.

Now let us evaluate Zeman's reading method against the four criteria we set up at the beginning of the section. This reading clearly passes the necessary condition for a reading, i.e., that it get correct results. I suspect the fourth criterion, the presentation of a comprehensive method without exception rules, is, in fact, the main goal of Zeman's method, and he is successful in that project. Every Beta graph may be translated into a first-order sentence in a uniform way. On the other hand, the efficiency of a reading method, the preservation of a similar structure between a given graph and its translation, and faithfulness to Peirce's main ideas were secondary in Zeman's project. Even though it can be a subjective matter to tell how easy a given reading method is, the general consensus is that Zeman's method is not easy enough to be used popularly. Above all, as I explained above in detail, Zeman's method clearly fails the third criterion, the implementation of Peirce's own motivation for Beta graphs. As a result, this reading method does not take full advantage of the iconic aspects of EG.

5.1.2 Roberts' reading

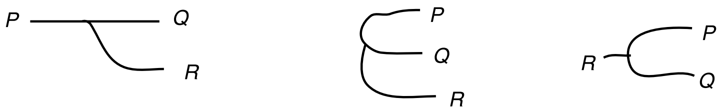
Don Roberts' important contribution to this subject has caused Peirce's EG to be more widely known than it otherwise would be. In this subsection I will focus on Roberts' work about how to read off Beta graphs of EG.

Roberts' approach is quite different from Zeman's. While Zeman is mainly interested in a formalized and generalized way of presenting the Beta system, Roberts aims to introduce this system in a more intuitive and informal way. Roberts explains informally what LIs are supposed to mean in the system and how quantification is expressed on the basis of Peirce's own conventions. With these informal explanations, Roberts shows us how to read a Beta graph.

A dot on a sheet of paper can be spread into a line. Roberts says "EG, thus, combines in one symbol [dot or line] the sign of individuality and the sign of quantification."[26] The following three of Peirce's conventions address how this one syntactic device performs what two syntactic devices of a first-order language, quantifiers and variables, do:

C6 The scribing of a heavy dot or unattached line on the sheet of assertion denotes the existence of a single, individual (but otherwise undesignated) object in the universe of discourse.
C7 A heavy line, called a line of identity, shall be a graph asserting the numerical identity of the individuals denoted by its two extremities.
C8 A branching line of identity with n number of branches will be used to express the identity of the n individuals denoted by its n extremities.[27]

One heavy dot is one of the smallest units of the system and cannot be analyzed further. It states the existence of an object, but no property is asserted about the object. Strictly speaking, there is no syntactic unit corresponding to a non-sentence formula of first-order languages.[28] This graphical object, the dot, has a great advantage in representing individuals. By spreading it into a heavy line and also into a line with many branches, without losing numerical identity, this line can be connected to more than one predicate, according to C7 and C8. Even though it sounds complicated verbally, when we see a line, it is visually clear that one and the same individual with three kinds of properties is mentioned in the following:



Roberts tells us to read off LIs in this most natural way. There is no breaking point for a branch, as Zeman's reading would have it. If Roberts had attempted to use variables, he would have said that one type of variable is given to one network of LIs. There would be no process corresponding to the three clauses under 1(a) of Zeman's algorithm. Roberts would not divide one network of LIs or bring in temporary vocabulary.

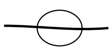
Now it is time to recall why Zeman adopted the clauses in 1(a). Zeman thought that if one variable is assigned to one network of LIs, a line with a loop would lead to the wrong result. However, as we saw at the end of the previous subsection, a line with a loop does not always get an incorrect result, only *sometimes*. Therefore, we concluded that Zeman went too far to include all those cases and ended up making many simple cases more complicated. In the case of Roberts' reading, we should ask the opposite question: Does Roberts' simple way of reading off identity lines take care of the genuine case of incorrect results that Zeman was rightly worried about?

To an answer to this question, let me bring back two of the graphs discussed before:

According to Zeman, both cases have a loop, and hence we will have two variables in each translation. In the first graph, Roberts would emphasize that this single line—which might be considered to have three parts, one outside the cut, the other two inside the cut—should denote the same individual. Therefore, we get a correct translation 'Something is not both *P* and *Q*.' This translation, '$\exists x \neg(Px \wedge Qx)$', looks much simpler than Zeman's logically equivalent translation for the same graph, '$\exists x \exists y(x = y \wedge \neg(Px \wedge Qy))$'.

However, the idea that one network of LIs (Peirce calls it a "ligature")[29] denotes the same individual does not work in the graph on the right side. Even though all three parts (two outside the cut and one inside the cut) are connected to look like a single line, this graph is referring to two different individuals. What it represents is the proposition 'Something is *P* and something is *Q* and these two objects are distinct from each other.' Zeman's reading gives us a correct result, '$\exists x \exists y(Px \wedge Qy \wedge \neg x = y)$'.

In his original discussion, Roberts starts a new paragraph with "One case involving a line crossing a cut remains to be discussed, the case in which the line of identity passes entirely through an empty cut [the following graph]:"[30]



Here is Roberts' intuitive and informal interpretation for this case:

> This device signifies the non-identity of the individuals denoted by the extremities of the ligature: 'There are two objects such that no third object is identical to both.'[31]

However, it is not so clear to me how he got the interpretation 'There are two objects and they are not identical with each other.'[32]

In his later work, Roberts, in a separate subsection entitled "Special Cases,"[33] presents several graphs as exceptions to the simple denotation relation between a single line and a single individual, and stipulates special readings for each of them. The graph in the earlier quoted passage is one of them, and this time, he gives a more detailed interpretation:

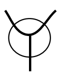
> Literally, this ["the graph in which a line passes entirely through an empty cut"] means "There is an object and there is an object and it is false that these objects"—denoted by the two unenclosed parts of the line—"are identical."[34]

Peirce himself treated this case in one of interpretational corollaries:

Interpretational Corollary 7 A line of identity traversing a sep [cut] will signify non-identity.[35]

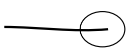
By using the word ‘corollary’, Peirce implies that the reading for this case follows from his conventions. However, since it is not so clear how to deduce this corollary from Peirce’s conventions, Roberts felt a special stipulation to be necessary. This is Roberts’ way of avoiding Zeman’s temporary predicate L^2.

How about the temporary predicate L^3? We have seen the case where Zeman’s L^3 introduction of three variables results in a complicated translation, while the intuitive idea about an LI would introduce just one variable.[36] Zeman’s awkward-looking choice comes from a correct realization that it is not always the case that we can introduce one variable for a network of LIs. Suppose that the joint part of three branches is enclosed by a cut:

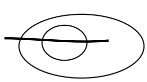


Again, Peirce’s conventions, and likewise Roberts’ reading, which relies on Peirce’s conventions, do not guide us clearly as to how to interpret this graph. Both readings give us the rather ambiguous English sentence ‘Three individuals are not all identical.’[37]

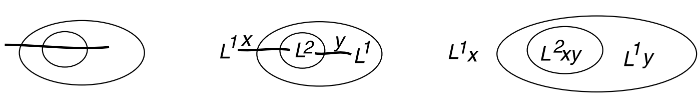
How does Roberts avoid having another of Zeman’s temporary predicates, L^1? I do not see how Roberts’ informal reading guides us in how to read the following simple-looking graph, for which Zeman would introduce L^1:



Instead, Roberts discusses the following graph as one of the special cases:



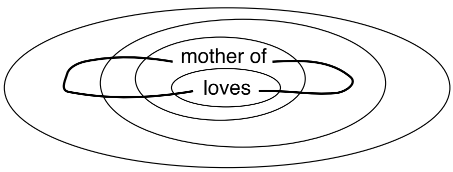
His interpretation is “There is something, and it is false that there is something else non-identical to it,” that is, “Something is identical with everything.”[38] It is not clear how this reading is obtained. On the other hand, Zeman’s reading will go through the following processes:



So we obtain the reading, '$\exists x[x = x \wedge \neg\exists y(y = y \wedge \neg(x = y))]$', which is logically equivalent to '$\exists x \forall y(x = y)$'.

Thus, Roberts' reading implements Peirce's main idea behind the use of identity lines to get a simple and intuitive reading, but this intuitive reading does not seem to cover all the cases. Therefore, some graphs need to be treated as special cases. The method can hardly be an ideal algorithm if we need stipulations whenever we run into a graph whose reading the intuitive method fails to obtain. First of all, there is no way to tell whether all exceptional cases are covered under the several different stipulations. Also, it is not easy to confirm whether a reading obtained as a special case is correct or not when we do not have a comprehensive formal algorithm.

There is a good example to illustrate this problem. Roberts himself presents the following graph for the proposition 'Every mother loves some child of hers':[39]



However, the graph expresses the sentence '$\forall x \exists y(\text{MotherOf}(x, y) \rightarrow \text{Loves}(x, y))$', which is not equivalent to the English sentence 'Every mother loves some child of hers.'[40] Roberts himself reads off this graph very simply, but unfortunately incorrectly:

> Let *i* be the individual denoted by the line of identity at the left (the mother), and *j* be the individual denoted by the line of identity at the right (the child), and read the graph as follows: 'Take any individual you please, say *i*, there is an individual *j*, such that if *i* is a mother of *j*, then *i* loves *j*'.[41]

This error is understandable, since Roberts does not have a complete algorithm against which to check his reading for this graph.

On the other hand, on Zeman's reading method, after the following quite complicated processes in figure 5.2, we get the reading $\neg\exists x \exists y \cdot [x = y \wedge \neg\exists u \exists v(u = v \wedge \neg(\text{MotherOf}(x, u) \wedge \neg\text{Loves}(y, v)))]$, which is logically equivalent to $\forall x \exists y(\text{MotherOf}(x, y) \rightarrow \text{Loves}(x, y))$. Then, we would know that this graph is not a correct translation for the English sentence 'Every mother loves some child of hers.'

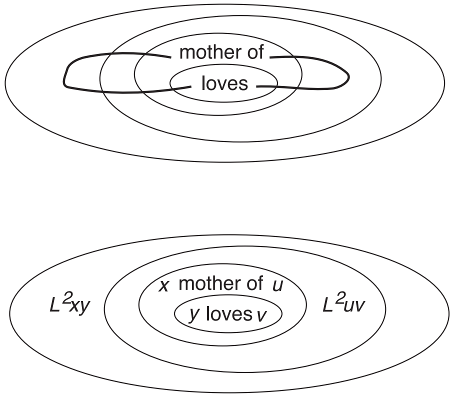
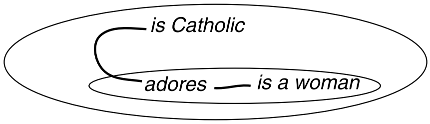


Figure 5.2
The transformation of the graph that Roberts suggests for 'Every mother loves some child of hers' on Zeman's reading method.

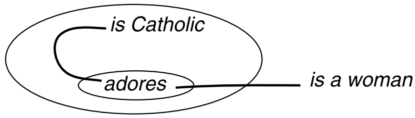
Roberts has been faithful to Peirce's intuition for identity lines, which leads him to two related kinds of trouble. One is that the method needs to stipulate exceptional cases, and the other, which is even more serious, is that the lack of a systematic reading method makes an incorrect result hard to detect.

When we move on to Peirce's visual distinction between universal and existential quantifiers, we find Roberts to be quite ambivalent towards implementing it. Interestingly enough, Roberts is fully aware of Peirce's idea about this distinction and discusses it in detail,[42] and sometimes Roberts implements the distinction in reading off a graph.[43] In general, however, Peirce's visual distinction between two kinds of quantifiers is hardly used in Roberts' reading method. Instead, as on Zeman's reading method, a line whose outermost part is enclosed in an O-area is read as a negation of an existential quantifier. Again, as with Zeman's method, we obtain a nest of negations and existential quantifiers.

Roberts' ambivalent attitude toward a visual distinction between two kinds of quantifiers is illustrated very well in his reading of the following two graphs. This example will also show us how a reading can become cumbersome when it relies on unclarified intuitions rather than specific guidelines.[44]



(1)



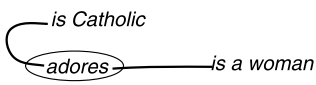
(2)

Roberts uses Peirce's visual distinction between the two kinds of quantifiers for translating graph (1):

According to C7,[45] this [having two disconnected lines of identity] means that Fig. 10 [graph (1)] refers to two individuals. And according to the method of interpretation discussed above, the line whose outer extremity is once (and therefore oddly) enclosed refers to every individual described as catholic; and the line whose outer extremity is twice (and therefore evenly) enclosed refers to some individual described as an adored woman. . . . Fig. 10 [graph (1)] can therefore be read as follows: 'Every catholic adores some woman.'[46]

However, when he reads off graph (2), this visual distinction does not play any role at all:

What about Fig. 11 [graph (2)]? . . . Consider first the graph which would result from Fig. 11 [graph (2)] if the outer cut were removed:



This means 'There are two individuals; one is a catholic and the other is a woman; and it is false that the catholic adores that woman'. In other words, 'There is some catholic who does not adore some particular woman'. Now if we restore a cut to obtain graph (2) again, and begin our interpretations (endoporeutically) at the outermost part of the graph, it reads as follows: 'There is some woman *and* it is false [that is the force of the restored outer cut] that some catholic does not adore this woman'. The second part of this reading, 'It is false that some catholic does not adore this woman', is a denial of an O proposition, but this is equivalent to the assertion of the A proposition 'Every catholic adores this woman'. Therefore graph (2) may be read 'There is a woman whom every catholic adores'.[47]

There is an inconsistency between Roberts' readings of graph (1) and graph (2) in these quotations. In graph (1), Roberts utilizes Peirce's distinction between universal and existential quantifiers and reads off a scroll directly into a conditional form, while he ignores both of these principles for reading graph (2). Instead, he first reads off a graph without the cut of graph (2) and then negates that interpretation. An interesting question is why Roberts gave up a seemingly better method (adopted for the reading of graph (1)) when he was reading graph (2). We can rather easily read off graph (2) in a way similar to that for graph (1). The line whose outer extremity is not enclosed by any cut (and therefore evenly enclosed) refers to some woman. And the line whose outer extremity is once (and therefore oddly) enclosed refers to every Catholic. By Peirce's ordering of lines, we should first read the line whose outermost part is enclosed by no cut. Therefore, we read 'There is some woman whom every Catholic adores.' I suspect that Roberts was not sure whether we can read off a scroll from graph (2), since the line which connects 'adores' and 'is a woman' does not fit in scroll form. On the other hand, Roberts perceived a scroll in graph (1). I claim that the lack of a systematic algorithm in Roberts' reading method explains his two different ways of handling graph (1) and graph (2).

Let me sum up the evaluation of Roberts' reading method against the four criteria I set up at the beginning of the section. Clearly, Roberts aims to satisfy the second criterion, ease of use. And he also tries to implement Peirce's intuitions for this graphical system. Hence, Roberts' informal way to read off graphs seems to be more intuitive and simpler than Zeman's. However, the price for this intuitive and simple reading is that when a graph becomes complicated, it is not clear how to use the informal reading method. This is why Roberts' method has to accommodate more stipulations or exceptions. At the same time, like Zeman's, Roberts' reading does not always implement Peirce's intuitions.

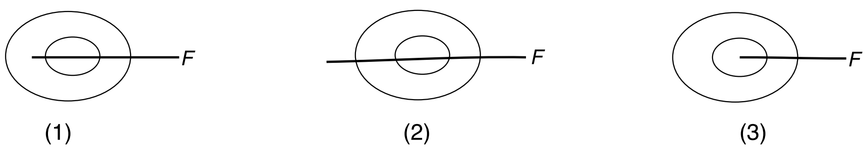
5.2 A new reading

In this section I present a new algorithm for reading Beta graphs that maintains the merits of the existing methods but addresses their defects. Zeman's reading is successful in providing us with a systematic way of reading off Beta graphs, and Roberts' reading seems to be more intuitive. Interestingly enough, but not surprisingly, the merit of one method seems to be the defect of the other. At the same time, neither of them implements Peirce's original intuitions for the Beta system. I attempt to draw on all these lessons in my new reading method: It will be a systematic and comprehensive algorithm to translate Beta graphs into first-order sentences, but at the same time it will remain intuitive. Most importantly, the intuitiveness of this new reading is obtained by being faithful to Peirce's basic ideas discussed in §3.3. Therefore, this reading will be free from the defect that the existing methods suffer from, i.e., ignoring Peirce's valuable insights on EG.

For further discussion, let us define a *double cut* and an *LI network* more carefully. In the case of Beta graphs, double cuts need to be defined in a more complicated way than in the case of Alpha graphs. Roberts was cautious to define a double cut for the Beta system in the following way:

> A special and important class of scrolls consists of those whose outer area *either* is blank *or* contains only portions of lines of identity which pass from inside the inner area to outside the outer area. Any such scroll is called a *double cut* (DC).[48]

For our purposes, let me modify this definition slightly without changing the content. In the Alpha system, there is no syntactic device which occupies more than one cut. Each token of a sentence symbol, if any, is written inside or outside a cut, if any. However, in the Beta system, an LI can cross a cut partially or entirely. In the following example, graph (1) is a case in which an LI crosses the inner cut entirely but the outer cut only partially, graph (2) is one in which an LI crosses both cuts entirely, and in (3) the LI crosses both cuts only partially:

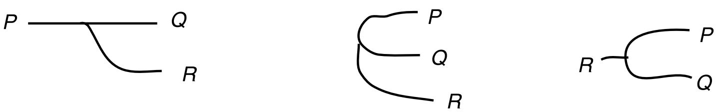


What is written in the outer area of a scroll determines whether the scroll is a double cut. If nothing is written in the outer area of a scroll, then it is a double cut, just as in Alpha graphs. Also, if a portion of an LI crosses the outer area of a scroll *entirely*, then a scroll is a double cut. Hence, in the above, there is no double cut in graph (1), but each scroll in (2) and (3) is a double cut.

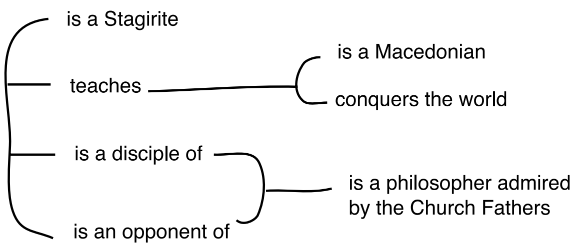
Definition 5.1 A scroll is a *double cut* if and only if nothing is written in the outer area of the scroll [i.e., the area between the inner and the outer cuts] except the portions of LIs which cross the outer area entirely.

Now we examine more carefully how identity lines are used in the Beta system. In the following two graphs, a simple identity line says that something is *P* and *Q* (*or* Some *P* is *Q*):

Suppose we need to express that something is *P*, *Q*, and *R*. In the case of first-order languages, the use of a variable and the scope of a quantifier enable us to express one more property about an object which is *P* and *Q*. In such cases for graphs, an LI needs to branch, as follows:



When an object instantiates more than two properties or relations, an LI branches. Thus, every branch of one and the same LI functions as one identity line; every branch denotes the same object. We call this connected web of branches an *LI network*. It should be noticed that a line of identity is the simplest form of LI network. In the graphs above, each graph has one LI network, just as a first-order language requires *one* type of variable under the scope of one existential quantifier. The following more complex graph comes from Roberts:[49]

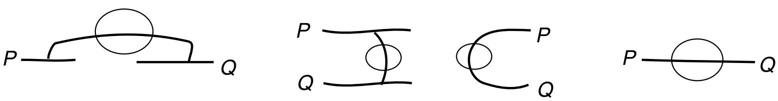


This graph, which expresses the proposition 'There is a Stagirite who teaches a Macedonian conqueror of the world and who is at once a disciple and an opponent of a philosopher admired by Fathers of the Church',[50] has three LI networks, and similarly, three bound variables will be adopted if a first-order language is used.

There seems to be a strong connection between an LI in Beta graphs and a variable in first-order languages, and we will implement this connection in our reading method. However, one main difference between LIs and variables is that an LI denotes an object, while a variable does not. Also, as we will see in the following, it is not always the case that one and the same LI network denotes the same object.

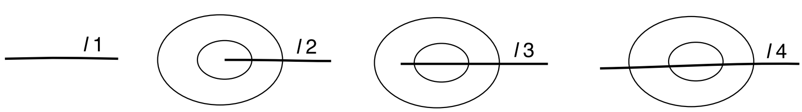
Suppose that we translate the proposition that there are a *P* and a *Q* and these two are not identical with each other. The sentence '$\exists x \exists y(Px \wedge Qy \wedge x \neq y)$' is a translation in a first-order language. Two variables are used under the scope of two existential quantifiers. However, if we use two LIs attached to predicates *P* and *Q* as follows, we do not obtain what we want:

What this graph says is that something is *P* and something is *Q*, i.e., $\exists xPx \wedge \exists xQx$. That is, this graph fails to express the relation of non-identity between the two objects. We need to *negate* the identity relation without negating property *P* or *Q*. Any of the following graphs does the job:[51]

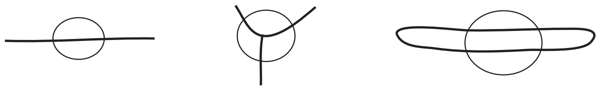


Therefore, when an LI crosses an odd number of cuts entirely, the generalization that each LI network refers to the same object is not true. On the contrary, we need to bring in (at least) two different objects.

If a portion of an LI crosses an odd number of cuts entirely, we say that line is *clipped*.[52] In the following, only line l_3 is clipped. Line l_1 crosses no cut, l_2 does not cross any cut entirely, and l_4 crosses an even number of cuts entirely.



When a line is clipped, we count the number of the parts (say *n*) which lie outside the odd number of cuts and say that there are *n clippings* in the LI network. Each of the following four graphs has only one LI network, but the LI network in graph (1) is clipped into two parts, in graph (2) into three parts, and in graph (3) into two parts, by one cut.[53]



(1) (2) (3)

In order to take care of clippings, the reading method presented below breaks the variable-assigning clause, clause 2, into two cases.

New reading algorithm

1. Erase a double cut.
2. Assign variables to LI networks:
   (a) If no branch of an LI network crosses an odd number of cuts entirely, then assign a new variable to the *outermost* part (i.e., its least enclosed part) of the LI network.
   (b) If a branch of an LI network crosses an odd number of cuts entirely (i.e., an odd number of cuts clip an LI into more than one part), then
      (i) assign a different type of variable to the *outermost* part of each clipping of the network, and
      (ii) at each joint of branches inside the innermost cut of these odd number of cuts, write $v_i = v_j$, where v_i and v_j are assigned to each clipping (in the above process).
3. Write atomic wffs for the LIs:
   (a) For each end of an LI with a predicate, say P, replace P with $Pv_1 \dots v_n$, respecting the order of hooks,[54] where $v_1, \dots,$ and v_n are assigned to the lines hooked to P.
   (b) For each loose end of an LI, i.e., an end without a predicate, write $\top$.
   (c) For an LI which does not get any atomic wff or $\top$, i.e., an unclipped cycle, write $\top$ at the outermost part and at the innermost part of the LI.[55]
4. Obtain complex wffs: We treat the atomic wffs or $\top$ (obtained by the previous steps) just like sentence symbols in the Alpha system. We apply the NNF reading of the Alpha system,[56] substituting sentence symbols for atomic wffs, to obtain a complex wff. (For this step, we ignore LIs and the variables assigned to LIs.[57]) We treat them

like Zeman's quasi-Alpha graphs.[58] Let G be a simple quasi-Alpha graph.[59] The following f is a basic function which translates a simple quasi-Alpha graph to a simple sentence:

$f(G) = \alpha$ if G is atomic wff α

$f(G) = \neg\alpha$ if G is a single cut of atomic wff α

$f(G) = \top$ if G is $\top$

$f(G) = \neg\top$ if G is a single cut of $\top$

$f(G) = \top$ if G is an empty space

$f(G) = \neg\top$ if G is an empty cut

Now we extend this function f to $\bar{f}$ to translate a quasi-Alpha graph into a complex wff:

(a) $\bar{f}(G) = f(G)$ if G is a simple quasi-Alpha graph

(b) $\bar{f}([[G]]) = \bar{f}(G)$[60]

(c) $\bar{f}(G_1 \dots G_n) = \bar{f}(G_1) \wedge \cdots \wedge \bar{f}(G_n)$

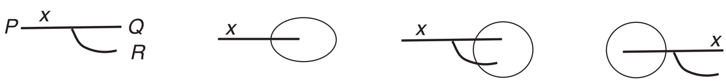
(d) $\bar{f}([G_1 \dots G_n]) = \bar{f}([G_1]) \vee \cdots \vee \bar{f}([G_n])$

5. Obtain sentences: For each variable v_i in the wff obtained by the previous step,
   (a) if v_i is written in an E-area,[61] then add $\exists v_i$ immediately in front of the smallest NNF subformula containing all occurrences of v_i,[62]
   (b) if v_i is written in an O-area, then add $\forall v_i$ immediately in front of the smallest NNF subformula containing all occurrences of v_i,[63] and
   (c) if more than one quantifier, say $\mathcal{Q}_1 v_1, \dots,$ and $\mathcal{Q}_n v_n$,[64] are added in front of the same NNF subformula (by clauses 5(a) or/and 5(b)), then the less enclosed v_i is, the bigger scope $\mathcal{Q}_i v_i$ gets.

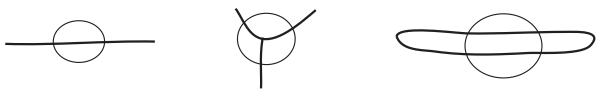
I will illustrate how each clause works through examples. Clause 1 tells us to change the graph on the left to the graph on the right in the following two cases:

However, neither of the following two graphs has a double cut to be erased:

Clause 2(a) respects Peirce's intuition about identity lines. That is, one identity line, however it may branch, visually represents a single object. Clause 2(a) is easily seen to result in the following:

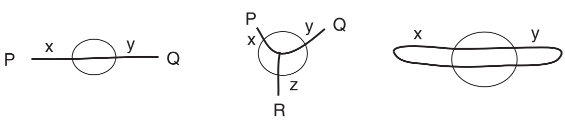


But, there is an exception for which this intuition breaks down. This is what clause 2(b)(i) is about.[65] As seen above, there are two clippings in the LI network in graph (1) below, and there are three clippings in graph (2), and two clippings in graph (3):

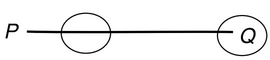


(1) (2) (3)

Clause 2(b)(i) suggests that a different variable be assigned to each clipping:

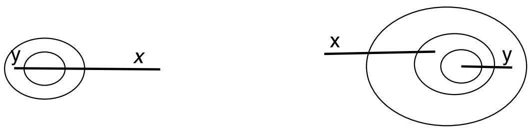


As with clause 2(a), it is important for clause 2(b) to assign a variable to the least enclosed part, since the location of each variable will determine which quantifier should be assigned. Suppose the following graph is given:

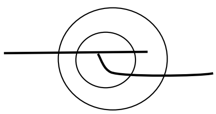


The first cut produces two clippings.[66] Two variables, x and y, should be assigned to the least enclosed part of each clipping. Hence, in the following, the graph on the left is correct, while the graph on the right is not:

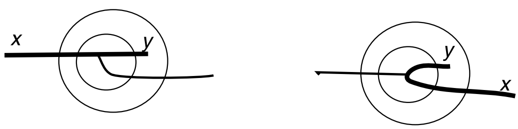
When applying clause 2, it is crucial to pay attention to whether a branch of an LI network crosses an *odd* number of cuts entirely. In the left-hand graph below, the LI crosses one cut entirely, while in the right-hand graph neither of the LIs crosses any cut entirely. Therefore, clause 2 lets us assign variables in the following way:[67]



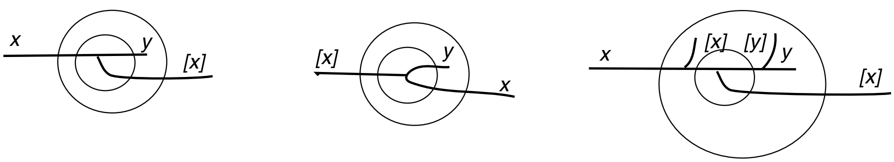
Now let's consider a somewhat complicated case as follows:



Notice that there is no double cut here, since a branch of the LI network ends in the outer area of the scroll. Importantly, a branch of this LI network crosses one cut entirely. We can identify this branch in either way, and assign variables according to clause 2(b):

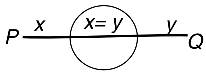


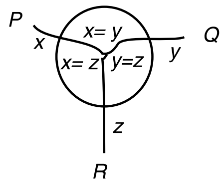
To keep track of one and the same unclipped LI network in complicated graphs, we may write down variables in brackets. For example,



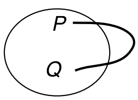
The reason for using brackets is to apply clause 5 more efficiently.[68] Notice that clause 2(b) has one more process, that is, to write down

$v_i = v_j$ inside the cut. This identity statement corresponds to the visual fact that these two clippings are connected with each other inside a cut. The following are the result of this clause being applied:

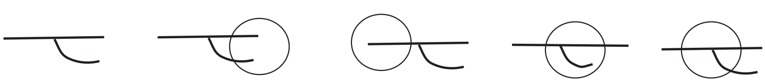




Now we can ask whether the distinction between 2(a) and 2(b) is the same as Zeman's distinction between geodesic and non-geodesic lines.[69] A difference between Zeman's distinction and mine becomes clear when variables are assigned to the following graph:

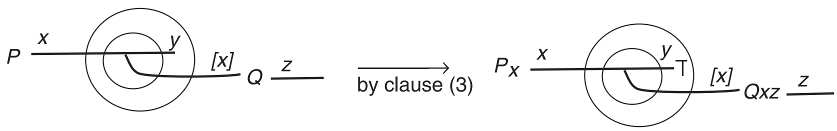


According to my clause 2(a), since no portion of this line crosses an odd number of cuts entirely, only one variable will be assigned to this one connected LI. However, this line crosses one and the same cut more than once and is, therefore, non-geodesic in Zeman's terms. Therefore, two variables will be assigned on Zeman's method. Clearly, the lines involved in my clause 2(b) (i.e., clipped lines) are always non-geodesic lines, but not vice versa. Therefore, our reading method will generate fewer variables than Zeman's. Also, recall that Zeman requires that any joint should be broken into three parts. For example, all the following graphs will receive three variables on Zeman's reading:



However, my new reading method assigns only one variable to the first three graphs, since no portion of any LI network in these three graphs crosses a cut entirely. There are two clippings in the fourth graph, and three clippings in the fifth graph. Therefore, two variables will be given to the fourth, and three to the fifth graph. In many cases, fewer variables are being generated by my reading method. We expect the difference in the number of types of variables will lead us to simpler looking first-order sentences as final translations of graphs. This benefit comes from our effort to implement Peirce's intuition about the continuity of a line with as little modification as possible.

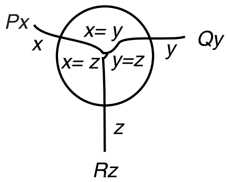
Let's move to clause 3. For each end of an LI network, if there is a predicate *P*, then replace this predicate with atomic wff $Pv_1 \ldots v_n$, and if there is no predicate, then write $\top$. The following shows how this rule is applied:



Recall that Zeman's reading also wrote an atomic wff at the end of an LI. The main difference between his method and this new one is that Zeman introduced three kinds of temporary predicates at loose ends, loops, and joints in order to write atomic wffs. My reading algorithm, by contrast, uses only existing predicates, and for a loose end we write '$\top$'. Clearly, this difference makes my new reading method much simpler than Zeman's.

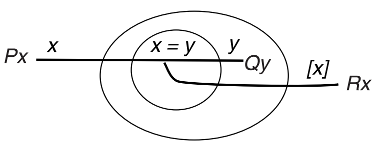
The next step, clause 4, shows a departure from Peirce's endoporeutic reading, which all the existing methods have followed. As I discussed in the previous section, the endoporeutic reading prevents us from implementing Peirce's visual distinction between universal and existential quantification. It seems to be quite clear that Peirce, like all of his followers, never considered an alternative to the endoporeutic reading. I strongly suspect that this is a main reason why Peirce himself did not insist on any systematic reading algorithm reflecting his own insight on the visual distinction between universal and existential quantification.

Here I suggest that the endoporeutic method be abandoned and the visual features for universal and existential quantification be saved. Instead of the endoporeutic method, I adopt the NNF reading of the Alpha system, which I introduced in the previous chapter. As argued before, the NNF reading method uses visual features which the endoporeutic method does not. We treat atomic wffs (written by clause 3(a)), $\top$ (written by clauses 3(b) and 3(c)), and identity wffs (written at the joints of the branches according to step 2(b)(ii)) as sentence symbols of the Alpha system. For example, from the following graph, clause 4 obtains the wff '$Px \wedge Qy \wedge Rz \wedge (\neg x = y \vee \neg y = z \vee \neg x = z)$':



Step 5 tells us how to obtain a final sentence out of the complex wff formed in the previous step, by reading off the visual distinction between *some* and *all*. Let's recall that in step 2 variables are written at the outermost parts of LIs.[70] According to whether a variable is written in an E-area or an O-area, quantifier ∃ or ∀ is adopted, respectively. In the graph above, since all of the variables are written down in an E-area, we use three existential quantifiers to get '$\exists x[Px \wedge \exists y(Qy \wedge \exists z[Rz \wedge (\neg x = y \vee \neg y = z \vee \neg x = z)])]$'.

The next example requires the use of a universal quantifier. Suppose that we have the following quasi-Alpha graph by steps 2 and 3:



Next, function $\bar{f}$ in step 4 is applied:

$\bar{f}(Px[[x = y]Qy]Rx)$

$= \bar{f}(Px) \wedge \bar{f}([[x = y]Qy]) \wedge \bar{f}(Rx)$ by 4(c)

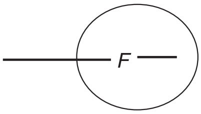
$= \bar{f}(Px) \wedge (\bar{f}([[x = y]]) \vee \bar{f}([Qy])) \wedge \bar{f}(Rx)$ by 4(d)

$= \bar{f}(Px) \wedge (\bar{f}(x = y) \vee \bar{f}([Qy])) \wedge \bar{f}(Rx)$ by 4(b)

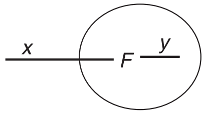
$= Px \wedge (x = y \vee \neg Qy) \wedge Rx$ by 4(a)

Variable x is written in an evenly enclosed area, and y in an oddly enclosed area. Therefore, step 5 tells us to introduce quantifiers to obtain '$\exists x[Px \wedge \forall y(x = y \vee \neg Qy) \wedge Rx]$'.

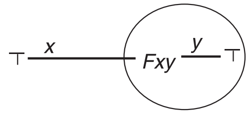
In addition to the choice of ∃ or ∀, we need to pay attention to scope. Clause 5(c) reflects Peirce's intuitive idea—the less enclosed an LI is, the bigger its scope is. In the above example, no subformula gets more than one quantifier. The following example will tell us how we apply clause 5(c) and clarify the meaning of 'the smallest NNF subformula'.



Beta Graph



by clause 2



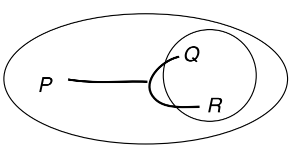
by clause 3

We obtain complex wff '$\top \wedge (\neg Fxy \vee \neg\top)$' by clause 4. Next, by clauses 5(a) and 5(b), we know $\exists x$ and $\forall y$ need to be added to get a sentence. Two points should be carefully observed. One is the condition 'the smallest NNF subformula', not just the smallest subformula; the smallest NNF subformula containing variable x is $\neg Fxy$, *not* Fxy. Likewise for variable y.[71] Both quantifiers need to be placed in front of one and the same smallest NNF subformula, and we know that the order of two different kinds of quantifier is crucial, which is the other important point to finish up the translation. Clause 5(c) draws our attention to the visual feature of where each variable is written. Since variable x is written in a less enclosed area than variable y is, we get the sentence '$\exists x \forall y \neg Fxy$'[72] as the final translation of the Beta graph above.

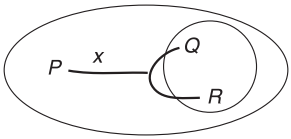
Notes 62 and 63 for step 5 are added for the following case:

'$\top \wedge \top$' is what we get by step 4. Variable x is written an E-area. However, since there is no subformula containing x, no quantifier is introduced. Hence, '$\top \wedge \top$' is the final form of the translation of this graph.

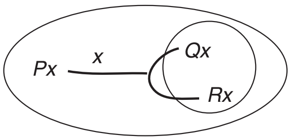
Now we will compare this new method with others through several examples. First, let's recall the following graph, which was used in the first section to show that Zeman's reading is not intuitive:[73]



We have seen step by step how Zeman's reading guides us to obtain the reading of this graph as '$\neg\exists x(Px \wedge \exists y \exists z(x = y \wedge y = z \wedge \neg(Qy \wedge Rz)))$', which is logically equivalent to a much simpler sentence '$\forall x(Px \rightarrow (Qx \wedge Rx))$'. My new algorithm leads us through the following steps:



by 2(a)



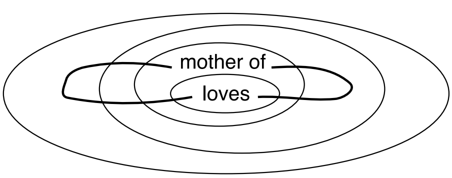
by 3(a)

$\neg Px \vee (Qx \wedge Rx)$ by 4

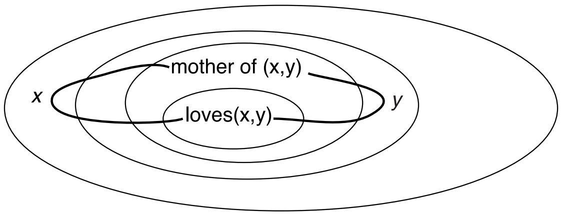
$\forall x[\neg Px \vee (Qx \wedge Rx)]$ by 5

This reading process is much simpler than Zeman's. Zeman's reading adopts three types of variables, while the new algorithm leads to only one. Hence, Zeman's algorithm, unlike the new one, does not read off the visual fact that this Beta graph has one unclipped LI network (*not* three). Also, Zeman's reading method never reads off the important iconic feature reflecting the distinction between universal and existential statements, but rather only produces existential quantifiers. Hence, universal statements are always expressed as a nested relation between negation and existential quantifiers. For these reasons, not only is the final translated sentence rather complicated looking, but also LIs become dispensable, which is why Zeman's method erases them in the process. This is quite ironic when we recall that the visualization of identity is one of the strengths of EG.

Another example I wish to revisit is the following graph, which was used in §5.1.2 to illustrate how the different goals of Zeman's and Roberts' projects on EG led to different problems:[74]



This graph has no double cut. Also, notice that neither of the two LIs has any portion which crosses any cut entirely. Therefore, we need one type of variable for each LI. Using these variables, we replace the two predicates with two atomic wffs 'MotherOf(x, y)' and 'Loves(x, y)':



Then, according to the NNF reading, we get the wff '¬MotherOf(x, y) ∨ Loves(x, y)'. Since x is written in an O-area and y in an E-area, the final reading is '$\forall x \exists y$(¬MotherOf(x, y) ∨ Loves(x, y))'. Only two variables and two quantifiers are introduced for two LIs. The final form of the translation does not need further manipulation to find a simpler equivalent sentence.

Recall that the major price that Robert's reading had to pay for its intuitiveness is its need to categorize certain kinds of graphs under special cases, rather than covering them under a general reading method.[75] Some of his special cases are given in figure 5.3. Under the heading *Special Cases*, Roberts gives us the following interpretations of the above graphs:

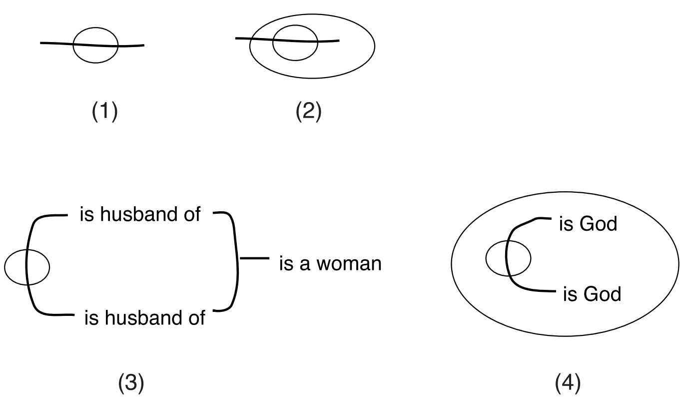


Figure 5.3
Examples of graphs that Roberts treats as special cases in his reading method.

Literally, [graph (1)] means "There is an object and there is an object and it is false that these objects"—denoted by the two unenclosed parts of the line—"are identical." By means of this graph, we can say such things as "Some woman has two husbands" ([graph (3)]). [Graph (2)] means "There is something, and it is false that there is something else non-identical to it"—that is, "Something is identical with everything." . . . [Graph (4)] expresses the Unitarian theology as summarized by Whitehead: "There is one God at most."[76]

Zeman's comprehensive reading method does not treat these as special cases. After broken points and temporary predicates have been introduced, we obtain rather complicated sentences as the translations of these graphs:

$$\exists x[x = x \wedge \exists y(x \neq y \wedge y = y)] \tag{1}$$

$$\exists x[x = x \wedge \neg\exists y(x \neq y \wedge y = y)] \tag{2}$$

$$\exists x\exists y[x \neq y \wedge \exists x\exists u\exists v(x = u \wedge u = v \wedge \text{HusbandOf}(x,z) \wedge \text{HusbandOf}(y,v) \wedge \text{Woman}(u))] \tag{3}$$

$$\neg\exists x\exists y(\text{God}(x) \wedge \text{God}(y) \wedge x \neq y) \tag{4}$$

According to my algorithm, after applying clauses 1, 2, and 3, the graphs are transformed into those given in figure 5.4. By clause 4, we obtain the following complex wffs:

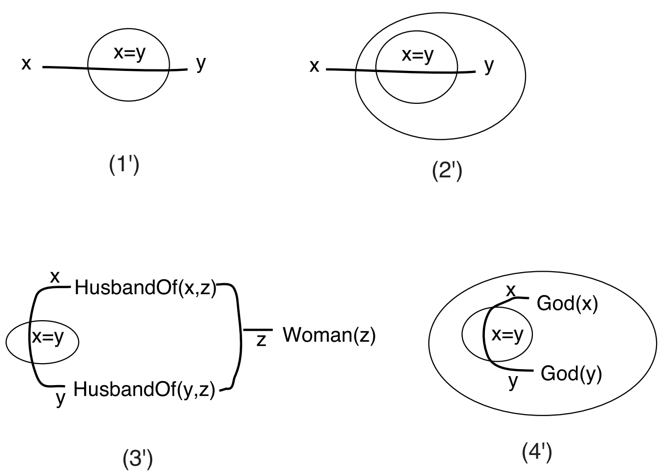


Figure 5.4
Roberts' special cases as transformed by my reading method.

$\top \wedge x \neq y \wedge \top$ from ($1'$)

$\top \wedge (x = y \vee \neg\top)$ from ($2'$)

$x \neq y \wedge \text{HusbandOf}(x, z) \wedge \text{Woman}(z) \wedge \text{Husband}(y, z)$ from ($3'$)

$x = y \vee \neg\text{God}(x) \vee \neg\text{God}(y)$ from ($4'$)

Depending on whether variables are written in an E-area or in an O-area, the last step (clause 5) tells us how to make sentences out of these wffs. The readings for these four graphs are as follows:

For ($1'$): $\top \wedge \exists x \exists y x \neq y \wedge \top$
(I.e., at least two things exist.)

For ($2'$): $\top \wedge (\exists x \forall y x = y \vee \neg\top)$
(I.e., something is identical with everything.)

For ($3'$): $\exists x \exists y(x \neq y \wedge \exists z[\text{HusbandOf}(x, z) \wedge \text{Woman}(z) \wedge \text{Husband}(y, z)])$
(I.e., some woman has at least two husbands.)

For ($4'$): $\forall x \forall y[x = y \vee \neg\text{God}(x) \vee \neg\text{God}(y)]$
(I.e., at most one God exists.)

Unlike Roberts' method, my new reading algorithm covers these cases without treating them as special. Hence, this reading is more systematic than Robert's. At the same time, this new method more easily yields simpler sentences than Zeman's. Is it as comprehensive as Zeman's? Yes, the equivalence between this reading and Zeman's can be proven quite easily. The main idea is to check whether the job of Zeman's temporary predicates is done in this new reading without them. According to Zeman, the temporary predicate L^1 is written at a loose end and is replaced by $x = x$, while a loose end is filled with $\top$ in the new method. Zeman's temporary predicate L^2 is written at a broken loop, and is later rewritten as $x = y$. As discussed, my clause 2(b), which assigns variables and writes an identity statement $x = y$ inside a cut, writes $x = y$ not for every loop, but only when a portion of an LI crosses an odd number of cuts entirely. Even when a loop takes place in an E-area, Zeman's final reading will contain the following formula: '$P(x) \wedge Q(y) \wedge x = y$'.[77] However, in the new reading, only one variable will be assigned in this case. Therefore, the corresponding formula will be '$P(x) \wedge Q(x)$', which is equivalent to '$P(x) \wedge Q(y) \wedge x = y$'. Zeman's temporary predicate L^3 can be explained away in similar fashion. In the previous chapter, we showed the equivalence between the endoporeutic and NNF readings. Zeman's reading, which adopts the endoporeutic method, allows us to use only existential quantifiers, while our method introduces both universal and existential quantifiers. However, these two are equivalent, since $\forall x \neg P(x)$ is logically equivalent to $\neg \exists x P(x)$.

5.3 Transformation rules

Skepticism toward the Beta system as a practical deductive system has been even stronger than in the case of the Alpha system. Not only are Beta graphs more difficult to read off than Alpha graphs, but also the transformation rules have been considered much more difficult to comprehend than Alpha rules. In this section, we find out where this strong criticism comes from and restate Peirce's original rules to make the Beta system more efficacious.

The inference rules of the Beta system are summarized by Don Roberts in the following way:[78]

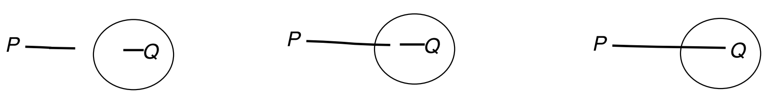
R1 *The rule of erasure* Any evenly enclosed graph and any evenly enclosed portion of a line of identity may be erased.
R2 *The rule of insertion* Any graph may be scribed on any oddly enclosed area, and two lines of identity (or portions of lines) oddly enclosed on the same area may be joined.
R3 *The rule of iteration* If a graph *P* occurs on *SA* [*the sheet of assertion*] or in a nest of cuts [*an area enclosed by cuts*], it may be scribed on any area not part of *P*, which is contained by $\{P\}$.[79] Consequently,
(a) a branch with a loose end may be added to any line of identity, provided that no crossing of cuts results from this addition;
(b) any loose end of a ligature may be extended inwards through cuts;
(c) any ligature thus extended may be joined to the corresponding ligature of an iterated instance of a graph; and
(d) a cycle may be formed by joining, by inward extensions, the two loose ends that are the innermost parts of a ligature.
R4 *The rule of deiteration* Any graph whose occurrence could be the result of iteration may be erased. *Consequently*,
(a) a branch with a loose end may be retracted into any line of identity, provided that no crossing of cuts occurs in the retraction;
(b) any loose end of a ligature may be retracted outwards through cuts; and

(c) any cyclical part of a ligature may be cut at its innermost part.

R5 The rule of the double cut The double cut may be inserted around or removed (where it occurs) from any graph on any area. And these transformations will not be prevented by the presence of ligatures passing from outside the outer cut to inside the inner cut.

The transformation rules for the Beta system are extensions of those for the Alpha system. Peirce and Peircean scholars extended the transformation rules of the Alpha system by retaining all the names of rules for the Alpha system and making the content of each rule more complicated in the Beta system with the addition of special instructions about LIs. For example, Peirce himself added two long notes under the rules of iteration and deiteration.[80] Roberts restructured these notes and wrote up four clauses after the iteration rule and three clauses after the deiteration rule, as seen in the rules above. The main motivation behind these efforts was to preserve simplicity so that both systems have the same number of and the same names for the rules. I claim that this way of handling the rules of the Beta system, in spite of some merits, made each rule quite complicated and difficult to use.

Through several examples, I will illustrate why it is not always easy to bring permissible syntactic manipulations of identity lines under the existing rules for the Alpha system. For example, the transformation from graph (1) to graph (3) should be allowed:



(1) (2) (3)

According to Roberts' interpretation, the iteration rule allows us to change (1) to (2), and the rule of insertion takes care of the transformation from (2) to (3).

I would like to raise questions about the first of these steps. How is the iteration rule applied for the transition from graph (1) to graph (2)? Which part of graph (1) is iterated in graph (2)? Is a line of identity iterated? Roberts would justify this step by citing the following clause of the iteration rule:

R3 The rule of iteration If a graph *P* occurs on *SA* or in a nest of cuts, it may be scribed on any area not part of *P*, which is contained by {*P*}. Consequently, ... (b) any loose end of a ligature may be extended inwards through cuts.[81]

The main idea of the iteration rule is that when certain conditions are satisfied, we are allowed to copy (i.e., reiterate) a certain part of a graph. However, it is not clear at all why (b), the extension of a line, is a consequence of the iteration rule.

Another point I would like to make is that when the task is to transform graph (1) to graph (3), it is not easy to see that the rule of iteration should be applied to (1) first since the difference between graphs (1) and (3) does not seem to be a result of the reiteration of certain part of graph (1). Therefore, it is rather difficult to choose the correct rules for this deduction.

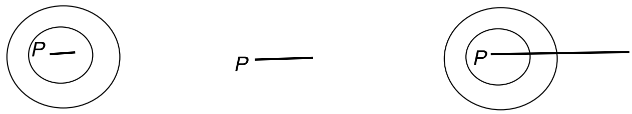
Roberts' presentation of clause R3(b) is a direct translation from the following passage of Peirce's note on the rule of iteration:

Note A ... the right to iterate includes the right to draw a new branch to each ligature of the original replica inwards to the new replica.[82]

This is not an explanation of an existing rule, i.e., the iteration rule, but a stipulation of a new rule under the same name so that the rule of iteration in the Beta system may handle more extensive cases than the rule of iteration in the Alpha system. Theoretically, this decision, like most other stipulations, is fine. However, from a practical point of view, this stipulation hurts EG as a useful deductive system: The meaning of 'iteration' is not intuitive any more, and accordingly, the iteration rule becomes quite difficult to apply. It is much more natural and intuitive to say that the line of identity in graph (1) is *extended*, rather than *iterated*, inward through a cut to obtain graph (2). Therefore, I suggest that the word 'extension', not 'iteration', be chosen for this operation, which should be separated from the iteration rule.

Further evidence for my main claim that separate rules for LIs should be stated can be found in the rule of biclosure, R5. Roberts' version of this rule is simple as quoted above. This rule justifies the transformation of the left-hand graph to the right-hand one:[83]

However, the simple version of the rule raises some problems. In the following, Roberts would transform graph (1) to graph (3) (or vice versa) by way of (2), that is, by applying R5 twice:

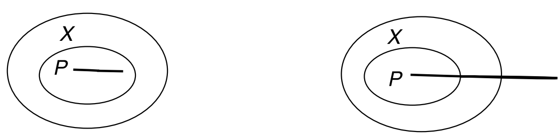


(1) (2) (3)

That is, the transformation from (1) to (3) is considered two manipulations of a double cut, not a single manipulation of a line of identity. This does not reflect our visual intuition about these two graphs at all. The difference in visual features between graphs (1) and (3) is quite clear: In one, the identity line ends inside the double cut and in the other outside it. This visual difference should be reflected in the transformation rules. If the system forces us to go through graph (2) to obtain (3), our visual instinct is not respected at all and the graphical system loses its visual power. This is another clear instance to show that visual intuitiveness requires a separate rule for identity lines allowing a direct transformation from graph (1) to graph (3), rather than going through graph (2).

The real problem with Roberts' simple way of handling the biclosure rule is illustrated in the following example:

Example 5.2 According to Roberts' formulation of the biclosure rule, we cannot get the transformations between the following two graphs (let *X* be any graph):



Since graph *X* is in the outer area, neither graph has a double cut, and Roberts' R5 does not seem to apply. As Roberts emphasizes after introducing the biclosure rule (R5), "the new application of R5 is restricted to cases in which lines of identity only are on the outer area of the double cut, and where these lines do not terminate on that area but pass all the way from outside the outer cut to inside the inner cut."[84] The case of the two graphs above cannot be handled in Roberts' restatements of the Beta rules, and this is a major flaw.

Zeman's double cut rule is more cautiously stated to handle similar cases as example 5.2:

Rule 0.11 Any graph may be enclosed by two cuts as in 0.05,[85] with *the added provision* that lines of identity may pass from entirely outside the outer to entirely inside the inner cut; all they are allowed is "free direct passage" through the annular space between the cuts.[86]

Zeman's rule 0.11 seems to involve two different kinds of manipulation: one for double cuts, and the other for "direct passage" for LIs. However, what he wants to address are the conditions under which we may draw a biclosure, not the conditions under which we may extend an LI. With the added provision, Zeman could say that the two graphs in example 5.2 are the same graph in that no transformation is required. Because he lacks a clear distinction between syntax and semantics, Zeman's biclosure rule for the Beta system ignores the following visual representing facts: when an LI terminates inside a cut and when an LI terminates outside a cut.

Peirce's original way to state his biclosure rule shows his confusion between syntax and semantics as well:

Rule 4 Called The Rule of Biclosure Two seps [scroll], one within the other, with nothing between them whose significance is affected by seps [cuts], may be withdrawn from about the graph they doubly close.[87]

Peirce's way of stating the biclosure rule has some room for accommodating the transformation in example 5.2 that Roberts' version lacks. The significance of graph *X* in the example is not affected in either graph since it stays in the same place and the extension of the LI does not make any change to the meaning of graph *X*. Hence, if the condition "nothing between them whose significance is affected by seps" is applied broadly, then Peirce's biclosure rule applies to this example. However, it should be noted that the phrase 'whose significance is affected', which is part of semantics, should not be used in transformation rules, which should remain strictly syntactic.

Again, this problem arises from a now-familiar source: Peirce and these scholars did not want to add any new inference rules to the list of the Alpha rules. Instead, provisions are added to each rule. Unfortunately, these provisions are either unclear or too far-fetched to be included under the relevant rubric.

Rather than interpreting 'drawing' (or 'erasing') in an O-area (or in an E-area) more extensively for identity lines,[88] I suggest that new rules be written (preferably in a symmetric form) to show how we may manipulate identity lines. My suggestion is consistent with the main idea of a natural deductive system, that is, that inference rules are written for each syntactic device of a given system. When the Beta system introduces another kind of syntactic object than in the Alpha system, i.e., identity lines, it is natural to expect additional rules for this new syntactic device.

In the previous chapter I restated the transformation rules of the Alpha system in terms of E-areas and O-areas, by implementing more visual features. I will preserve all of these rules for the Beta system without any modification and will introduce several new symmetric transformation rules for identity lines. In the case of sentence symbols or cuts, only two kinds of manipulating options are available: either to erase or to draw them. However, in the case of LIs, more kinds of syntactic operations are needed: (i) to extend or retract a portion of an LI through cuts, (ii) to join or disjoin an LI, and (iii) to add or erase branches of an LI. The following rules allow these manipulations more easily.

Transformation rules reformulated

NR1 In an E-area, say area *a*,

(a) we may *erase* any graph, and

(b) we may *draw* an LI or graph *X* if there is a token of *X*

  (i) in the same area, i.e., area *a*, or

  (ii) in the next-outer area from area *a*.

NR2 In an O-area, say area *a*,

(a) we may *erase* graph *X* if there is another token of *X*

  (i) in the same area, i.e., area *a*, or

  (ii) in the next-outer area from area *a*, and

(b) we may *draw* any graph.

NR3 We may *extend* a loose end of an LI

(a) inwards through cut(s), and

(b) outwards

  (i) from an O-area to an E-area,

  (ii) from an O-area to an O-area, or

  (iii) from an E-area to an E-area unless there is another LI

    (A) which is attached to the same predicates,

    (B) whose scope is bigger than the LI we are interested in extending, and

    (C) whose outermost part is in an O-area.

NR4 We may *retract* a loose end of an LI

(a) inwards

  (i) from an O-area to an E-area,

  (ii) from an E-area to an E-area, or

  (iii) from an O-area to an O-area unless there is another LI

    (A) which is attached to the same predicates,

    (B) whose scope is bigger than the LI we are interested in retracting, and

    (C) whose outermost part is in an E-area, and

(b) outwards through cut(s).

NR5 We may *join* two loose ends of identity lines

(a) in an O-area, or

(b) in an E-area if the subgraphs to be connected are tokens of the same type.

NR6 We may *disjoin* an identity line

(a) in an O-area if subgraphs to be disjoined are tokens of the same type, or

(b) in an E-area.

NR7 A *double cut* may be erased or drawn around any part of a graph.

NR8 We may add or erase any *branch* of a line of identity without crossing a cut.

Table 5.1
Rules NR1 and NR2 for Beta graph *X*

	E-area	O-area
Erase	*X*	*X* if there is another *X* either in the same area or in the next-outer area
Draw	*X* if there is another *X* either in the same area or in the next-outer area	*X*

NR9 We may form or break a *cycle* (closed curve) of a branch of an LI in the innermost part of the ligature.

The symmetries in the first six rules can be summarized as in tables 5.1 through 5.3.

How I restated the transformation rules not only makes them easy to use but also easy to understand. Hence, the soundness of the system becomes more intuitive than before.[89] Since we showed the validity of NR1, NR2, and NR7 in the previous chapter on the Alpha system,[90] we need only to look into the validity of the new rules in the Beta system which involve the manipulations of LIs, i.e., NR3 through NR6, NR8, and NR9.

Two crucial visual features of LIs should be noted in the Beta system. One is whether the outermost part of an LI is in an E-area or in an O-area, since in the new reading algorithm, this feature determines whether universal or existential quantification is assigned. The other key feature is the degree of enclosure of the outermost part of an LI: the less it is enclosed, the larger its scope is. Keeping these two points in mind, we check the validity of NR3 and NR4.

Since neither NR3(a), inward extension, nor NR4(b), outward retraction, makes any change to the outermost part of an LI, there will be no change either in the kind of quantification assigned to the LI or in its scope. There are two cases in which NR3(a) may be applied:

i. When an innermost loose end of an LI extends from an O-area to an E-area. When a loose end is in an O-area, it is translated into $\neg\top$. Sup-

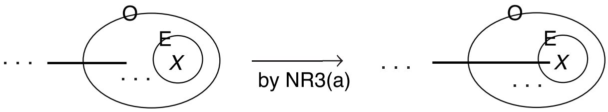
Table 5.2
Rules NR3 and NR4 for LIs

	inwards	outwards
Extend	through cut(s).	from an O-area to an E-area, from an O-area to an O-area, or from an E-area to an E-area unless there is an LI (A) which is attached to the same predicates, (B) whose scope is bigger than the LI we are interested in extending, and (C) whose outermost part is in an O-area.
Retract	from an O-area to an E-area, from an E-area to an E-area, or from an O-area to an O-area unless there is an LI (A) which is attached to the same predicates, (B) whose scope is bigger than the LI we are interested in extending, and (C) whose outermost part is in an E-area.	through cut(s).

Table 5.3
Rules NR5 and NR6 for LIs

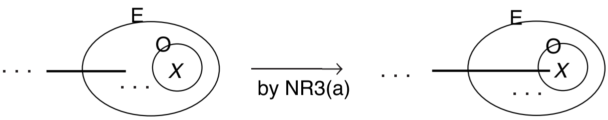
	E-area	O-area
Join	if subgraphs to be connected are tokens of the same type	yes
Disjoin	yes	if subgraphs to be disjoined are tokens of the same type

pose that there is a graph *X* inside the E-area into which the loose end is extended. That is,



Suppose that the translation of graph *X* is α. Then the translation of the original graph (the graph on the left-hand side) looks like '$\ldots\neg\top\ldots\vee\alpha\ldots$', and the translation of the graph obtained by applying NR3(a) (the graph on the right-hand side) would be '$\ldots\vee[\top\wedge\alpha]\ldots$'.[91] So this transformation is valid.

ii. When an innermost loose end of an LI extends from an E-area to an O-area. That is,

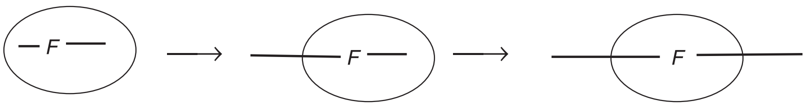


The graph on the left-hand side is translated into '$\ldots\top\wedge\neg\alpha\ldots$', while the translation of the graph on the right-hand is '$\ldots\neg\top\vee\neg\alpha$'. Again, this manipulation is valid. NR4(b) is a mirror image of NR3(a).

However, when we extend an LI outwards or retract an LI inwards, it might alter the following two important representing facts identified above: (i) whether the outermost part of an LI lies in an E-area or in an O-area, and (ii) which LI's outermost part is less enclosed. The line that might be extended outwards or retracted inwards might get a different quantification or it might change the scope among the LIs.[92] Again, in showing the validity of each subclause of rules NR3(b) and NR4(a), the meanings of E-areas and O-areas play an important role. Since we do not want to derive a universal statement from an existential statement, neither NR3(b) nor NR4(a) allows extension or retraction from an E-area to an O-area. We will look at the details for the other three cases, i.e., from an O-area to an E-area, from an O-area to an O-area, and from an E-area to an E-area.[93]

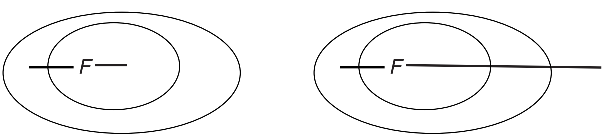
NR3(b)(i) allows us to extend an LI from an O-area to an E-area. Then two kinds of changes may occur in the translation involving the

LI. One is the change from a universal quantifier to an existential quantifier, because of the change in the area where its outermost part ends. The other possible modification is that the extension of this line might change the scope relative to other lines. Hence, manipulation by NR3(b)(i) corresponds to $\ldots\forall x\ldots \Rightarrow \ldots\exists x\ldots$,[94] $\ldots\forall y\underline{\forall x}\ldots \Rightarrow \ldots\underline{\exists x}\forall y\ldots$,[95] or $\ldots\exists y\underline{\forall x}\ldots \Rightarrow \ldots\underline{\exists x}\exists y\ldots$.[96] We find each case valid. For example, the following transformations are allowed by NR3(b)(i):



NR3(b)(ii) does not change the kind of quantifier assigned to the extended LI, but it might change its scope. Then the derivation would be either $\ldots\forall y\underline{\forall x}\ldots \Rightarrow \ldots\underline{\forall x}\forall y\ldots$ or $\ldots\exists y\underline{\forall x}\ldots \Rightarrow \ldots\underline{\forall x}\exists y\ldots$. Either case would be valid.

However, when we extend a loose end from an E-area to an E-area, a possible change in scope becomes crucial. For example, no system should allow the derivation of $\exists y\forall x$Loves(x, y) from $\forall x\exists y$Loves(x, y). In the case of the Beta system, we should prevent the extension of an LI from an E-area to an E-area if the extension overrides an existing universal quantifier. The three subclauses of NR3(b)(iii) describe the constraint to block the following invalid step: $\ldots\forall y\underline{\exists x}\ldots \Rightarrow \ldots\underline{\exists x}\forall y\ldots$. Hence, in the following, graph (1) cannot be transformed to graph (2), since there is an LI satisfying three subclauses (A), (B), and (C) of NR3(b)(iii). On the other hand, we may transform graph (2) to graph (1) by NR4(a)(ii).



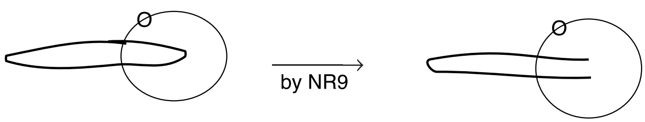
(1) (2)

NR5 and NR6 can be understood to be valid when we recall the discussion in §4.2.1 about the different meanings of juxtaposition in an O-area and juxtaposition in an E-area. The processes in NR5 and NR6 are quite parallel to those of NR1 and NR2, which is the main reason why Peirce included both joining and disjoining LIs in his insertion and erasure rules, respectively.[97]

The operation by NR8 may take place without crossing a cut, which means that there will be no change in the kind of quantifier already attached to the LI and no change in scope. Thus, it is valid.

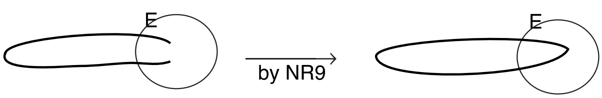
Half of NR9 is clearly valid. Since we may join two loose ends in an O-area and we may disjoin an LI in an E-area, forming a cycle in an O-area and breaking a cycle in an E-area are valid steps. Since the interpretations of a cycle and a line drawn in one and the same area are identical with each other, breaking or forming a cycle in one area is a valid step. Hence, two additional cases need to be examined:

i. Breaking a cycle in an O-area



The part on the left side is translated into '… $\top \wedge (\neg\top \vee \ldots) \wedge \ldots$', and the part on the right side is '… $(\neg\top \vee \neg\top \vee \ldots) \wedge \ldots$'. So this is a valid process.

ii. Forming a cycle in an E-area



By forming a cycle in an E-area, we transform $\top \wedge \top$ to $\neg\top \vee \top$, which is legitimate.

One more question remains: Are these reformulated rules equivalent to Peirce's original rules? I have shown in §4.4.2 that my rules (without the rules for LIs) give exactly the same results as Peirce intends for his Alpha rules to do. Now we have to check whether the new rules for Beta graphs, i.e., NR3 through NR6, NR8, and NR9, cover what Peirce's original "extended" rules awkwardly state about manipulating lines of identities. My new version categorizes rules according to what kinds of symmetric operations are made to lines—extending versus retracting,

Table 5.4
Equivalences between Peirce's rules and my versions

Peirce's version	New version	Operations on identity lines
R3(a)	NR8 [add]	Add a branch
R3(b)	NR3(a)	Extend a line
R3(c) & R2 [line part]	NR5	Join loose ends
R3(d)	NR9 [form]	Form a cycle
R4(a)	NR8 [erase]	Erase a branch
R4(b)	NR4(b)	Retract a line
R4(c)	NR9 [break]	Break a cycle

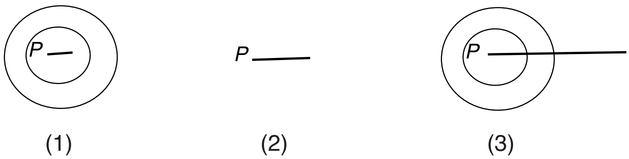
joining versus disjoining, adding versus erasing branches, and forming versus breaking a cycle. The symmetry displayed in my reformulation of rules is much more pervasive than Peirce's symmetry between iteration and deiteration. Table 5.4 is a summary of the equivalences between Peirce's original rules and my new versions.

There seems to be no rule equivalent to NR6, disjoining an identity line. But Peirce's R1 allows us to erase a portion of a line in an E-area to disjoin the line. Hence, the transformation from the graph on the left to the graph on the right is allowed in both sets of rules:

How about the following transformation?

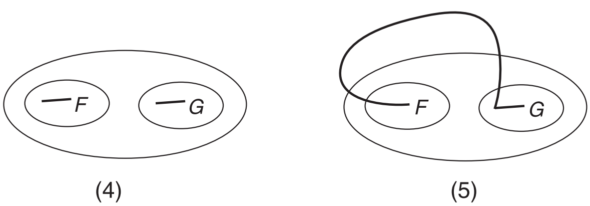
It is not difficult at all to see that NR6(a) is needed for this manipulation, since the visual difference from the first graph to the second is that the line has become disjoined. Peirce allows this in a rather complicated way. None of the clauses under his deiteration rule addresses this specific manipulation,[98] but the rule is defined as follows: *Any graph whose occurrence could be the result of iteration may be erased.* No rule of disjoining a line is specified. Instead, we should consider a hypothetical application of the rule for joining LIs and reverse it, this rule says. The manipulations in the two examples above are handled under the same rule in my set of rules, but not in Peirce's. Because disjoining a line is a clear visual feature common to these two cases, my way of handling the inference is more efficacious than Peirce's.

Also, according to table 5.4, there seems to be no rules of Peirce's corresponding to NR3(b) and NR4(a). I will show that these two rules cover the case that led to a question about Roberts' version of the Beta system at the beginning of the section. Let's return to the following example:

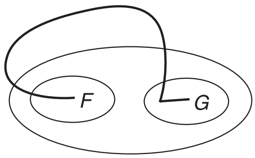


Since there is no rule of Peirce's that directly allows the transformation from graph (1) to graph (3),[99] the double cut rule, R5, is applied twice for this transformation. In my new reformulated rules, there is no need to take a detour through graph (2), since I allow a line to be extended or to be retracted through two cuts in a row: The extension rule NR3(b) can be directly applied to graph (1) to obtain graph (3), and the retraction rule NR4(a) to graph (3) to obtain graph (1). The visual difference between the two graphs in this example, that is, that the identity line is extended in the second graph, is the key to selecting an appropriate rule.

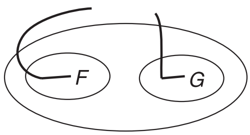
Before finishing the section, let's see how my reformulated rules, along with our new understanding of more fine-grained visual features of the Beta system, could help us to derive one graph from another more easily than before. The following pair of logically equivalent graphs is cited as a "more complex proof" by Roberts.[100]



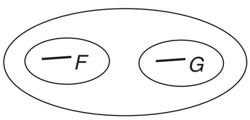
As Roberts says, the inference from graph (5) to graph (4) is easier than the inference in the other direction. The first thing that the reader would notice is that we need to disjoin the line in graph (5) and retract both ends. Hence, all we have to check are the rules for disjoining an LI in an E-area and for retracting an LI from an E-area to an E-area. Clauses NR6(b) and NR4(a)(ii) allow these two kinds of operations.



(5)

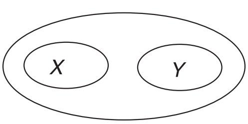


NR6(b)



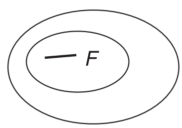
NR4(a)(ii)

A similar approach would not work for the inference from graph (4) to graph (5). We can extend both LIs in graph (4) to the outermost area in the graph, but we cannot join two LIs in an E-area. Since we have no rule to allow the extension of an LI from an E-area to an O-area, each line in graph (4) ends in an E-area, and since we may join LIs only in an O-area, there seems to be no way to join these two lines as the graph stands now. Let's recall the reading of the following graph schema in our Multiple-Readings method for the Alpha system:

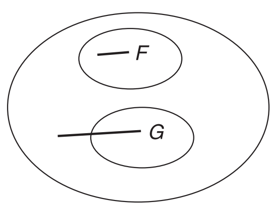


According to my new algorithm, this graph represents a piece of disjunctive information, either *X* or *Y*. Hence, graph (4) means that *either* something is *F or* something is *G*. What we need to prove is that in either case we may deduce that something is *F* or *G*. The idea is to draw (by applying rules) the same graph (5) in two areas in which *F* is written or in which *G* is written. Hence, we would like to check whether we can derive graph (5) from the following graph, which means that something is *F*.

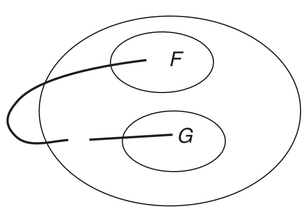
First of all, we need to create an O-area in order to draw *G* with a line in the graph. When we do not have any token of *G*, the only permissible area for drawing is an O-area. So we apply the double cut rule to obtain the following:



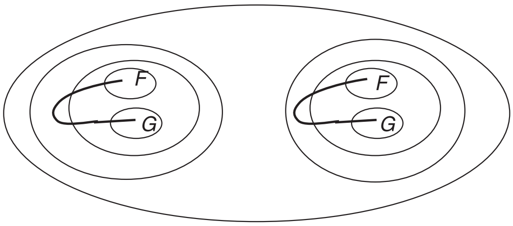
It is tricky to find out which graph we need to draw in the O-area. In the above, we noticed that we could not directly transform graph (4) to graph (5) mainly because we could not find a way to make two loose ends of LIs meet in an O-area. Hence, we need to draw a graph in the O-area of the graph immediately above satisfying the following conditions: (i) It should have predicate *G* and an attached line, (ii) it should have a cut, and (iii) the line should end in the O-area. Therefore, the following graph is produced by applying NR2(b):



Now we apply the extension rule twice, i.e., NR3(b)(iii) for the outward extension and NR3(a) for the inward extension, to obtain the following graph:



NR5(a) allows us to join the two loose ends to get graph (5). Similarly, we can also derive graph (5) from the graph that something is *G*. Therefore, we know that the following graph is derivable from graph (4):



We then apply the rule of erasure in an O-area (i.e., NR2(a)) and the rule of a double cut to prove that graph (5) is derivable from graph (4).

In this example, using fine-grained visual distinctions between the graph that we have and the graph that we aim to get, we searched for correct rules to apply. We used the meaning of graph (4) given by our Multiple-Readings method, which operations are permissible in which areas (either an E-area or an O-area), retraction or extension of LIs, and joining and disjoining LIs. As shown, my new version of the transformation rules corresponds to the visual features easily noticeable in inference steps and makes the user choose correct rules more efficiently.

The revision of the Beta rules confirms the conjecture made at the end of the revision of the Alpha rules: To add efficacy to a representation system, one needs to understand the nature of the system, because there are different ways for different kinds of representation systems to be natural. Further testing of this conjecture is left for future work.

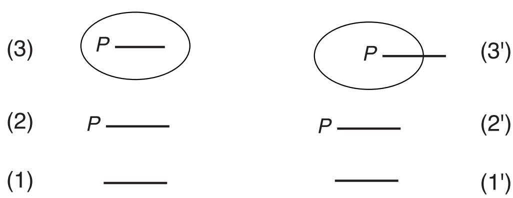
5.4 Appendix: direct semantics

In one of the most recent works on Peirce's Beta graphs, Hammer provides a Tarskian semantics for the EG Beta system.[101] Hammer's work adopts a different approach from the works we examined in §5.1. Both Zeman and Roberts focused on how to read off Beta graphs. Since each of these two readings is written in (or easily convertible into) a first-order language, Tarskian semantics can be applied afterwards to get the semantics of the system. By contrast, Hammer applies a Tarskian semantics *directly* to the Beta system, not by a detour through a first-order-language reading. This attempt sounds attractive, since it seems to suggest a more direct way of understanding the system. After I outline Hammer's method with some revisions for clarification, we will see why a direct Tarskian semantics does not guarantee an easier understanding of Beta graphs.

Hammer extends the set of Beta graphs to a new set, called the set of *well-formed graphs*, by allowing variables to be written in a Beta graph, and he gives a Tarskian recursive semantics to the extended set. The motivation for the new extended set of *well-formed graphs* is expressed thus:

> Peirce did not allow variables to appear in his graphs. . . . The endoporeutic method of interpretation, however, suggests that a formal semantics will need to make use of some sort of bookkeeping device to keep track of the lines already interpreted.[102]

Hammer claims that the endoporeutic interpretation requires the introduction of variables to graphs. Why so? Despite the fact that the set of Beta graphs is defined inductively, its semantics cannot be given recursively on the basis of these inductive definitions. For example, let us compare the derivational histories of the following two graphs, (3) and (3′):



When we write up the relevant clauses for the history of each graph, interestingly enough we find the same clauses are needed for both of them.

1. A line of identify is a Beta graph.

2. If we write a unary predicate P at a loose end of the line of a Beta graph, then the result is also a Beta graph. (By predicate closure)

3. If we draw a single cut in any subpart of a Beta graph, then the result is also a Beta graph. (By cut closure)

A problem arises when we try to provide a recursive semantics for graph (3) and graph (3′) based upon this inductive clause. How can we get a correct result such that the first graph is read $\neg\exists xPx$ and the second $\exists x\neg Px$? Relying on a syntactic derivational history is not enough, since the same clauses are applied in the same order to get two graphs whose meanings are different from each other. We need the semantics to reflect the scope relation of an identity line, which the simple syntactic definition above cannot.[103] I suspect that Hammer realizes the difficulty in getting the correct semantics out of a simply written syntactic definition of a Beta graph. According to the endoporeutic method, we interpret the cut prior to the line in graph (3), while we should interpret the line prior to the cut in graph (3′). In the latter case, before we interpret the cut to be a negation, we need to make a record to remind us that there was a line. Hammer's idea is to erase lines of identity one by one, but to keep track of them with variables.[104] This is what Hammer means by 'bookkeeping device' in the above quotation and is the motivation behind the following definition of the set of well-formed graphs.[105]

Definition 5.2 The set of *well-formed graphs*, say $\mathcal{G}_v$, is the smallest set satisfying the following:

S1 Basic set The following are in $\mathcal{G}_v$:

(a) $Rx_1 \ldots x_n$, where R is a n n-place predicate, and $x_1, \ldots, x_n$ are variables.

(b) a line of identify or branching line of identity.

S2 Inductive clauses Let G and G' be in $\mathcal{G}_v$ and x be a variable.

(a) If G has a loose end and x is attached to a loose end, then the result is also in $\mathcal{G}_v$.

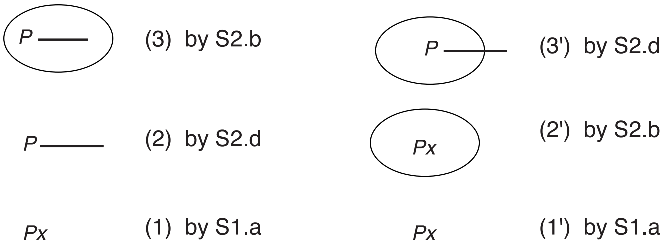
(b) If G is enclosed in a cut, then the result is also in $\mathcal{G}_v$.

(c) If G and G' are juxtaposed with each other, then the result is also in in $\mathcal{G}_v$.

(d) If x in G is replaced with a line of identity which is drawn outwards through all the cuts enclosing x, then the result is also in $\mathcal{G}_v$.

(e) If G has two loose ends and these are connected, then the result is also in $\mathcal{G}_v$.

Now the building histories for the graphs we discussed before are given in the following way (notice that Hammer's "bookkeeping device," i.e., variables, keeps track of the scope relation):

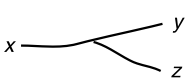


Since the original set of Beta graphs, say $\mathscr{G}$, is a subset of the inductive set $\mathscr{G}_v$, it suffices to define a semantics of $\mathscr{G}_v$ recursively using this inductive definition as a basis. I believe this is what Hammer aimed to do.

Before presenting the semantics, he introduces the notion of the *retraction of a loose end of a line with a new variable x*. The retraction process is needed to define the semantics of graphs constructed by inductive clause S2.d. In the example above, one should be able to retract a line both in (2) and (3′) to obtain (1) and (2′), respectively. Hammer presents an algorithm which substitutes lines for variables.[106] The main idea is to retract lines until the graph is decomposed into some of the following graphs[107] or cuts of one of the following graphs (let R be an n-ary predicate and x, y, z, $x_1, \ldots, x_n$ be variables):

$Rx_1 \ldots x_n$

x —— y



Suppose that there is no loose end. Then break the line at any part enclosed by no cut, write one new variable x at both of the just created loose ends and proceed with the retraction process as follows:

i. If there is a branch, the retraction stops just before the joint.

ii. If one reaches a loop, the retraction stops somewhere within the cut.

iii. If one reaches a loose end, the retraction stops just before the loose end.

iv. If one reaches another variable, the retraction stops just before the variable, leaving a line between these two variables.

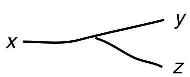
Let us follow Hammer's notation and let M be a model or interpretation $\langle U, I \rangle$, where U is a non-empty domain and I is a function for predicates and variables, and M_a^x is the same as M except that x is assigned to a in the domain by the new model. With these notations, the following is Hammer's semantics for set $\mathscr{G}_v$:[108]

Direct semantics Let M be an interpretation.

M1 $Rx_1 \ldots x_n$ is true in M iff $\langle I(x_1), \ldots, I(x_n) \rangle \in I(R)$.

M2 x —— y is true in M iff $I(x) = I(y)$.

M3 Graph



is true in M iff $I(x) = I(y) = I(z)$.

M4 $[G]$ (a single cut of G) is true in M iff G is false in M.

M5 $G_1 \ldots G_n$ is true in M iff $G_1, \ldots,$ and G_n are true in M.

M6 Let $G(x)$ be the same as G except that a loose end enclosed by no cut of G is retracted by a new variable x. Then G is true in M iff for some a in the domain, $G(x)$ is true in M_a^x.

M7 Let $G(x)$ be the same as G except that each half of a portion of a line enclosed by no cut of G is retracted by a new variable x. Then, G is true in M iff for some a in the domain, $G(x)$ is true in M_a^x.

Let us compare these semantic clauses with the syntactic clauses for well-formed graphs $\mathscr{G}_v$. Two differences are noticeable. One is that no single semantic clause assigns truth values to atomic graphs defined by syntactic clause S1.b. A single identity line is a well-formed atomic graph by syntactic clause S1.b. However, for its semantics, we need to apply more than one clause, as if it were a non-atomic graph, as follows:

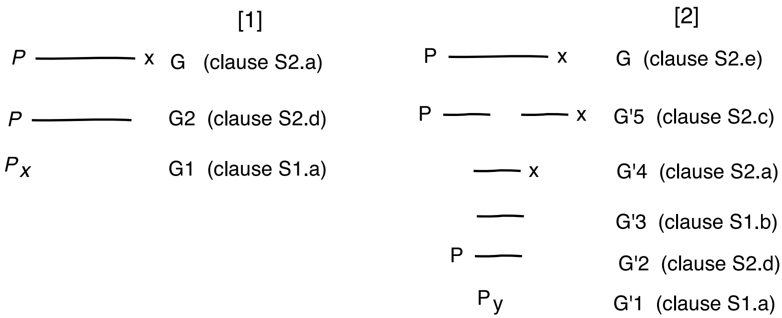


Figure 5.5
Graph G lacks a unique derivational history.

—— is true in M

iff for some a, x —— is true in M^x_a by M6

iff for some a and b, x —— y is true in M^{xy}_{ab} by M6

iff for some a and b, $I(x) = I(y)$ in M^{xy}_{ab} by M2

That is, the semantics for a single line is recursively defined, while it is atomic syntactically. This is not desirable, but it is not a serious defect.

The second difference is more crucial: There is no recursive clause corresponding to syntactic clauses S2.a and S2.d. Let me explain why Hammer has to choose this mismatch through the following example:

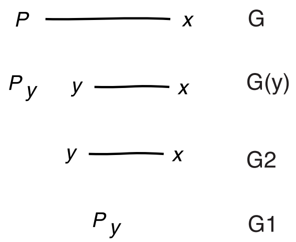
Example 5.3 There is no unique derivational history for graph G in figure 5.5. Obviously, any attempt to give a recursive semantics based upon this inductive definition would require one to prove that the uniqueness of the meaning of a graph is guaranteed in spite of this non-uniqueness of its syntactic history. Rather than proving this proposition, Hammer makes his semantics independent of his syntax. Therefore, the semantics of graph G does not reflect either of the above syntactic histories:

Given model M, G is true in M

iff $G(y)$ is true in M^y_a for some a by M7

iff G_1 and G_2 are true in M^y_a for some a by M5

iff $I(y) \in I(P)$ and $I(y) = I(x)$ in M^y_a for some a by M1 and M2



One more example (figure 5.6) will show the mismatch between syntax and semantics in a more striking way (we will come back to this example later).

Hammer was aware of this mismatch when he says "The semantic content of this syntactic operation is, in general, not obvious, but always involves some element of identifying two variables"[109] at the end of clause S2.d.[110] Rather than revising the syntactic or semantic definition, Hammer chose to sacrifice parallelism between syntax and semantics. This choice turns out to be undesirable both from the syntactic and the

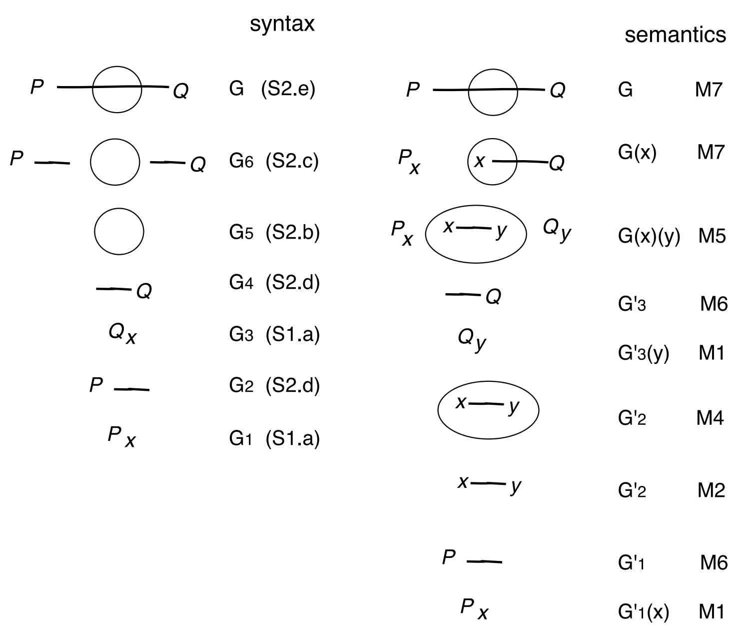


Figure 5.6
The mismatch between syntax and semantics in Hammer's system.

semantic point of view. Hammer's syntactic definition leaves the reader puzzled, since examples of its use are not given, and more importantly, his semantic clauses leave the reader wondering whether they are exhaustive or not. The goal of Hammer's task is simple, but the process remains unclear.

Since a lack of correlation between (inductive) syntactic and (recursive) semantic definitions makes Hammer's project fall short, these should be improved before we evaluate the merit of the project. We replace Hammer's set of well-formed graphs with the set of v-Beta graphs and revise the semantics so that the semantics is defined recursively on the basis of the inductive structure of the graphs. This revision keeps Hammer's original idea: Give a direct semantics to Beta graphs by extending Beta graphs to open Beta graphs. However, the main difference between Hammer's presentation and my revision is that the syntactic and semantic definitions of the set of v-Beta graphs ('v' for 'variable') are parallel to each other in this proposal, while in Hammer's they are not.

Definition 5.3 The set of v-Beta graphs $\mathcal{G}_{v\beta}$ is the smallest set satisfying the following:[111]

1. Let R be an n-place predicate and $x_1, \ldots, x_n$ variables. Then, $Rx_1, \ldots, x_n$ is a v-Beta graph. If R is an identity sign, then we write $x_1 = x_2$.
2. Let G be $x_1 = x_2$. If G' is the same as G except that we replace the identity sign with a line of identity (we write $G^{=}_{-}$), then G' is also a v-Beta graph.
3. Let G be a graph with variable x. If G' is the same as G except that we replace x with a line which extends through all the cuts enclosing x (we write G^{x}_{-}), then G' is also a v-Beta graph.
4. Let G be a graph with more than one token of variable x. If G' is the same as G except that we connect all tokens of x and the loose ends (if any) at which x is written, crossing no whole cut, and we erase the x's (we write $G^{x \ldots x}_{-}$), then G' is also a v-Beta graph.
5. Let G be a v-Beta graph. Then a cut of G (we write $[G]$) is also a v-Beta graph.

6. Let $G_1, \ldots, G_n$ be v-Beta graphs. Then, the juxtaposition of all these n graphs (we write $G_1 \ldots G_n$) is also a v-Beta graph.

The remaining step is to define the semantics. The inductive definition of the set of v-Beta graphs makes it clear how to define a recursive semantic function. We adopt Hammer's notation and let M be a model $\langle U, I \rangle$, where U is a non-empty domain and I is a function for predicates and variables. M_a^x is the same as M except that x is assigned to a in the domain by the model.

Definition 5.4 A graph is defined as true in a model as follows:

1. $Rx_1, \ldots, x_n$ is true in M iff $\langle I(x_1), \ldots, I(x_n) \rangle \in I(R)$.
2. Let G' be $G^{\underline{=}}$. Then G' is true in M iff G is true in M.
3. Let G' be $G^{\underline{x}}$. Then G' is true in M iff G is true in M_a^x for some a in the domain.
4. Let G' be $G^{\underline{x \ldots x}}$. Then G' is true in M iff G is true in M_a^x for some a in the domain.
5. Let G be $[G]$. Then G' is true in M iff G is false in M.
6. Let G' be $G_1 \ldots G_n$. Then G' is true in M iff $G_1, \ldots, G_n$ are each true in M.

This semantics is a mirror image of the revised syntax. Since the extended set contains Peirce's original Beta graphs, my revision completes what Hammer was aiming for: the assignment of a direct semantics to Beta graphs. At the same time, my revision solves the questions raised about Hammer's project relating to the mismatch between syntax and semantics.

Let me illustrate, through the example in figure 5.6, how these improvements make the original project easier to accomplish. Graph G in this example gets the unique syntactic history in figure 5.7. We obtain the semantics of graph G from the revised semantics as follows:

Given model M, G is true in M

iff G_7 is true in M_a^x for some a in the domain — by 4

iff G_6 is true in M_{ab}^{xy} for some a and b in the domain — by 4

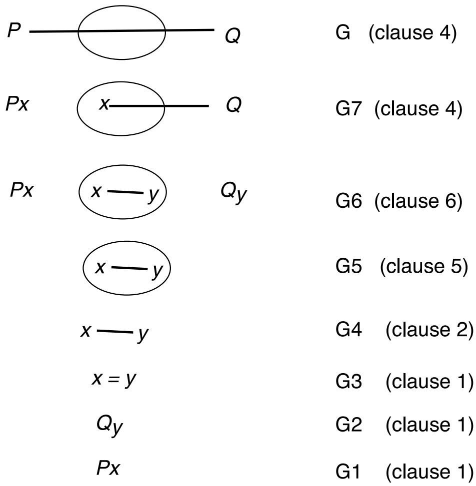


Figure 5.7
The unique syntactic history of graph *G*.

iff G_1, G_5, and G_2 are true in M^{xy}_{ab} for some a and b in the domain — by 6

iff G_1 and G_2 are true and G_4 is false in M^{xy}_{ab} for some a and b in the domain — by 5

iff G_1 and G_2 are true and G_3 is false in M^{xy}_{ab} for some a and b in the domain — by 2

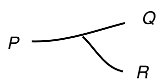
iff $I(x) \in I(P)$, $I(y) \in I(Q)$, and $I(x) \neq I(y)$ in M^{xy}_{ab} for some a and b in the domain — by 1

iff for objects a and b in the domain, $a \in I(P)$, $b \in I(Q)$, and $a \neq b$

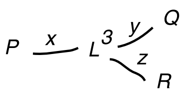
Note that there is a one-to-one correspondence between the syntax of *G* and its semantics.

Now we move to evaluating the project of giving a direct semantics to Beta graphs. The similarity between Zeman's reading and this direct semantics is striking. First, both approaches decompose a Beta graph by replacing lines of identity with variables. Second, every identity line is interpreted as an existential quantifier in both methods. Third, Peirce's endoporeutic reading is adopted by Hammer as well as by Zeman. Therefore, two of the criticisms made against Zeman's reading in §5.1.1 of this chapter apply to Hammer's method as well. One is that the visuality of identity lines disappears, and the other is that Peirce's visual distinction between existential and universal quantifiers is not used at all. On the other hand, Hammer's method is not as rigorous as Zeman's. In the case of branching lines, Hammer's algorithm for retracting lines with variables does not seem to be as clear as Zeman's.

Let us compare Zeman's and Hammer's analyses for the following graph (call it G):



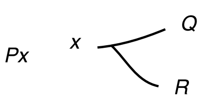
Zeman's algorithm clearly gets us the following result:



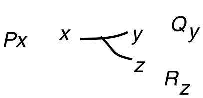
This result is read '$\exists x \exists y \exists z(Px \wedge Qy \wedge Rz \wedge x = y \wedge y = z)$'. We found the result undesirable, since it turns a single network of lines into three different variables. Hammer's semantics does exactly the same thing, but in a more cumbersome manner. For the semantics of graph G, we need to get $G(x)$ first. The following is Hammer's explanation:

> Let G be a graph having a portion of a line of identity l that is enclosed by no cuts, and let $G(x)$ result from G by breaking l at any part enclosed by no cuts and then retracting each of the halves with a new variable (the same variable for each half).[112]

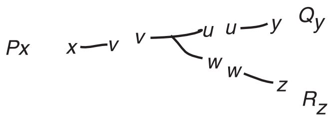
Let's apply this process to the line connecting P and Q in graph G. Then the following result is $G(x)$:



By continuing to apply this method, we get $G(x)(y)(z)$:



In this way Hammer also turns the network of lines into three different variables, x, y, z. Furthermore, nothing can prevent one from analyzing this last graph of Hammer's into the following awkward looking graph, that is, $G(x)(y)(z)(u)(v)(w)$:



The difference between Hammer's result and Zeman's result is not substantial, except that the reader finds Hammer's algorithm of retracting lines with variables even less intuitive than Zeman's algorithm of breaking a line and assigning a temporary predicate.

Hammer's project can be called a *direct* semantics because it assigns the semantics to Beta graphs directly, not through first-order sentences. However, as we have seen so far, this directness does not provide us with any substantial increase in understanding graphs. I do not think that a detour through a first-order reading is a major problem for understanding Beta graphs. But we should find a way to implement the visual benefits of the system so that we may read off Beta graphs more easily in a first-order language or in any existing language. After all, what Hammer actually did under the name of direct semantics is to translate Zeman's reading into a Tarskian semantics and connect the latter with a graph.[113] This is not an actual gain at all. If we could take full advantage of visuality (especially those three aspects that Peirce believed that EG possesses) to find a more efficient way of obtaining a first-order reading, existing Tarskian semantics would take care of the rest.

6

Logical System versus Calculus

My project on EG had two different goals, one theoretical and the other practical. One was to articulate the philosophical framework that the inventor of EG had in mind. The other was to explore how the graphs of EG have been traditionally used and to suggest a better method of using them. These latter practical suggestions for the use of EG were based on Peirce's general insights into heterogeneous systems. Now the question becomes why Peirce and Peircean scholars failed to apply Peirce's original intuitions about EG in its actual use.

A main reason for this neglect of the original intuitions, I believe, is that all Peircean scholars have clung to Peirce's endoporeutic reading, which allows only one way of reading off graphs. As seen in chapters 4 and 5, as long as this method is observed, there is not much room for other visual features to be read off.[1]

Clinging to the endoporeutic reading seems excusable for those who adopted reading instructions from Peirce and chose to dismiss other discussions by Peirce. But how about Peirce himself? Didn't he realize that his endoporeutic interpretation would make the visual distinctions he emphasized for Beta graphs unnecessary? As shown in the first half of the book, Peirce himself did not have any prejudice against the use of icons in a logical system, and he was perfectly aware that EG is not purely symbolic. Nevertheless, he did not fully take advantage of the iconic elements of EG. The reader, if convinced by my arguments in chapters 2 and 3, must wonder why Peirce's rich theoretical framework for heterogeneous systems was not fully implemented in his understanding of his own invention EG.

Thus, we raise the following historically interesting question: Why did Peirce himself, who was fully aware of the iconic advantages EG has, not exploit the visual features discovered in this book? The following passage of Peirce's, where he states the goal of a logical system, is one of the best places to start:

> [The purpose and end of a system of logical symbols] is simply and solely the investigation of the theory of logic, and not at all the construction of a calculus to aid the drawing inferences.[2]

A logical system is designed for the investigation of the theory of logic, while a calculus is for making inferences. This distinction is somewhat puzzling to us, since many of us do not hesitate to call some deductive logical systems deductive calculi. Logical systems are widely used for making inferences, we believe. Isn't it the case that some logical systems pursue the two goals—to investigate logical theories and to make inferences—at the same time? According to Peirce, these two goals are so different from each other that they cannot be pursued by one and the same system. That is, they are "incompatible," for the following reason:

> These two purposes [to investigate logical theories and to aid the drawing of inferences] are incompatible, for the reason that the system devised for the investigation of logic should be as analytical as possible, breaking up inferences into the greatest possible number of steps, and exhibiting them under the most general categories possible; while a calculus would aim, on the contrary, to reduce the number of processes as much as possible, and to specialize the symbols so as to adapt them to special kinds of inference.[3]

In exploring logical theories, we are interested in theoretical aspects of inference, that is, *why* an inference step is valid, rather than in its application, that is, *how* we make inferences. For the theoretical aspects of inference, we need to go into the details of inferences as much as possible. The more an inference step is analyzed, the less likely we are to make an error about that inference, since all the details are scrutinized so that we may obtain a full and clear justification for the step. Thus, Peirce explicitly defines "the business of logic to be analysis and theory of reasoning, but not the practice of it."[4]

On the other hand, a calculus is mainly invented for application, according to Peirce. In the practice of reasoning, we want to get to the conclusion by way of as few steps as possible, as long as the validity of the steps has previously been proven. The fewer steps a calculus allows us to take, the easier the system is to use. As a contrast with the business of logic, Peirce says the following:

> I consider that the business of drawing demonstrative conclusions from assumed premises, in cases so difficult as to call for the services of a specialist, is the sole business of the mathematicians.[5]

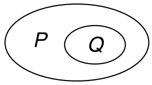
I argue that Peirce's distinction between the goal of a logical system and the goal a calculus answers our question raised above: Why did Peirce himself fail to make his own system, EG, more efficacious? Since according to Peirce the purpose of logic is to theorize about reasoning (as opposed to putting it into use), a logical system is designed to find out what has been going on in reasoning (as opposed to teaching or helping us to reason). Therefore, the designer of a logical system is not interested in answering whether the user can use the system in an easier or clearer way. Efficiency is not important for a logical system, since efficacy becomes important only when we care about the practical use of a system.

Related to the efficiency issue, the following passage from Peirce tells us a more specific contrast found in a logical systems and a calculus:

> It should be recognized as a defect of a system intended for logical study that it has two ways of expressing the same fact, or any superfluity of symbols, although it would not be a serious defect for a calculus to have two ways of expressing a fact.[6]

Let me illustrate the point of this passage through propositional languages. For example, the sentences '$\neg P \vee Q$', '$P \rightarrow Q$', and '$\neg(P \wedge \neg Q)$' all express the same fact, which is what Peirce means by a "superfluity of symbols." As long as a system is truth-functionally complete, Peirce would recommend a logical system with fewer logical connectives. Hence, having two connectives, for example, $\neg$ and $\vee$, would be better than having five connectives, $\neg$, $\vee$, $\wedge$, $\rightarrow$, and $\leftrightarrow$. The main reason behind this choice is directly related to the goal of a logical system that Peirce emphasized above: to investigate logical theories, as opposed to aiding the practice of reasoning. Having five connectives makes our life easier but is redundant from a theoretical point of view. For a system used as a calculus, Peirce would choose a system with five connectives, but not for one used as a logical system. While expressive power remains the same, if a system has more connectives, more metatheorems need to be proven. This view is not alien to us. Many logic textbooks that focus on metatheories of first-order logic adopt not a language with all of these five connectives but rather a language with one of the smallest truth-functionally complete sets of connectives.

In the case of EG, the Alpha system presents only the following diagram[7] for the fact expressed by the three different sentences '$\neg P \vee Q$', '$P \rightarrow Q$', and '$\neg(P \wedge \neg Q)$':



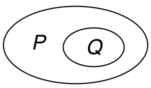
As explained in chapter 3, a cut in the Alpha system represents negation, and a juxtaposition represents conjunction. There seems to be no iconic sign which corresponds to '$\vee$', '$\rightarrow$', or '$\leftrightarrow$'. Traditionally, this has been an obstacle to the popular use of EG, since the Alpha system was treated as equivalent to a propositional language with only two kinds of connectives. However, as made clear in the above quotation, according to Peirce, the aspect of having fewer connectives makes the Alpha system a more desirable logical system than a propositional language with five connectives. This situation is analogous to the case where a logician would choose a language with only '$\neg$' and '$\wedge$' (rather than a language with more connectives) to explore metatheories of propositional logic.

Peirce's reason why he was not interested in a more useful and clearer symmetry of EG's transformation rules also stems from his distinction between logical systems and calculi:

> It is true that if our purpose were to make a calculus, the two operations, aggregation and composition, would go admirably together. Symmetry in a calculus is a great point, and always involves superfluity, as in homogeneous coordinates and in quaternions. Superfluities which bring symmetry are immense economies in a calculus. But for our purposes of analysis they are great evils.[8]

Now it becomes clear why Peirce himself did not come up with the reading algorithms and the transformation rules presented in chapters 4 and 5 which are more natural and more efficient than the traditional ones. Peirce was not interested in *adopting* EG as an aid for the practice of reasoning. That is, he did not consider it a deductive calculus at all. Hence, there was no need to explore easier or more natural reading algorithms for these graphs. There was no incentive to make EG more efficacious, either.

As discussed in §4.3.1, Peirce suggested '$P \rightarrow Q$' as a reading of the following diagram, which is an alternative to the endoporeutic reading.



However, Peirce never developed the scroll reading fully enough to systemize it. In the case of the Beta system, Peirce made a clear visual distinction between universal and existential quantifiers.[9] However, he did not fully incorporate this distinction into a comprehensive reading algorithm. Nor has any Peircean scholar. Things are quite similar when it comes to transformation rules. Peirce classified transformations into insertion and erasure, and placed each transformation in one of these two categories, ignoring finer distinctions. So have Peircean scholars.

However, one main difference between Peirce and others is that Peirce did not intend to utilize EG as a deductive calculus but others did. All Peirce wanted to do with this system is to explore logical concepts and theories. On the other hand, logicians have complained against EG mainly because of the difficulty in its practical use as a deductive system. Peirce would not have cared about these criticisms. He would have responded to them by pointing out that EG was not invented for drawing inferences. Therefore, we should not be surprised that Peirce kept to the endoporeutic method (even though it yields quite cumbersome readings) and that he stated his inference rules in a rather awkward and inefficient way.

After understanding why Peirce did not make EG more efficacious, we naturally would like to ask the following question: Are the goals of a logical system (to investigate logical theories) and a calculus (to draw inferences) incompatible with each other, as Peirce says? If this answer is yes, as Peirce believed, then for a given system we should choose to use it either for a logical system or for a calculus, but not for both. There is indeed a grain of truth in Peirce's intuition about this matter. A comparison between axiomatic systems and natural deductive systems nicely illustrates this point. An axiomatic system (with more axioms and with fewer inference rules) is adopted to explore metatheories, while a natural deductive system (with fewer or no axioms and with more rules) is used to draw conclusions from premises. A choice among different first-order languages is determined by our purpose. A language with more connectives or quantifiers is chosen as an aid for the practice of reasoning, while a language with fewer connectives or quantifiers is adopted for the analysis of reasoning. That is, there seems to be a trade-off between the suitability of a language as a logical system and as a calculus.

Is this trade-off necessary regardless of types of representation systems? I claim that we do *not* have to trade one function for the other in the case of EG and that this is a great advantage graphical systems have over symbolic ones. One and the same system, if a graphical one, could be both a desirable logical system and an efficacious deductive calculus.

In a symbolic propositional language, when we have only two connectives, for example, $\neg$ and $\vee$, our expressions sometimes become too cumbersome to read off easily. This is why a deductive calculus has other connectives, i.e., $\wedge$, $\rightarrow$, and $\leftrightarrow$. The increase in logical connectives does not increase its expressive power but improves its practicality. At the same time, more metatheorems are needed for the justification of these new connectives. So when we are concerned about metatheories of logic, there is no need for this theoretically redundant vocabulary.

Things are quite different in the case of EG. Let us recall how I made EG more efficacious in chapters 4 and 5. When my new reading algorithms and new formulation of transformation rules were presented, no new vocabulary was introduced. The increase in practicality did not come from any increase in either its iconic or symbolic notation, unlike with symbolic languages. According to Peirce's terminology quoted above, no "superfluity of symbols," which is a defect for logical systems, is created. Hence, no additional metatheorems are required.

How, then, is it possible for a language to become easier to understand or to use without any new vocabulary being added? In the case of symbolic languages, this seems to be impossible; there is no more than one way of reading off a linear expression such as a sentence. But in the case of EG, we have discovered that there are multiple ways of reading off one and the same graph. In principle, Peirce would not have any objection to this discovery. Peirce wanted to prevent a logical system having multiple expressions for the same fact,[10] but did *not* seek to prevent multiple interpretations for the same expression as long as they do not create ambiguity. When the flexibility for multiple readings is discovered and implemented in the interpretations of graphs, possibilities open up to reconsider and reformulate inference rules for more efficacious systems. When we fully implement visual distinctions present in graphs, the system becomes not only more intuitive and useful but also more efficacious.

These newly interpreted and restated systems obtain a practical advantage, as illustrated in §4.4.4, §4.5, and §5.3. At the same time, these improved systems did not lose their suitability for investigating logical theories, as Peirce intended for the original systems. Since no new syntactic devices were introduced, no new metatheorems involving basic vocabulary are needed. Also, we did not increase the number of transformation rules but rearranged them with more specific symmetries based on visual distinctions. Again, no new metatheorem is generated at this level either. The trade-off between an axiomatic system and a natural deductive system does not apply here. Our goal to make EG more intuitive, useful, and efficacious is achieved without its power as a logical system being diminished. Granting Peirce's distinction between the purpose of logical systems and the purpose of calculi, we presented a way to satisfy these two different goals at the same time. I do not see how Peirce would object to this achievement.

Why did Peirce not realize that the two different goals he identified for a logical system and a calculus are compatible in his own system? At this point, we can only speculate. Peirce was not fully aware of every possible difference among different kinds of representation, in spite of his theory of signs. Unlike with a sequence of symbols, when we carve up one and the same graph in more than one way, this does *not* make the system ambiguous. This difference does not seem to have occurred to the inventor of the first comprehensive and logically powerful graphical system, EG.

Of course, we do not hold Peirce accountable for this neglect. After a long, almost a century's, prejudice against non-symbolic representation, we, at the end of the twentieth century, might be in a better position to see the differences more sharply than did Peirce, who did not face any bias for or against one kind of representation.

7 Conclusion

Peirce invented the system of Existential Graphs as a logical system. I identified two main theoretical bases behind the invention: One is Peirce's theory of signs and representation, and the other is Peirce's distinction between a logical system and a calculus. These unique assumptions about representation and logical systems played a key role not only in how EG was created but also in how it has been understood. Peirce's insightful theory of signs and representation was a main source for the birth of this impressive non-symbolic system, while Peirce's intention to make EG a logical system (not a calculus) prevented Peirce himself from exploiting a better way to understand his own system. Ironically, Peirce's rather limited approach to EG has been inherited by Peircean scholars, but his rich philosophical motivation behind EG has been neglected in the course of understanding the system. My work sought to resolve this irony in the spirit of contemporary research on heterogenous reasoning.

According to Peirce's theory of signs and representation, there is no intrinsic reason why the formalization of symbols, but not the formalization of icons, can or should be pursued. Moreover, the mixture of symbols and icons is not only theoretically possible, but more desirable than the "purity" of homogeneous systems, since each kind of symbol has its own strengths and weaknesses. Our examination of EG in terms of both symbolicity and iconicity supports my claim that EG is a heterogeneous system which aims to take advantage of both kinds of signs, symbols and icons. This claim not only challenges the well-accepted view that EG is a diagrammatic system but also urges us to approach EG from two different aspects, its symbolicity and its iconicity. Otherwise, we would fail to take full advantage of the heterogeneity of the system. As we saw in chapters 4 and 5, this has been one of the main mistakes shared by every existing reading method and formulation of the inference rules of EG, including Peirce's.

Examining the traditional reading methods and the existing transformation rules for EG led us to conclude that EG has been understood and used as if it were a symbolic system. Why is this so? Three related facts are responsible for this somewhat surprising discovery. One is the general prejudice for symbolic systems, which forces us to understand and evaluate a non-symbolic system in terms of the criteria for symbolic systems rather than recognizing the non-symbolic system's own strength.[1] A second reason for the misunderstanding of EG as a symbolic language, in spite of the principle behind the birth of EG, is that Peirce's theory of signs and representation was not studied in the context of research on heterogeneous systems, but only in the context of semiotics. Making a convincing connection between Peirce's theory of signs and his own product probably requires a strong interest in heterogeneous systems, which has taken place in various areas only recently.

But how about Peirce himself? Why did Peirce himself fail to utilize the iconic features unique to this graphical system? Peirce's distinction between a logical system and a calculus is the third reason that EG has been understood in terms of symbolic systems. EG was supposed to serve as a logical system, not as a calculus, and no system could be both a good logical system and a good calculus, according to Peirce. However, an interesting twist to the story is that Peirce's distinction between logical system and calculus is only valid for symbolic languages but not for graphical languages. The inventor of one of the most impressive heterogeneous systems applied to graphical languages what is true only of symbolic languages.

With these three sources of misunderstanding in mind, I re-analyzed both the Alpha and Beta systems of EG and presented new reading algorithms and new sets of transformation rules, which make EG not only more accessible but also more efficacious. This new approach fully incorporates previously ignored iconic elements and provides a *natural* way to read off graphs and a *natural* way to manipulate graphs, but without losing its status as a logical system.

Reconstructing Peirce's philosophy of logical notations and making complete use of iconic aspects in EG have enriched our understanding of the system, and we are closer to responding to Quine's following evaluation of Peirce's EG:

> One questions the efficacy of Peirce's diagrams, however, in their analytical capacity as well. Their basic machinery is too complex to allow one much satisfaction in analyzing propositional structure into terms of that machinery. While it is not inconceivable that advances in the diagrammatic method might open possibilities of analysis superior to those afforded by the algebraic method, yet an examination of Peirce's product tends rather, apagogically as it were, to confirm one's faith in the algebraic approach.[2]

In particular, the difficulty with disjunction in Peirce's system[3] has led some to doubt the efficacy of graphical methods in general. Surely, this is a false conclusion.[4] When one does not use the basic machinery of a system fully, it is not surprising that one might not be able to obtain what could be achieved quite readily. But when one recognizes additional visual features of the system, the difficulties perceived by Quine can be reduced. Furthermore, as we have seen in the last section of chapter 4, for certain purposes this graphical system is more efficient than symbolic languages.

We distinguish among different symbolic logical systems according to the purpose at hand. An axiomatic system is chosen for investigating logical theories, and a natural deductive system for deduction. Similarly, if we can articulate the differences between symbolic and graphical systems, we may sometimes choose one over the other, depending on our purpose.

All this suggests that our investigation of graphical systems should go further than proving the logical equivalence between symbolic and graphical systems. This is one of the main reasons why interdisciplinary work is crucial for further fruitful results in research on heterogeneous systems. As explained in the Introduction, logicians, cognitive scientists, philosophers, AI researchers, and design theorists approach this topic from different perspectives and have produced important and fascinating results. Now the time has come to relate the different approaches and interests of various groups of researchers, and the work presented in this book illustrates one of the many directions we could take for further investigation of graphical representation as interdisciplinary research.

It has not been my purpose to argue that EG is better than first-order languages. On the contrary, an absolute comparison among different representation systems seems quite misguided. What I have presented in this limited space is a case study in understanding one graphical system on the basis of its own strengths, not on the basis of the criteria used for symbolic systems. I hope this approach will inspire others not only to find further visual aspects of Peirce's EG[5] and other graphical systems, but also to find more concrete differences between graphical and symbolic systems. In this way, we will be able to identify different strengths of the different kinds of representation systems so that we may take full advantage of the differences either in understanding existing heterogeneous systems or in inventing new systems.

Notes

Chapter 1

1. How this interdisciplinary project has been formed will be discussed in a later part of this chapter. See the website Diagrammatic Reasoning Site, designed by Michael Anderson (http://www.hcrc.ed.ac.uk/gal/Diagrams/) for current conferences, books, journals, and research sites.

2. Several pieces of evidence for Peirce's strong preference of EG to his own algebras of logic are available. First of all, Peirce invented EG not just as another form of logical system but as an improved version of his previous symbolic logic: "[A thorough understanding of mathematical reasoning], then, is the purpose for which my logical algebras were designed, but which, in my opinion, they do not sufficiently fulfill. The present system of existential graphs is far more perfect in that respect" (Pierce, *CP*, 4.429). As Roberts cites from Peirce's letter to William James, Peirce believed that EG "ought to be the logic of the future" (Ms L 224, cited in Roberts 1973, p. 11). Also, it is well known that Peirce gave the subtitle "My *chef d'oeuvre*" to the paper on EG.

3. Chandrasekaran, Glasgow, and Narayanan 1995, p. xvii.

4. For example, Stenning and Yule's project relies on this relation: "This paper applies semantic and computational analysis to the apparently contrasting *external* representations which figure in psychological theories about the *internal* representations engaged in syllogistic reasoning" (Stenning and Yule, p. 110.)

5. Johnson-Laird's development of Mental Models theory is a good example in which a distinction between these two levels of representation is not necessary.

6. Block 1981 is one of the best collections of important papers on this debate, and Block 1983 presents a succinct summary of this controversy and raises insightful philosophical questions about the debate. Chapters 1–4 of Tye 1991 is a good overview of both cognitive scientists' and philosophers' various positions on this issue.

7. Kosslyn 1980a, 1980b; Shepard and Metzler 1971; Fodor 1981; Kosslyn, Pinker, Smith and Schwartz 1981; and Finke and Pinker 1982.

8. Pylyshyn 1973, 1981a, 1981b; Dennett 1981; Shorter 1952; and Sterelny 1986.

9. The phrase 'mental images' has been quite ambiguous throughout the literature. It sometimes means any form of mental representation. In this sense, both pictorialists and descriptionalists agree that there are mental images, but disagree that there are picture-like mental images. However, since the word 'image' has been closely related to a picture-like form in our ordinary usage, some have equated images with visual picture-like images. Hence, some descriptionalists claim that there are no mental images. Also, some pictorialists discuss the role of images, assuming that these images are picture-like. From now on, I will use 'images' or 'mental images' in this latter sense, unless I need to make a distinction between visual images and descriptional images.

10. See Aristotle *On the Soul* and *On the Memory and Recollection*.

11. I borrowed this phrase from Hacking 1975. This mainly refers to the seventeenth century.

12. Smart 1959.

13. See Block 1983, Dennett 1981, and Fodor 1981. Also, Cummins 1996, chapter 9, touches on this issue from the philosopher's point of view.

14. In the middle of the 1960s, interest in images revived in psychology. See Holt 1964.

15. Haber and Bower 1970a and 1970b.

16. Oatley 1977.

17. Huttenlocher and Stenning 1996.

18. Even though most of these works assume that there are mental images (that is, they accept the pictorialists' claim), strictly speaking they do not have to commit themselves to the view that these images exist as basic units in our cognition. Not surprisingly, various works on the functions of images have been cited to support the pictorialists' main point. Descriptionalists, on the other hand, do not have to discard the discussions of the functions of images, but only need to add that these images are not primitive units stored in our memory but formed out of structured descriptions more like the sentences of a language. (See Pylyshyn 1979.) However, it has been the case that those working on the roles of images in cognition usually belong to the pictorialists' camp, even though their main interest is slightly different.

19. Lindsay's "Images and inference" (1988) is a good example for this. As the title indicates, this is about a distinct function of images in our inferences. Without paying much attention to the distinction between internal and external representations, Lindsay, at the beginning of the paper, relates images to the context of memory, which clearly belongs to internal representation, but his main points are made about external diagrammatic representations (which he persists in calling 'images'). Some might find a mixture between internal and external representation in the discussion of diagrams in reasoning in Glasgow and Papadias 1992.

20. Larkin and Simon 1987, p. 69.

21. A nice illustration: In Simon 1978, the author devotes a majority of the paper to discussing whether there is more than one kind of mental representation, but as mentioned above, in Larkin and Simon 1987 (where a distinct property of externally drawn diagrams is discussed) there is no need of discussion to convince the reader of the existence of diagrams.

22. Recall that a contrast between picture-like mental images and linguistic descriptions is a main issue in the imagery debate as well.

23. Simon 1978.

24. He refers to it as 'images'.

25. Lindsay calls this way of reading off a conclusion 'non-deductive'.

26. Refer to Sloman 1971, 1985, 1995.

27. Reasoning with Diagrammatic Representations: 1992 AAAI Spring Symposium; Cognitive and Computational Models of Spatial Representation: 1996 AAAI Spring Symposium; Reasoning with Diagrammatic Representations II: 1997 AAAI Fall Symposium; and Formalizing Reasoning with Visual and Diagrammatic Representations: 1998 AAAI Fall Symposium. Also refer to Narayanan 1993.

28. The following conferences are good evidence for this effort: VISUAL '98: Visualization Issues in Formal Methods (Lisbon); International Roundtable Conference on Visual and Spatial Reasoning in Design (MIT, 1999); Theories of Visual Languages—Track of VL '99: 1999 IEEE Symposium on Visual Languages.

29. Shin 1994.

30. Zeman 1964, Roberts 1973, and Sowa 1984.

31. Barwise and Etchemendy 1995, p. 214.

32. Barwise and Etchemendy 1994.

33. For example, Larkin and Simon's (1987) diagram of a pulley problem.

34. Sowa's famous application of EG in knowledge representation is an excellent example. Refer to Sowa 1994, 2000.

35. To borrow Simon's (1978) phrases, these two systems are informationally equivalent, but not computationally equivalent.

36. I would like to point out another important benefit from choosing Peirce's EG as my case study. That is our familiarity with EG and the quality and depth of the existing work about technical aspects of the system. Thanks to pioneering work by Don Roberts, Jay Zeman, and John Sowa, graphical notations and the meanings of the graphs are well established. The soundness and the completeness of the system have been proven. Most importantly, EG has been proven to be equivalent to a first-order symbolic system. Therefore, rather than starting from scratch, I can take advantage of these results to explore my issues.

37. Most of them are discussed in the context of semiotics.

38. As we will see in chapters 4 and 5, the traditional ways of understanding EG originate from Peirce.

Chapter 2

1. For Frege's iconic script, refer to Frege's *Begriffsschrift* (1967 [1879]) or Kneale and Kneale 1962, chapter 8.

2. In this article, van Heijenoort aims to clarify Frege's statement "[U]nlike Boole's, his logic is not a *calculus ratiocinator*, or not merely a *calculus ratiocinator*, but a *lingua characterica*" (van Heijenoort, p. 324).

3. Sluga raises a question about this relation, pointing out that "neither Boole nor Schröder engaged in metamathematical investigations" (1987, p. 83).

4. Van Heijenoort 1967, p. 325.

5. Again, Sluga (1987) claims Frege's and Schröder's debate over interpretations of logical symbolism is *not* about different domains of individuals but about whether one and the same logical notation can be used for arithmetic, for propositional logic, and for class logic. And the main reason why Frege was against the multiple interpretations of the algebraic notation is, according to Sluga, Frege's goal to unite all of these three—algebra, propositional logic and class logic—in a single theory.

6. Van Heijenoort 1967, p. 326.

7. Goldfarb 1979, p. 352.

8. Tappenden, in his impressive work "Metatheory and mathematical practice in Frege" (1997), objects to this rather well-circulated view that Frege's work does not allow us to do metatheory. (Refer to the numerous citations from many different philosophers in Tappenden 1997, pp. 220–221.) He argues that this interpretation of Frege's position originates from a hasty equation of the metatheory of logic and Tarski's model theory. If this equation is assumed, what Frege does is not metatheoretic. But Tappenden claims that this way of understanding 'metatheoretic approach' is too narrow to be interesting and that Frege's work is compatible with metatheoretic investigation when a proper understanding of "metatheoretic" method is explored.

9. Goldfarb 1979, p. 356.

10. Goldfarb 1979, p. 354. (Italics are mine.)

11. Goldfarb (1979) suggests one important aspect of the tradition of a *calculus ratiocinator* that van Heijenoort did not. That is, the calculus approach lacks the notion of formal proof. Therefore, "[t]o arrive at metamathematics from the algebra of logic [the tradition described in the previous quotation] we must add the "mathematics," that is, an accurate appreciation of how the system may be used to encode mathematics, and hence of how our metasystematic analyses can be taken to be about mathematics" (Goldfarb 1979, p. 356). Since this aspect is not directly related to our main discussion, I will not go into the details.

12. Goldfarb (1979) would hesitate to use the term 'model-theoretic' to refer to Boole's logic, since "it is misleading to speak as if Löwenheim were in full possession of the model theory of first-order logic: what is primarily missing is a full sense of the role of the object language in formalization of mathematics" (1979, p. 355).

13. Hintikka 1997, p. 16.

14. Hintikka 1997, p. 24.

15. Dipert 1995, p. 34.

16. Dipert 1995, p. 35.

17. Dipert 1995, p. 35.

18. In the later part of the paper, Dipert uses the word 'formal' in an important way, but in a different context: "[I]f we understand 'formal' in the sense of a multiply interpretable ('uninterpreted') symbolic calculus,..." (Dipert 1995, p. 44). However, this use of 'formal' is rather close to the model-theoretic approach I discussed in the previous subsection, but is not related to the current discussion.

19. Dipert 1995, p. 35.

20. Even though Dipert seems to suggest that De Morgan knows "what it is to be usefully formal" better than we do, it is far from being clear what De Morgan meant by 'formal' in *Formal Logic*. De Morgan emphasizes the importance of 'the structure of a sentence' as the essence of logical *form* (1847, chapter 1). However, some could take DeMorgan's emphasis on logical form as support for symbolic formalization (as opposed to non-symbolic formalization), as saying that the structure of a sentence is represented more easily in a symbolic language rather than in a graphical language.

21. Peirce, *NEM*, 4:314. We can find a very similar passage: "The first things I found out were that all mathematical reasoning is diagrammatic and that all necessary reasoning is mathematical reasoning, no matter how simple it may be" (Peirce, *NEM*, 4:47).

22. Peirce, *NEM*, 4:47–48.

23. Note that Peirce explicitly mentions "constructing a diagram" in the quotation above.

24. Peirce, *NEM*, 4:44.

25. Peirce's following remarks confirm this conclusion: "In order to expound my proposition that all necessary reasoning is diagrammatic, it is requisite that I explain exactly what I mean by a Diagram, a word which I employ in a wider sense than is usual" (Peirce, *NEM*, 4:315, n. 1). We will see more specific quotations to support this widely accepted view. On the other hand, there are passages which seem to indicate that Peirce's use of 'diagram' is not so different from ours. For example, "A *diagram* is an *icon* or schematic image embodying the meaning of a general predicate" (Peirce, *NEM*, 4:238). However, the evidence for Peirce's broader use of 'diagram' is overwhelming. Dipert correctly

points out that "[Peirce] does not use the key term 'diagram' with perfect consistency" (Dipert 1996, p. 391).

26. The above quotation is an example. Another example is, "In order to show that this inference [an example of deductive inference] is (or that it is not) absolutely necessary, it is requisite to have something analogous to a diagram" (Peirce, *CP*, 3.418).

27. Ketner 1985, p. 408.

28. Peirce, *CP*, 3.418.

29. The diagram Peirce is using as an example is supposed to infer certain facts about two provinces. (This note is mine.)

30. Peirce, *CP*, 3.418.

31. This is so even though it is not clear whether Peirce himself made a distinction between the syntax and semantics of a diagram. We find Peirce's lack of a clear distinction between syntax and semantics when Peirce presents the rules of transformation rules for Euler [Venn] diagrams. For more details, refer to Shin 1994, pp. 24–40. However, Müller (forthcoming) argues that there was no confusion between syntax and semantics in EG's Gamma system.

32. Peirce, *CP*, 4.530.

33. Peirce, *CP*, 3.419.

34. Peirce, *CP*, 4.533.

35. Peirce, *CP*, 2.228.

36. Peirce calls them *interpretant*, *objects*, and *ground*, respectively. Refer to Peirce, *CP*, 2.228. There is a difference between the sign itself and the interpretant, but for the currentt discussion it can be ignored.

37. Peirce, *CP*, 2.229.

38. When Peirce takes up this question seriously, he immediately notices that the representation of a language cannot be defined in that language itself: "Such a figure [a diagram or diagrammatoidal figure] cannot, however, show what it is to which it is intended to be applied" (Peirce, *CP*, 3.419). It is important to note that this statement does not contradict Hintikka's classification of Peirce's logical theory as being within the model-theoretic tradition (discussed in §2.1.1). If Peirce acknowledges any attempt to talk about representation, we can infer that he admits a meta-language level. It is not clear whether Peirce believed that we can talk about the representation of a natural language. On the other hand, it seems quite clear that he does not deny this effort for any formal logical language but distinguishes between representing and represented facts. In this respect, Peirce's view is not so different from Tarski's position toward natural and formal languages.

39. Peirce, *CP*, 3.419.

40. As is well known, Peirce's theory of signs is notoriously complicated. It starts with the sentence "We have seen ... that every sign has these three elements:

First, the qualities which belong to it in itself as an object; second, the character of addressing itself to a mind; and thirdly, a causal connection with the thing it signifies" (*CE*, 3:77). Again, each of these three elements has its own trichotomy. Among these trichotomies, the second trichotomy, that is, the division of signs according to the relation between the sign and the object, is related to this discussion, and hence this will be the only classification I will discuss below. Much research has been undertaken on Peirce's theory of sings which covers all of these divisions. See Weiss and Burks 1945, Sanders 1970, and Olsen, forthcoming.

41. Peirce, *CE*, 3:65. This is only one of the three trichotomies Peirce classified, but it is the one in which we are interested.

42. For basic distinctions among these three, see Peirce, *CP*, 3.359–362. In some of his writings, Peirce uses the term *token* instead of *symbol*. His paper "On the algebra of logic" (1884) is an example. Later, he consistently adopts the word *symbol* (refer to "Prolegomena to an apology for pragmaticism" in *The Monist* vol. 16 [1906], and Peirce, *CP*, 2.227–307.) Especially when Peirce discusses types versus tokens, the choice of *token* as one kind of a sign could cause some confusion. Therefore, I think the word *symbol* is a better choice than the word *token*. Also, when I discuss symbolic versus graphical systems later, the term *symbol* will turn out to be more appropriate for my purpose.

43. As said above, this is one of the three trichotomies of signs.

44. These differences, as we examine them one by one, are directly related to many important topics in Peirce's own philosophy of mathematics and logic that will need to be discussed soon.

45. Peirce's complicated taxonomy of signs allows for gray areas and many mixtures among the three different kinds of signs.

46. We cannot absolutely rule out a conventional aspect of English indexicals. After all, some words are indexicals, according to English convention. However, the way we pick out an object by using an indexical is different from the way we understand the meaning of the word 'book'. In the case of an indexical, we need more than English conventions for it to complete its mission.

47. As is well known, different theories of proper names give different answers to the question of whether proper names have any descriptive meaning.

48. See Goudge 1965.

49. Peirce, *CP*, 2.286.

50. Numerous passages are found in Peirce's writings about icons. For some of those passages, see Dipert 1996, pp. 388–389.

51. Peirce is rather ambiguous about the status of photographs. There is a causal relation between what Tom looks like and his picture. In this sense, a picture has an element of an indexical function as smoke is an index of fire. (This was pointed out by Keith Stenning.) However, the important difference is that in the case of smoke/fire, understanding a causal relation is necessary to know that

smoke indicates that there is fire, while in the case of picture/person, we can tell whom this picture represents without realizing the causal relation but only by resemblance. Therefore, there is an iconic element, which pure indices do not possess.

52. Peirce, *CP*, 2.280.

53. Peirce, *CE*, 3:65.

54. Some might say that this relation is not as arbitrary as the word 'rain' is to rain, since this symbol emerges from our knowledge that doves are peaceful birds. However, a dove cannot be said to be an icon of peace, since a bird cannot resemble peace.

55. Goodman 1976, pp. 3–5. Dipert (1996) tracks down the root of Goodman's fever for conventions and presents a substantial criticism of Goodman's attack against resemblance.

56. Dipert 1996, p. 381.

57. Hence, resemblance among representing facts is not in question here. Neither is resemblance among represented facts.

58. Peirce, *CP*, 2.281.

59. Peirce, *CP*, 2.282.

60. Peirce, *CP*, 2.282.

61. According to Dipert, Peirce "notices that using circles to stand for entities with certain properties is perfectly conventional (symbolic)" (Dipert 1996, p. 389).

62. Photographs are good examples to show that not all visual representations are icons.

63. For more elaborate discussions of this topic, refer to Shin 1994, pp. 168–173.

64. The resemblance between being in set *A* (an abstract relation) and being inside a circle (a visually observable relation) is a good example for the 'direct' semantics of diagrammatic languages argued for by Gurr, Lee, and Stenning: "A fundamental aspect of diagrammatic languages, which distinguishes them from sentential ones, is that generally the representing relations between diagrammatic tokens are "directly" semantically interpreted. That is to say that, unlike sentential languages, there is typically a direct mapping from some representing relation in a diagram to the relevant semantic relation." (1998, p. 538.) I believe the direct mapping that these authors emphasize is closely related to the characteristic feature of icons—resemblance.

65. The abstract concept of inclusion is represented by the visually observable relation of containment. The latter property resembles the former and hence is an iconic representation. So are overlapping and disjoint relations. This resemble relation is what Dipert had in mind when he stated that "the arrangement of these circles [Euler/Venn Circles] as, for example, overlapping or not, is (in some respects) iconic" (1996, p. 389).

66. No intuitive homomorphism is found in the case of corresponding symbolic representations $\subseteq$, $\cup$, and $\cap$.

67. This is another aspect of representation Goodman's (1976) criticism of resemblance does not address. Goodman seems to assume either a complete resemblance in every aspect or no resemblance at all.

68. Peirce, *NEM*, 4:47–48.

69. It is not clear whether Peirce would agree with this explanation, since he sometimes seems to believe that an icon, especially a pure icon, represents something in particular only. In this context he says "A diagram [of geometry], indeed, so far as it has a general signification, is not a pure icon" (Peirce 1884, p. 181).

70. As mentioned in the previous note, this is what Peirce sometimes seems to believe.

71. For an example and a longer discussion, refer to Shin 1994, chapter 1.

72. Some might say that a triangle on a sheet of paper is only a triangle-like figure which we can draw or erase, while a triangle in reality is an abstract entity.

73. See Lindsay 1988; Larkin and Simon 1987; Shimojima 1996; and Gurr, Lee, and Stenning 1998.

74. "It is always as here [the (false) axiom that the whole is greater than its part] because it tempts him to draw a figure and judge by the looks of it what is part and what whole" (Peirce, *NEM*, 4:44).

75. Peirce, *CP*, 4.433.

76. Peirce, *CP*, 2.279.

77. Lindsay (1988) would find this case as a prime example for "nonproof procedure," which he claims is one of the characteristics that inference using visual images possesses. According to Shimojima (1996), this is a free ride. Gurr, Lee and Stenning (1998) say that this is a cheaper ride than many other kinds of inference we could adopt for reasoning about the relation between sets A and C, for example, manipulating symbolic representation.

78. Peirce 1884, p. 181. As I said earlier in note 42 in this chapter, he uses the word 'token' here, but with the same meaning as 'symbol.'

79. Venn 1971 (1881), p. 510.

80. For more details on this issue, refer to Shin 1994, chapter 2.

81. Some think that shading an area is very similar to erasing the area. So it is an iconic representation of emptiness. But, compared with how Euler represents emptiness (i.e., not having an area), Venn's shading requires a convention. In addition to stating that "this shading is a conventional sign of the nature of a token [symbol]," it seems obvious that Peirce treats shadings as a symbol. When he revises Venn diagrams, he replaces shading with the symbol 'o' (Peirce, *CP*, 4.357–4.360). I have also observed that many students find the meaning of shading rather arbitrary.

82. Peirce 1884, p. 181.
83. Peirce, *CP*, 4.372.
84. Peirce, *CP*, 4.530.

Chapter 3

1. Those who are familiar with Peirce's Alpha and Beta systems can skip §3.1.
2. Peirce *CP*, 4.378. According to this linear notation, cuts are represented by a pair of brackets (or parenthese), and juxtaposition is represented by concatenation. So '[*G*]' is Peirce's linear notation of the graph *G* enclosed in a single cut.
3. Two of these examples are borrowed from Roberts 1974, p. 51.
4. For now, let's assume that there is no function symbol in $\mathscr{L}_p$.
5. Zeman 1964, p. 72.
6. We do not allow the overlapping between any parts of these graphs.
7. We make a distinction between a cut and a cycle by the thickness of a line: An LI, and hence a cycle, is thicker than a cut.
8. The first graph means "It is not the case that something is *F*," the second graph "Something is not *F*," the third graph "Something is *F*, and there is at least one more thing," and the fourth graph "Some *F* is not identical with itself." We will look at the semantics of Beta graphs more carefully in the next chapter.
9. Many different orders are available, for example, (i) clauses 2, 2, 4, 4, 3, and 5, (ii) clauses 2, 2, 4, 3, 5, and 4, etc. The meaning of the graph is "Some *F* or some *G* is not identical with itself."
10. Peirce, *CP*, 4.378.
11. Peirce, *CP*, 4.385–390.
12. The reader may skip the rest of this subsection without loss of understanding.
13. Zeman 1964, pp. 72–73. I change some of his terminology and sentences (without changing the content of the definition) so that the comparison between Zeman's and my approaches to the Beta system may become clearer.
14. Peirce, Ms. 439, p. 16, quoted in Roberts, p. 115.
15. Peirce, *CP*, 4.391.
16. See Roberts 1973, p. 18.
17. Peirce 1897.
18. Roberts 1973, p. 19.
19. Roberts 1973, p. 20.
20. As I will discuss in the next section, the notational device of identity lines as the representation of something has a strong iconic element.
21. Roberts 1973, p. 20.

22. Zeman 1968, p. 147. (The italics are mine.)

23. Peirce 1897, p. 174.

24. Refer to Roberts 1973, pp. 25–26.

25. For more details, refer to Shin 1994, pp. 163–165 and pp. 173–176.

26. As said above, displaying each piece of information without any sign intervening is the closest iconic representation of conjunction we can get.

27. Recall that it is not possible to represent disjunction iconically, since disjunction itself is not displayed in reality.

28. Shin 1994.

29. Peirce, *CP*, 4.356. Throughout the discussion, Peirce mistakenly calls Venn diagrams "Euler diagrams." Peirce also named the section "Of Euler diagrams."

30. Interestingly enough, the title of the paper in which his first graphical system, Entitative Graphs, was presented is "The logic of relatives" (1897).

31. Peirce, *CP*, 4.442.

32. Peirce, *CP*, 4.442. (The italics are mine.)

33. Peirce, *CP*, 4.444.

34. Peirce, Ms., 462, p. 8.

35. Zeman 1968, pp. 145–146. (The italics are mine.)

36. Of course, we cannot say this convention is absolutely arbitrary, since it relies on tokens of the *same* type of a variable.

37. "No way of doing this can be more perfectly iconic than that" (Peirce, *CP*, 4.442).

38. Roberts 1973, p. 47.

39. Peirce, *CP*, 4.458. *X* is evenly (oddly) enclosed if and only *X* is enclosed by an even (odd) number of cuts.

40. These two graphs are borrowed from Roberts 1973, p. 51.

41. Zeman cites the above passage from Peirce, *CP*, 4.458, at Zeman 1964, p. 11. Roberts also says the following:

The 'principle secret' of interpretation (as Peirce phrased it in Ms 454, p. 18) is this: we are to consider a line of identity to be as much enclosed as its least enclosed (that is, its outermost) part (cf. 4.387); and if this part is on SA [the sheet of assertion] it denotes *something suitably chosen*, but if its outermost part is once enclosed it denotes *anything you please* (cf. 4.458). (Roberts 1973, p. 51)

[I]f this outermost part is evenly enclosed the line refers to "some" suitably chosen individual, while if the outermost part is oddly enclosed the line refers to "any" individual you please. (Roberts 1992, p. 645.)

42. Roberts 1973, p. 53.

43. Roberts 1973, p. 52.

Chapter 4

1. Zeman 1964, pp. 124–137, and Roberts 1973, pp. 139–151.

2. In chapter 6 relevant passages are cited from Peirce and discussed at great length.

3. Throughout chapters 4 and 5, I concentrate on particular criticisms against EG, rather than problems with graphical systems in general.

4. For a discussion of the visual power of Euler diagrams, refer to Hammer and Shin 1996.

5. As introduced in the previous chapter, the Alpha system, the EG equivalent to sentential languages, consists of only two kinds of syntactic devices, that is, cut and juxtaposition. Hence, intuitively, this system seems similar to a sentential language which has only two kinds of connectives, i.e., conjunction and negation.

6. Roberts 1973, p. 39.

7. Roberts 1973, p. 39.

8. Peirce, Ms. 650, pp. 18–19, quoted by Roberts 1973, p. 39, n. 13.

9. A sentence is in *negation normal form* iff every negation symbol is immediately followed by a sentence symbol only.

10. Throughout the chapter, 'subgraph' has the same meaning as Peirce's 'partial graph'. For Peirce's meaning of 'partial graph', see Roberts 1973, p. 33.

11. It can be easily shown that the set $\mathscr{G}'$ is equivalent to the set $\mathscr{G}_\alpha$ defined in §3.1.1.

12. Peirce, *CP*, 4.378. Peirce also introduces parentheses, brackets, and braces for his linear notation, which seem to have been intended for visual clarity. For example, according to Peirce, the graph in example 4.1 would be translated into $\{(P)[(Q)R]\}$. White also adopts this linear notation not only because it is easier to put these notations into print than graphical ones, but also because he wants to show that "the Alpha system is distinguished not by being graphical or iconic, but by its ingenious inference rules" (1984, p. 352). White also makes a distinction between even and odd areas, by replacing "the parentheses immediately enclosing an even area with square brackets" (1984, p. 353). According to this notation, the graph in example 4.1 is translated into $[(P)[(Q)R]]$. However, both Peirce's original notation and White's modified version easily lead to a confusion when a graph is complicated in having several nestings. Therefore, I do not intend to transform the entire Alpha system into this linear notation system. On the other hand, I agree with White that it is not always easy to draw graphs on paper. Hence, I adopt brackets, '[ ]', for convenience only in simple cases, that is, *only when* it is not visually confusing. I chose to use brackets, not parentheses, in order to avoid confusion with parentheses in first-order translation sentences.

13. Without this condition, we would obtain an infinite number of derivational histories for one and the same graph, which would prevent us from getting a

recursive definition of unique translations. Note that graph P could be a juxtaposition of P and an empty space, *or* a juxtaposition of P, an empty space, and an empty space, etc.

14. Recall the condition in the third clause of the set of Alpha graphs: None of $G_1, \ldots, G_n$ is an empty space. We do not want to obtain an infinite number of readings from one graph.

15. Roberts 1973, p. 46.

16. Set $\mathscr{G}'$ is defined in §4.2.1. After proving that $\mathscr{G}_\alpha = \mathscr{G}'$, one may prove lemma 4.1 using the definition of $\mathscr{G}_\alpha$. In that case, a cut of a sentence letter and an empty cut are not included in the basis cases. Also, there will be only two inductive cases: (i) $v \models_f [G]$ iff $v \not\models_f [[G]]$, and (ii) $v \models_f G_1 \ldots G_n$ iff $v \not\models_f [G_1 \ldots G_n]$.

17. We adopt the definition of Alpha graphs presented in §3.1.1, $\mathscr{G}_\alpha$, which is equivalent to the definition of $\mathscr{G}'$ presented in §4.2.1.

18. According to Roberts' rearrangement of Peirce's conventions, the convention for material implication became convention 4, while it was the third convention according to Peirce. For Peirce's numberings, see Peirce, *CP*, 4.436–437. (My note.)

19. Roberts 1973, pp. 33–35.

20. For a linear representation, Peirce adopts brackets and parentheses to express a scroll. That is, '$[X\,[Y]]$' is expressed as '$[X(Y)]$' by Peirce. See Peirce, *CP*, 4.378.

21. In this case, it happens to be the same as the formula obtained by the NNF reading method of the second section.

22. The following four cases correspond to the four cases listed in example 4.9 in the previous subsection.

23. This is the clause for a translation of graph $[[X][Y]]$ into the sentence $(\alpha \vee \beta)$, where α is a translation of X and β is a translation of Y.

24. "For any graph P, let '{P}' denote the place of P" (Roberts 1973, p. 38). (My note.)

25. Roberts 1973, pp. 41–45.

26. These four do not exhaust the possible construction sequences.

27. $\mathscr{G}_\alpha$ in §3.1.1 and $\mathscr{G}'$ in §4.2.1.

28. Peirce, *CP*, 4.380.

29. For more discussion of this issue, refer to Shin 1994, chapter 6.

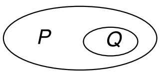
30. Richard White also rewrote Peirce's permissions (Peirce's rules) to make the symmetry more specific, by using 'even introduction', 'even elimination', 'odd introduction', and 'odd elimination' (see White 1984, p. 353). However, his project starts with the conviction that "'iconic' notation is logically inessential" (White 1984, p. 352). Hence, he transforms Alpha graphs into linear

expressions, which does not make the meanings of inference rules understood easily. White seems to believe that graphical notations are confusing, but he does not realize that linear expressions with many parentheses, not having visual clarity, are even more confusing. For instance, in one of his examples, ([A([B(C)])][([A(B)][(A[C])])]), it is not easy to see how parentheses and brackets are matched (White 1984, p. 353).

31. I adopt Roberts' notation here. Roberts says, "It is convenient to inject some order into the areas of graphs. For any graph P, let $\{P\}$ denote the place of P. And let the relation symbol $\supseteq$ be defined as follows: $\{B\}$ *is enclosed by every cut that encloses* $\{A\}$ *if and only if* $\{A\} \supseteq \{B\}$. The sign $\supseteq$ may be read 'contains'" (Roberts 1973, p. 38). With this notation, Roberts defines the identity of the areas: "If $\{P\} \supseteq \{Q\}$ and $\{Q\} \supseteq \{P\}$, then $\{P\} = \{Q\}$, i.e., P and Q are scribed on the same area" (Roberts 1973, n. 11, p. 38). Then we give the following definition:

Definition Area b is the *next-outer area* from area a if and only if (i) $a \subseteq b$, (ii) $a \neq b$, and (iii) there is no other area between a and b.

For example, in the following, P is in the next-outer area from $\{Q\}$:



32. A double cut consists of two cuts where one is enclosed by the other and nothing is written between these two cuts.

33. We write $a \subseteq b$ if and only if area a is contained by area b. (See note 31.)

34. Dots express other possible subparts of a graph. The letters 'E' and 'O' are not part of the graphs but marks for E-area and O-area, respectively. For example, Y is in an area enclosed by an even number of cuts.

35. The erasure rule allows us to erase anything in an E-area. So we do not need the deiteration rule in the case of E-areas.

36. There seems to be no rule in EG which corresponds to the $\wedge$-introduction rule. Hammer notices that "Any needed information must be carried along as one proceeds" (Hammer 1995 p. 105).

37. Dots express other possible subparts of a graph.

38. This redundancy is explained in §4.4.1 and §4.4.2 in detail.

39. Dots express other possible subparts of a graph.

40. This reading is obtained by the NNF reading method.

41. This example was used by Peirce to explain "logically necessary reasoning" and was also cited in Roberts 1973, p. 111.

42. Refer to examples 4.11, 4.12, and 4.13.

43. A way to get around this problem is by introducing substitution among sentence letters and subgraphs.

Chapter 5

1. Zeman 1964, chapter 2.

2. Roberts 1973, chapter 4.

3. Refer to §3.1.2.

4. In §5.4, I will discuss recent work on Peirce's Beta graphs by Eric Hammer.

5. Zeman 1964, chapter 2.

6. Roberts 1973, chapter 4.

7. Zeman 1964, chapter 1. By "the available algorithm for Alpha graphs," I mean the endoporeutic reading method discussed in §4.1.

8. He also correctly realizes that LIs of the Beta system (roughly) correspond to variables of a first-order language.

9. Zeman 1964, p. 72.

10. Zeman 1964, p. 74.

11. Zeman himself uses L, but for a clearer comparison, I adopt a letter with a superscript, 2, for a temporary binary predicate.

12. Zeman himself uses B, but for the same reason as above, I adopt a different symbol from Zeman's.

13. Zeman 1964, p. 79. Again, according to Zeman, these are L and B.

14. Zeman uses the symbol Q instead of L^1. Again, I adopt a letter with a superscript 1 to make it clear that it is a unary temporary predicate.

15. Note that the order of the variables in the formula 'L^2xy' (of the third graph) is important. So the stipulation of how to preserve the order of variables is needed.

16. Zeman 1964, p. 81.

17. Note that $\neg\exists x P(x) \not\equiv \exists x \neg P(x)$.

18. These clauses are a total reconstruction of the content in Zeman 1964, pp. 78–88.

19. Temporary predicates are also included.

20. This is the traditional reading method (i.e., the endoporeutic method) of the Alpha graphs discussed in the previous chapter, §4.1.

21. Refer to the third clause of Zeman's algorithm above.

22. Refer to note 41 in chapter 3.

23. Zeman 1964, p. 12.

24. Refer to §4.1.

25. When a wff is transformed into a sentence, "for each variable x in the wff, write $\exists x$ in front of *the shortest subformula* which contains all the occurrences of x." (See Zeman's reading algorithm clause (3).)

26. Roberts 1992, p. 643.

27. Roberts 1973, p. 137. These conventions are rewritten by Roberts, and at the end of each convention Roberts indicates its source from Peirce, *CP*.

28. The closest translation of a heavy dot is '$\exists x(x = x)$'.

29. Roberts 1973, p. 49.

30. Roberts 1973, p. 53.

31. Roberts 1973, p. 53.

32. I suspect that Roberts analyzes the line to be one network of a line with three branches, like the following:

33. Roberts 1992, p. 645.

34. Roberts 1992, p. 645.

35. Peirce, *CP*, 4.468.

36. Refer to the previous subsection, §5.1.1.

37. Peirce, *CP*, 4.469, and Roberts 1973, p. 53.

38. Roberts 1992, p. 645.

39. Roberts 1973, p. 60.

40. The sentence '$\forall x \exists y(\text{MotherOf}(x, y) \rightarrow \text{Loves}(x, y))$' is equivalent to '$\forall x \exists y(\neg\text{MotherOf}(x, y) \vee \text{Loves}(x, y))$.' Hence, this sentence is always true, since no one is a mother of himself or herself. On the other hand, a correct translation for the English sentence 'Every mother loves some child of hers' is "$\forall x(\exists y\text{MotherOf}(x, y) \rightarrow \exists z[\text{MotherOf}(x, z) \wedge \text{Loves}(x, z)])$."

41. Roberts 1973, p. 60.

42. Robert 1973, p. 51.

43. The quotation of note 41 above is an example of this, even though it yields an incorrect result for a different reason.

44. These two graphs are borrowed from Peirce, Ms., p. 3. Also refer to Peirce, *CP*, 4.452.

45. "*C7* A heavy line, called a line of identity, shall be a graph asserting the numerical identity of the individuals denoted by its two extremities" (Roberts 1973, p. 137; my note).

46. Roberts 1973, p. 52.

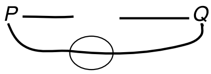
47. Roberts 1973, p. 52.

48. Roberts 1992, pp. 645–646.

49. Roberts 1992, p. 644.

50. Roberts 1992, p. 644.

51. Notice that the following graph would not be a correct one, since both predicates *P* and *Q* become binary in this graph, while *P* and *Q* are unary in '$\exists x \exists y(Px \wedge Qy \wedge x \neq y)$'.



52. Since the line is not disconnected or divided visually, as seen in the examples above, we do not want to say that the line is disconnected or divided.

53. For graph (3), recall that a cycle is a special case of an LI. See §3.1.2.

54. In a clockwise direction.

55. If a cycle is clipped, then by clause 2(b)(ii) an identity wff is written inside the cut.

56. Refer to §4.2.1.

57. However, we do not erase them, since we need them later.

58. A main difference between the graphs we have now and Zeman's quasi-Alpha graphs is that ours do have LIs but Zeman's do not. Since we will not read off LIs in this process, we may call them quasi-Alpha graphs.

59. A quasi-Alpha graph is the same as an Alpha graph except that it has an atomic wff or $\top$ instead of sentence symbols. Very similarly to what I did for Alpha graphs in §4.2.1, I define a simple quasi-Alpha graph as follows:

Definition Graph *G* is a simple quasi-Alpha graph iff *G* is an atomic wff, a single cut of an atomic wff, $\top$, a single cut of $\top$, an empty space, or a cut of an empty cut.

60. In this linear notation, cuts are represented by brackets.

61. Recall that in step 2 variables are written at the outermost part of the line.

62. If there is no subformula containing v_i, this step is skipped. When we illustrate each clause of the algorithm, the condition 'the smallest negation-normal-form subformula' will be clarified.

63. If there is no subformula containing v_i, this step is skipped.

64. Let's say $\mathcal{Q}_i$ is either $\exists$ or $\forall$ and is decided depending on where variable v_i is written.

65. The condition in clause 2(b) that an LI crosses an *odd* number of cuts is important.

66. Notice that the second cut does not clip the line.

67. Note that clause 2(b)(ii) is not applied yet.

68. We sometimes would like to write down variable $[v_i]$ in a non-outermost part of the line so that we may keep track of it. But in this case, without brackets, there would be confusion as to how to apply clause 5 correctly.

69. Zeman defines a non-geodesic line to be a line which crosses one and the same cut more than once, and his reading requires that a non-geodesic line be broken into two parts to receive two different types of variables. For more details, refer to §5.1.1.

70. When we need to write down a token of the same type of variable to keep track of an unclipped part of an LI, brackets are used. For step 5, since we are interested in the outermost part of an LI, we ignore the positions of the variables in brackets.

71. Note that Zeman's reading requires us to add a quantifier in front of the smallest subformula. Let's recall that Zeman's algorithm never allows us to add a universal quantifier. So, by adding an existential quantifier in front of the smallest subformula, an existential quantifier is always placed next to a negation symbol, if any, which is the way a universal quantifier is expressed.

72. The reader can easily figure out '$\top$' and '$\neg\top$' can be ignored in '$\top \wedge (\exists x \forall y \neg Fxy \vee \neg\top)$'.

73. Refer to §5.1.1.

74. We have seen how Zeman's reading results in $\neg\exists x\exists y[x = y \wedge \neg\exists u\exists v \cdot (u = v \wedge \neg(\text{MotherOf}(x, u) \wedge \neg\text{Loves}(y, v)))]$, which is logically equivalent to $\forall x\exists y(\text{MotherOf}(x, y) \rightarrow \text{Loves}(x, y))$ but uses four types of variables. I pointed out that Robert mistakenly thought that this graph represents the English sentence 'Every mother loves some child of hers'.

75. Refer to §5.1.2 and Roberts 1992, pp. 645–646.

76. Roberts 1992, p. 645.

77. '$P(x)$' means x occurs in a formula as one of the arguments.

78. Roberts 1973, pp. 56–60. For the original version of Peirce's rules, see Peirce, *CP*, 4.505–509.

79. "For any graph P, let '$\{P\}$' denote the place of P" (Roberts 1973, p. 38; my note).

80. Peirce, *CP*, 4.506.

81. Roberts 1973, p. 57.

82. Peirce, *CP*, 4.506.

83. Roberts 1973, p. 57.

84. Roberts 1973, p. 59.

85. This is the double cut rule for the Alpha system. (My note.)

86. Zeman, p. 16. The italics are mine.

87. Peirce, *CP*, 4.508.

88. This is what Peirce, Zeman, and Roberts have done. Refer to the parts which address the manipulations of LIs in R1, R2, R3, and R4 above.

89. As Peirce originally stated his rules, the validity of several subclauses of both the iteration and deiteration rules is far from being clear.

90. The only addition in the Beta system is that we are allowed to draw an LI in an E-area. However, since an LI in an E-area without any predicate attached means $\top$, this addition is obviously legitimate.

91. The parts "..." are the same in both translations.

92. This is why NR3(b) and NR4(a) have more subclauses than NR3(a) and NR4(b).

93. I will go through the three clauses of NR3(b), and since NR4(a) is a mirror image of NR3(b), I will leave it to the reader.

94. Suppose that there is no other LI which this extension overrides.

95. Suppose that the extension overrides another LI whose outermost part ends in an O-area. Assume that the variable *x* is assigned to the LI which we are interested in extending.

96. Suppose that the extension overrides another LI whose outermost part ends in an E-area. Assume that the variable *x* is assigned to the LI which we are interested in extending.

97. I explained above why I decided to add these two new rules, instead of adding them to the two existing rules NR1 and NR2.

98. Notice that Roberts did not state the counterpart of his R3(c) in his deiteration rule R4.

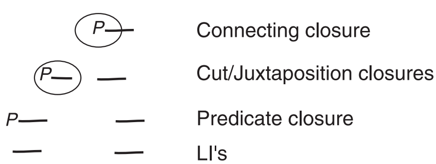
99. Note that R3(c) allows the extension of an LI, but only inwards.

100. Roberts 1992, p. 652. Graph (4) corresponds to Roberts' figure 65, and graph (5) to figure 66.

101. Hammer 1998.

102. Hammer 1998, p. 492.

103. Some might notice that Zeman's definition of Beta graphs give us two different derivational histories for graph (3) and graph (3′), since Zeman's cut closure does not allow us to draw a single cut in any subpart of a given Beta graph, but only a full cut. So for Zeman, the history of graph (3′) is the following:



However, Hammer does not consider this syntactic definition at all. As we will see shortly, Hammer's semantics does not treat the identity line in graph (3′) as two different identity lines to be joined by Zeman's connecting closure. Instead both identity lines in (3) and (3′) are treated as one single line.

104. Later we will examine Hammer's algorithm for this process.

105. Hammer 1998, pp. 498–499.

106. Hammer 1998, pp. 499–500.

107. Note that these are not the same as the graphs in the basic set of well-formed graphs.

108. Hammer 1998, pp. 500–501.

109. Hammer 1998, p. 499.

110. According to his presentation, clause 7.

111. The set of Hammer's well-formed graphs is a subset of the set of v-Beta graphs. All we have to prove is that the graphs constructed by Hammer's S1(b), S2(a), and S2(e) may be constructed by our new clauses. I leave the details of the proof to the reader.

112. Hammer 1998, p. 499.

113. A slight modification is that Hammer uses the notion of retracting lines with variables, while Zeman introduces temporary predicates.

Chapter 6

1. Recall that adopting the endoporeutic reading does not allow us to read off a visual distinction for universal and existential quantifiers. Another clear visual feature Peirce emphasized—the treatment of scope among quantifiers—is not useful, either. Moreover, perceiving a scroll is not reflected in the traditional method.

2. Peirce, *CP*, 4.373.

3. Peirce, *CP*, 4.373.

4. Peirce, *CP*, 4.134, quoted by Haack 1993, p. 48.

5. Peirce, *CP*, 4.134.

6. Peirce, *CP*, 4.373.

7. One exception is the use of double cuts: We could draw double cuts to make a different diagram but represent the same fact.

8. Peirce, *CP*, 4.375.

9. Refer to §3.3.2.

10. Refer to the quotation of note 6, from Peirce, *CP*, 4.373.

Chapter 7

1. As identified in the Introduction, this is a classic example to illustrate a vicious circle between a prejudice for symbolic systems and an unclear distinction between symbolic and graphical systems.

2. Quine 1934, p. 552, quoted in Roberts 1973, p. 13.

3. Traditionally, disjunction is believed to be indirectly expressed in EG in terms of negation and conjunction.

4. Even though there is no separate graphical sign for disjunction, we are able to read off disjunctive information from Peirce's graphs. See §4.2.1. As argued in the previous chapter, this is a strong point of this graphical system, since it does not generate more metatheorems but increases its expressive power.

5. It would be an interesting and quite feasible project to apply this idea to Peirce's Gamma system.

Bibliography

Barwise, Jon (1993). "Heterogeneous reasoning." In *Conceptual Graphs for Knowledge Representation*, G. Mineau, B. Moulin and J. Sowa, eds. Berlin: Springer-Verlag. 64–74.

Barwise, Jon, and Gerard Allwein, eds. (1996). *Logical Reasoning with Diagrams*. New York: Oxford University Press.

Barwise, Jon, and John Etchemendy (1989). "Information, infons and inference." In *Situation Theory and Its Applications I*, vol. 1, Cooper, Mukai and Perry, eds. Stanford, CA: Center for the Study of Language and Information. 33–78.

Barwise, Jon, and John Etchemendy (1991). "Visual information and valid reasoning." In *Visualization in Teaching and Learning Mathematics*. W. Zimmerman and S. Cunningham, eds. Washington, DC: Mathematical Association of America. 9–24.

Barwise, Jon, and John Etchemendy (1993). *The Language of First-Order Logic*. 3rd. ed. Stanford, CA: Center for the Study of Language and Information.

Barwise, Jon, and John Etchemendy (1994). *Hyperproof*. Stanford, CA: Center for the Study of Language and Information.

Barwise, Jon, and John Etchemendy (1995). "Heterogeneous reasoning." In Chandrasekaran, Glasgow and Narayanan (1995). 211–234.

Barwise, Jon, and Eric Hammer (1994). "Diagrams and the concept of logical system." In *What Is a Logical System?* D. M. Gabbay, ed. Oxford: Clarendon Press; New York: Oxford University Press.

Block, Ned, ed. (1981). *Imagery*. Cambridge: MIT Press (Bradford).

Block, Ned (1983). "Mental pictures and cognitive science." In *Philosophical Review* 92, 499–541.

Bower, Gordon (1970a). "Imagery as a relational organizer in associative learning." In *Journal of Verbal Learning and Verbal Behavior* 8, 529–533.

Bower, Gordon (1970b). "Analysis of the mnemonic device." In *American Scientist* 58, 496–501.

Burch, Robert (1991). *Peircean Reduction Thesis: The Foundation of Topological Logic*. Lubbock, Tex.: Texas Tech University Press.

Chandrasekaran, B., J. Glasgow and N. H. Narayanan, eds. (1995). *Diagrammatic Reasoning: Cognitive and Computational Perspective*. Menlo Park, Calif.: AAAI Press; Cambridge, Mass.: MIT Press.

Cummins, Robert (1996). *Representation, Targets, and Attitudes*. Cambridge, Mass.: MIT Press. A Bradford Book.

Dennett, Daniel (1981). "The nature of images and introspective trap." In Block (1981), 87–107.

DeMorgan (1847). *Formal Logic*. London: Taylor and Walton.

Dipert, Randall (1995). "Peirce's underestimated place in the history of logic: a response to Quine." In *Pierce and Contemporary Thought*. New York: Fordham University Press. 32–58.

Dipert, Randall (1996). "Reflections on iconicity, representation, and resemblance: Peirce's theory of signs, Goodman on resemblance, and modern philosophies of language and mind." In *Synthese* 106: 373–397.

Dipert, Randall (1997). "The mathematical structure of the world: the world as graph." In *Journal of Philosophy* 94(7), 329–358.

Euler, Leonhard (1986). *Briefe an eine deutsche Prinzessin*. Braunschweig: Vieweg.

Finke, R., and S. Pinker (1982). "Spontaneous imagery scanning in mental extrapolation." In *Journal of Experimental Psychology: Learning, Memory and Cognition* 2, 142–147.

Fodor, Jerry (1981). "Imagistic representation." In Block (1981), 63–86.

Freeman, Eugene, ed. (1983). *The Relevance of Charles Peirce*. La Salle, IL: Hegler Institute.

Frege, G. (1967 [1879]). *Begriffsschrift*. In *From Frege to Gödel*, van Heijenoort, ed. Cambridge, Mass.: Harvard University Press.

Gardner, Martin (1982 [1958]). *Logic Machines and Diagrams*. Chicago: University of Chicago Press. 2nd ed.

Glasgow, Janice, and Dimitri Papadias (1992). "Computational imagery." In *Cognitive Science* 16, 355–394.

Goel, Vinod (1995). *Sketches of Thought*. Cambridge, Mass.: MIT Press.

Goldfarb, Warren (1979). "Logic in the twenties: the nature of the quantifier." In *Journal of Symbolic Logic* 44(3), 351–368.

Goodman, Nelson (1976). *Languages of Art*. Indianapolis: Hackett. 2nd ed.

Goudge, Thomas (1965). "Peirce's index." In *Transactions of the Charles S. Peirce Society* 1, 52–70.

Gurr, Corin, John Lee and Keith Stenning (1998). "Theories of diagrammatic reasoning: distinguishing component problems." In *Minds and Machines* 8, 533–557.

Haack, Susan (1993). "Peirce and logicism." In *Transactions of the Charles S. Peirce Society* 29(1), 33–56.

Haber, Ralph N. (1970). "How we remember what we see." In *Scientific American* 222, 104–112.

Hacking, Ian (1975). *Why Does Language Matter to Philosophy?* New York: Cambridge University Press.

Hammer, Eric (1995). *Logic and Visual Information*. Stanford, CA: Center for the Study of Language and Information.

Hammer, Eric (1998). "Semantics for Existential Graphs." In *Journal of Philosophical Logic* 27, 489–503.

Hammer, Eric, and Sun-Joo Shin (1996). "Euler and the role of visualization in logic." In *Logic, Language and Computation*, CSLI, Lecture Notes 58, 271–286. Jerry Seligman and Dag Westerstaåhl, eds. Stanford, CA: Center for the Study of Language and Information.

Hammer, Eric, and Sun-Joo Shin (1998). "Euler's visual logic." In *History and Philosophy of Logic* 19(1), 1–29.

Harel, David (1988). "On visual formalisms." In *Communications of the ACM* 13(5), 514–530.

Hauser, Nathan, Don Roberts and James van Evra, eds. *Studies in the Logic of Charles S. Peirce*. Bloomington: Indiana University Press.

Hintikka, Jaakko (1988). "On the development of the model-theoretic viewpoint in logical theory." In *Synthese* 77, 1–36.

Hintikka, Jaakko (1990). "Quine as a member of the tradition of the universality of language." In *Perspective on Quine*, Robert Barret and Roger Gibson, eds. 159–175.

Hintikka, Jaakko (1997). "The Place of C. S. Peirce in the history of logical theory." In *The Rule of Reason*, Brunning and Forster, eds. Toronto: University of Toronto Press.

Holt, Robert (1964). "Imagery: the return of the ostracized." In *American Psychologist* 19, 254–264.

Howse, John, Fernando Molina, and John Taylor (2000). "SD2: a sound and complete diagrammatic reasoning system." In *Proc. Artificial Intelligence and Soft Computing* (ASC 2000), Banff, July 2000, 402–408.

Huttenlocher, J. (1968). "Constructing spatial images: a strategy in reasoning." In *Psychological Review* 75: 550–560.

Johnson-Laird, Philip (1983). *Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness*. Cambridge, Mass.: Harvard University Press.

Kent, Beverley (1987). *Charles S. Peirce: Logic and the Classification of the Sciences*. Kingston: McGill-Queen's University Press.

Kent, Beverley (1997). "The interconnectedness of Peirce's diagrammatic thought." In Hintikka (1988), 445–459.

Ketner, Kenneth L. (1982). "Carolyn Eisele's place in Peirce's studies." In *Historia Mathematica* 9, 326–332.

Ketner, Kenneth L. (1985). "How Hintikka misunderstood Peirce's account of theorematic reasoning." In *Transactions of the Charles S. Peirce Society* 21, 407–418.

Ketner, Kenneth L. (1987). "Identifying Peirce's 'most lucid and interesting paper'." In *Transactions of the Charles S. Peirce Society* 23(4), 539–555.

Kneale, William, and Martha Kneale (1962). *The Development of Logic*. Oxford: Clarendon Press.

Kosslyn, Stephen (1980a). *Image and Mind*. Cambridge: Harvard University Press.

Kosslyn, Stephen (1980b). "The medium and the message in mental imagery: a theory." In Block (1981), 207–244.

Kosslyn, Pinker, Smith and Schwartz (1981). "On the demystification of mental imagery." In Block (1981), 131–150.

Larkin, Jill, and Herbert Simon (1987). "Why a diagram is (sometimes) worth ten thousand words." In *Cognitive Science* 11, 65–99.

Levy, Stephen (1997). "Peirce's theorematic/corollarial distinction and the interconnection between mathematics and logic." In Hintikka (1988), 85–110.

Lindsay, Robert (1988). "Images and inferences." In *Cognition* 29, 229–250.

Müller, Ralf (forthcoming). "Interpretation of modality: epistemic logic and Peirce's logic of ignorance." In *Festschrift for Thomas M. Seebohm.*

Narayanan, N. H., ed. (1993). "Taking issue/forum: the imagery debate revisited." In *Computational Intelligence* 9(4), 303–435.

Oatley, Keith (1977). "Inference, navigation and cognitive mans." In *Thinking: Readings in Cognitive Science*, Johnson-Laird and Wason, eds. Cambridge: Cambridge University Press. 537–547.

Olsen, Leonard (forthcoming). "On Peirce's systematic division of signs." In *The Transactions of the Charles S. Peirce Society*.

Peirce, Charles (1884). "On the algebra of logic." In *American Journal of Mathematics* 7, 180–202.

Peirce, Charles (1897). "The logic of relatives." In *The Monist* 7(2), 161–217.

Peirce, Charles S. (1931–1958). *Collected Papers of Charles Sanders Peirce*, Charles Hartshorne and Paul Weiss, eds. Cambridge: Harvard University Press. [In the text, this is abbreviated as *CP*.]

Peirce, Charles S. Manuscripts in Houghton Library, Harvard University. [In the text, this is abbreviated as *Ms*.]

Peirce, Charles S. (1976). *The New Elements of Mathematics*. C. Eisele, ed. Hague: Mouton Publishers; Atlantic Highlands, N.J.: Humanities Press. [In the text, this is abbreviated as *NEM*.]

Peirce, Charles S. (1982–). *Writings of Charles S. Peirce: A Chronological Edition*. M. Fisch, general ed. Bloomington: Indiana University Press. [In the text, this is abbreviated as *CE*.]

Peirce, Charles S. (1992). *Reasoning and the Logic of Things: The Cambridge Conferences Lectures of 1898*. K. L. Ketner, ed. Cambridge, Mass.: Harvard University Press.

Putnam, Hilary (1982). "Peirce as logician." In *Historia Mathematica* 9, 290–301.

Pylyshyn, Zenon (1973). "What the mind's eye tells the mind's brain: a critique of mental imagery." In *Psychological Bulletin* 80, 1–24.

Pylyshyn, Zenon (1979). "The rate of 'mental rotation' of images: a test of a holistic analogue hypothesis." In *Memory and Cognition* 7(1), 19–28.

Pylyshyn, Zenon (1981a). "Imagery and artificial intelligence." In *Readings in Philosophy of Psychology*, vol. 2, Ned Block, ed., 170–194.

Pylyshyn, Zenon (1981b). "Image debate: analog media versus tacit knowledge." In *Psychological Review* 88, 16–45.

Quine, Williard Van Orman (1934). "Review of the Collected Papers of Charles Sanders Peirce, Volume 4: *The Simplest Mathematics*." In *Isis* 22, 551–553.

Roberts, Don (1963). "The Existential Graphs of Charles S. Peirce" Ph.D. Diss. University of Illinois.

Roberts, Don (1973). *The Existential Graphs of Charles S. Peirce*. The Hague: Mouton.

Roberts, Don (1992). "The Existential Graphs." In *Computer and Math. Applic.* 23, 639–663.

Sanders, Gary (1970). "Peirce's sixty-six signs?" In *Transactions of the Charles S. Peirce Society* 6(1), 3–16.

Shepard, Roger, and Jacqueline Metzler (1971). "Mental rotation of three-dimensional objects." In *Science* 171, 701–703.

Shimojima, Atsushi (1996). "Operational constraints in diagrammatic reasoning." In Barwise and Allwein (1996), 27–48.

Shin, Sun-Joo (1994). *The Logical Status of Diagrams*. New York: Cambridge University Press.

Shin, Sun-Joo (1997). "Kant's syntheticity revisited by Peirce." In *Synthese* 113(1), 1–41.

Shin, Sun-Joo (1999). "Reconstituting Beta Graphs into an efficacious system." In *Journal of Logic, Language and Information* 8, 273–295.

Shin, Sun-Joo (2000). "Reviving the iconicity of Beta Graphs." In *Theory and Application of Diagrams*, Anderson, Cheng and Haarslev, eds. Berlin: Springer. 58–73.

Shin, Sun-Joo (forthcoming). "Multiple readings of Peirce's Alpha Graphs." In *Thinking with Diagrams 98*. Springer.

Shorter, J. M. (1952). "Imagination." In *Mind* 61, 528–542.

Simon, Herbert (1978). "On the forms of mental representation." In *Perception and Cognitive Issues in the Foundation of Psychology*, Minnesota Studies in the Philosophy of Science, vol. 9, C. Wade Savage, ed.

Sloman, Aaron (1971). "Interaction between philosophy and AI: the role of intuition and non-logical reasoning in intelligence." In *Proceedings Second International Joint Conference on Artificial Intelligence*. London, San Francisco: Morgan Kaufmann.

Sloman, Aaron (1985). "Why we need many knowledge representation formalisms." In *Research and Development in Expert Systems*, M. Bramer, ed. 163–183.

Sloman, Aaron (1995). "Musings on the roles of logical and nonlogical representations in intelligence." In Chandrasekaran, Glasgow and Narayanan (1995), 7–32.

Sluga, Hans (1987). "Frege against the Booleans." In *Notre Dame Journal of Formal Logic* 28(1), 80–98.

Smart, J. J. C. (1959). "Sensations and brain processes." In *Philosophical Review* 68, 141–156.

Sowa, John (1984). *Conceptual Structure: Information Processing in Mind and Machine*. Reading, Mass.: Addison-Wesley.

Sowa, John (2000). *Knowledge Representation: Logical, Philosophical, Computational Foundations*. Belmont, Calif.: Brooks/Cole.

Stenning, Keith (1996). "The cognitive impact of diagrams." In *Philosophy and Cognitive Science*, A. Clark et al., eds. Kluwer Academic Publishers. 181–196.

Stenning, Kith, and Peter Yule (1997). "Image and language in human reasoning: a syllogistic illustration." In *Cognitive Psychology* 34, 109–159.

Sterelny, Kim (1986). "The imagery debate." In *Philosophy of Science* 53, 560–583.

Tappenden, Jamie (1997). "Metatheory and mathematical practice in Frege." In *Philosophical Topics* 25(2), 213–264.

Tye, Michael (1991). *The Imagery Debate*. Cambridge: MIT Press (Bradford).

Van Heijenoort, Jean (1967). "Logic as calculus and logic as language." In *Synthese* 17, 324–330.

Venn, John (1971 [1881]). *Symbolic Logic*. New York: Burt Franklin.

Wang, Dejuan, and John Lee (1993). "Visual reasoning: its formal semantics and applications." In *Journal of Visual Languages and Computing* 4, 327–356.

Weiss, Paul, and A. W. Burks (1945). "Peirce's sixty-six signs." In *Journal of Philosophy* 42, 383–388.

White, Richard (1984). "Peirce's Alpha Graphs: the completeness of propositional logic and the fast simplification of truth-function." In *Transactions of the Charles S. Peirce Society* 20, 351–361.

Zeman, Jay (1964). "The Graphical Logic of C. S. Peirce." Ph.D. Diss. University of Chicago.

Zeman, Jay (1968). "Peirce's Graphs—the continuity interpretation." In *Transactions of the Charles S. Peirce Society* 4(3), 144–154.

Index

Abstraction, 6, 24, 26–27, 29
Accidental properties, 20, 29, 35
Algebra, 15, 21–22, 23, 25, 34, 175 (n. 2), 178 (n. 5)
Alpha system
  definitions of, 38, 65, 74
  endoporeutic readings of, 61–63
  introduced, 37–38
  linear notation for, 43, 184 (n. 2), 186 (n. 12)
  multiple readings in, 75
  negation-normal-form readings of, 65–66
  semantics of, 68–71
  transformation rules of, 81, 84–85
Anderson, Michael, 175 (n. 1)
Aristotle, 17, 176 (n. 10)
Artificial intelligence (AI), 5, 7, 9–10, 173, 177 (n. 27)
Axiomatic systems, 168–169
Axiomatization, 17, 20

Barwise, Jon, 10, 177 (nn. 31, 32)
Beta system
  definitions of, 41, 43
  introduced, 39–40
  Roberts' readings of, 111–118
  Shin's readings of, 122–123
  transformation rules for, 134–135, 139–141
  Zeman's readings of, 104
Block, Ned, 175 (n. 6), 176 (n. 13)
Boole, George, 14–15
Bower, Gordon, 176 (n. 15)
Burks, A. W., 181 (n. 40)

Calculus, 14–17, 56, 59, 179 (n. 18)
  vs. logical system, 11, 163–172
Calculus ratiocinator, 14–16
Carnap, Rudolf, 15
Chandrasekaran, B., 5
Chef d'oeuvre, 49, 175 (n. 2)
Clippings, introduced, 121–122
Cognition, 5–10
Cognitive science, 1, 5–10, 173
Conditional information represented in EG, 72
Conjunctive information represented in EG, 38, 61–63
Conjunctive vs. disjunctive information represented in EG, 50–53, 64–65
Convention, 23–27, 30, 32–34, 53–54
Cummins, Robert, 176 (n. 13)
Cut, introduced, 3–4
Cycle, introduced, 41

Deiteration rule, 81, 134
De Morgan, William, 17, 179 (n. 20)
De Morgan's laws, 62, 97, 110
Dennet, Daniel, 176 (nn. 8, 13)
Depictionalists, 6
Derivational/syntactic history, 4, 81–82, 92, 151–152, 155
Descartes, René, 6
Descriptionalists, 6
Design theory, 5, 9–10, 173

Diagram (Peirce), 19–22
Diagrammatic reasoning (Peirce), 13, 19–22
Diagrammatic Reasoning Site (website), 175 (n. 1)
Dipert, Randall, 14, 17–18, 25, 34, 179 (nn. 18, 25), 181 (n. 50), 182 (nn. 55, 56, 61, 65)
Disjunctive information represented in EG, 50–53, 64–65
Double cut
 in the Alpha system, 65, 67–68, 81, 84, 90, 95–96
 in the Beta system, 119, 122, 124–125, 131, 135, 137–138, 140, 147, 149–150

E-area, defined, 64
Efficiency, 3–4, 8, 30, 32, 60–61, 80, 83, 90–93, 97, 110, 150, 161, 165, 167, 173
E-jux, defined, 64
Empty cut, defined, 65
Empty space, defined, 65
Endoporeutic reading, 61–63, 103, 109–110, 117, 127, 133, 151–152, 160, 163, 167
Entitative Graphs, 37, 48–53, 185 (n. 30)
Equivalence
 computational, 8, 175 (n. 35)
 informational, 8, 175 (n. 35)
 logical, 4, 10, 13, 51, 59–60, 70, 74–75, 85, 95–96, 99, 105–106, 113, 115–117, 131, 134, 145, 147, 166, 177 (n. 36)
Etchemendy, John, 10, 177 (nn. 31, 32)
Euclidean geometry, 19, 21, 27
Euler diagrams, 10, 26–27, 32–33, 48, 60, 180 (n. 31)
Evenly enclosed, 56, 64
External representation, 5–10

Fodor, Jerry, 175 (n. 7), 176 (n. 13)
Formalism/formalization, 17–18, 30–35
Free ride, 8, 183 (n. 77)
Frege, G., 13–18, 178 (n. 1)

Gamma system, 37, 180 (n. 31), 195 (n. 5)
Geodesic vs. non-geodesic, 101, 108–109
Glasgow, Janice, 5, 176 (n. 19)
Gödel, Kurt, 15
Goldfarb, Warren, 14–15
Goodman, Nelson, 25, 182 (n. 55), 183 (n. 67)
Goudge, Thomas, 181 (n. 48)
Gurr, Corin, 8, 182 (n. 64), 183 (nn. 73, 77)

Haber, Ralph N., 176 (n. 5)
Hacking, Ian, 176 (n. 11)
Hammer, Eric, 10, 150–161, 188 (n. 36)
Harel, David, 9
Heterogeneous reasoning/representation/system, 1, 3, 7–12, 13, 23, 27, 31–35, 37, 49, 56, 163, 171–174
Heuristic, 3, 9
Higraphs, 9
Hilbert, David, 20
Hintikka, Jaakko, 14–17, 34, 180 (n. 38)
Hobbes, Thomas, 6
Holt, Robert, 176 (n. 14)
Homomorphic relation, 26–28, 31, 35, 58, 183 (n. 66)
Hume, David, 6
Hyperproof, 10

Icon/iconic features/iconicity, 3, 11, 13, 16, 23–35, 37, 53–58, 111, 130, 163, 166, 168, 171–173. *See also* Visual aspects/distinction/features
Iconic formalization. *See* Heterogeneous reasoning/representation/system
Image, visual, 6

Imagery debate, 6
Index, 23–25
Information represented in EG
conditional, 72
conjunctive, 38, 61–63
conjunctive vs. disjunctive, 50–53, 64–65
negative, 38, 51–53, 61–63
quantified, 49, 50, 56–58
Internal representation, 5–10
Irreducibility of triads, 48
Iteration rule, 81, 134

James, William, 175 (n. 2)
Johnson-Laird, Philip, 175 (n. 5)
Juxtaposition(s)
conjunctive vs. disjunctive, 64–65
introduced, 38

Ketner, Kenneth L., 180 (n. 27)
Kosslyn, Stephen, 6, 175 (n. 7)

Larkin, Jill, 7–8, 177 (nn. 20, 21), 183 (n. 73)
Lee, John, 8, 9, 182 (n. 64), 183 (nn. 73, 77)
Likeness, 23–31, 33, 35, 51–52
Lindsay, Robert, 8, 176 (n. 19), 177 (n. 25), 183 (nn. 73, 77)
Line(s) of identity (LI), 39, 53–55
disjoining of, 140
extension of, 140
joining of, 140
retraction of, 140
LI network, introduced, 120
Lingua characterica, 14–16
Locke, John, 6
Logical notation, Peirce's theory of, 11
Logical system vs. calculus, 11, 163–172
Logicism, 15
Logic proper (Peirce), 22
Loop, defined, 102
Loose end, introduced, 40
Löwenheim, Leopold, 15

Mental-model theory (Johnson-Laird), 175 (n. 5)
Metzler, Jacqueline, 175 (n. 7)
Model theory, 14–16
Müller, Ralf, 180 (n. 31)
Multi-modal reasoning, 1, 3, 7–12, 13, 23, 27, 31–35, 37, 49, 56, 163, 171–174
Multiple Carving Principle, 76–80
Multiple-Readings Algorithm, 75

Narayanan, N. H., 5, 177 (n. 27)
Naturalness, 4, 10, 26, 55, 58, 60–61, 72, 73, 77–84, 92–93, 112, 136, 139, 150, 167, 172
Natural deductive system, 3, 60, 80–84, 88, 91–92, 139, 168–169, 173
vs. Alpha system, 80–83
vs. axiomatic system, 168–169
Negation-normal-form reading (NNF reading), defined, 65–66
Negative information represented in EG, 38, 51–53, 61–6311pt

O-area, defined, 64
Oatley, Keith, 176 (n. 16)
Ockham, William, 17
Oddly enclosed, 56, 64
O-jux, defined, 64
Olsen, Len, 181 (n. 40)11pt

Papadias, Dimitri, 176 (n. 19)
Peano, Giuseppe, 16
Pictorialists, 6
Propositionalists, 6
Pseudo-Alpha graphs, 101, 104–105, 123, 191 (n. 59)
Pure grammar (Peirce), 22
Pure rhetoric (Peirce), 22
Pylyshyn, Zenon, 611pt

Quantified information represented in EG, 49, 50, 56–58
Quasi-Alpha graphs, 101, 104–105, 123, 191 (n. 59)
Quine, Williard Van Orman, 15, 173

Representation, 5–10, 13, 19–22, 26
Representing facts vs. represented facts, 27–31
Resemblance, 23–31, 33, 35, 51–52
Roberts, Don, 49–50, 62, 99, 111–118, 175 (n. 2), 177 (nn. 30, 36)
Russell, Bertrand, 14–15

Sanders, Gary, 181 (n. 40)
Schröder, Ernst, 15
Scroll, 72–76
Semiotics, 172, 177 (n. 37)
Sense datum, theory of, 6
Shepard, Roger, 175 (n. 7)
Shimojima, Atsushi, 8, 183 (nn. 73, 77)
Shin, Sun-Joo, 9–10, 177 (n. 29), 182 (n. 63), 183 (nn. 71, 80), 185 (n. 28)
Shorter, J. M. , 176 (n. 8)
Signs, Peirce's theory of, 11, 23–27, 31–35
Simon, Herbert, 7–8, 177 (nn. 20, 21, 23, 35), 183 (n. 73)
Simple formula, defined, 65
Simple graph, defined, 65
Sloman, Aaron, 177 (n. 26)
Sluga, Hans, 178 (nn. 3, 5)
Smart, J. J. C., 176 (n. 12)
Sowa, John, 177 (nn. 30, 34, 36)
Stenning, Keith, 8, 175 (n. 4), 176 (n. 17), 181 (n. 51), 182 (n. 64), 183 (nn. 73, 77)
Sterelny, Kim, 176 (n. 8)
Superfluity of symbols, 165–167, 168
Symbol/symbolic features/symbolicity, 23–35, 48–53
Symbolism/symbolization vs. formalism/formalization, 17–18
Symmetry, 80–81, 83–85, 139, 141, 145–146, 166, 169
Syntactic/derivational history, 4, 81–82, 92, 151–152, 155

Tappenden, Jamie, 178 (n. 8)
Tarski, Alfred, 15
Temporary predicates (Zeman), 102–105
Truth-functional completeness, 73
Tye, Michael, 175 (n. 6)

Unique Readability, 4, 79

Van Heijenoort, Jean, 14–15
Venn diagrams, 9–10, 32–33, 53, 180 (n. 31), 182 (n. 65), 183 (nn. 79, 81)
  primary, 33
  shaded, 32–33, 183 (n. 81)
Visual aspects/distinction/features, 4, 11, 20, 26–27, 31, 55, 60, 63–64, 66, 71, 74, 77, 80–84, 89–93, 97, 99, 106–111, 116–118, 124, 126–130, 137–139, 141, 146–147, 150, 160–161, 163, 167, 169, 173. *See also* Icon/iconic features/iconicity
Visual information, 7. *See also* Visual aspects/distinction/features
Visual language, 9, 177 (n. 28)
Visual reasoning/representation, 9, 13, 19–22, 26
Visualization, 51, 53–58

Wang, Dejuan, 9
Weiss, Paul, 181 (n. 40)
White, Richard, 186 (n. 12), 187 (n. 30)
Whitehead, Alfred North, 15
Wittgenstein, Ludwig, 14–15

Yule, Peter, 175 (n. 4)

Zeman, Jay, 43–48, 50, 55, 99, 100–111, 138, 177 (nn. 30, 36)