Saying It with Pictures: a logical landscape of conceptual graphs
Kerdiles, G.

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Chapter 1

Diagrams and visual information

Figure 1.1: Message sent on the Voyager space probe

Diagrams occur in almost every domain where information is communicated. Examples are numerous, from the geometrical demonstration of Pythagoras theorem to a diagrammatic message sent to possible extra-terrestrial life forms. Part of the success of diagrams as a means of communication is due to the simple way in which complex information is represented. This chapter examines several aspects of the diagrammatic representation of knowledge, with a constant focus on a particular point of interest, conceptual graph diagrams.

To begin with, the occurrence of diagrammatic representations in the history of logic is explored. Peirce’s predicate logic of existential graphs was introduced at the turn of the nineteenth century. It is a particularly important source of inspiration for the development of conceptual graphs.

In computer science, graphical features are extensively exploited on the representational level as well as on the computational level. Some of these applications to artificial intelligence are examined in part two of this chapter.

The essence of diagram processing resides in their prime perceptual effect. In the third part, the cognitive impact of conceptual graph diagrams is explored.

As a preponderant form of communication between humans, natural language should not be disregarded. The last part of this chapter is an attempt to relate
the structure of conceptual graph drawings to the structure of discourse in natural language.

**Diagrams?** Before diagrams are explored, a brief presentation of some terms, that all correspond to graphical representations, is necessary.

*Picture* will be used as a generic term to refer to a graphical representation laid on a delimited zone of a two-dimensional space. *Images* and *drawings* are both pictures. Conventionally, images refer to pictures that can be decomposed into a finite amount of minimal points (e.g., pixels, bitmaps), whereas drawings can be formed of continuous lines. With the assistance of computers for drawing pictures this difference between images and drawings is even more subtle: while a computer picture can be conceived and stored as a drawing (e.g., a vectorial representation), its printing on a screen will be an image limited by the resolution of the screen.

*Diagrams* correspond to schematised drawings in which graphical constituents are associated with a well-defined semantics. Finally, a *graph* refers to an abstract mathematical object composed of nodes connected by edges. It can be physically represented by a diagram.

The specific nature of diagrams will be discussed in the part dedicated to the cognitive impact of conceptual graph diagrams (Chapter 1.3). Before that, some graphical systems that preceded conceptual graphs in logic are presented.

### 1.1 Diagrams in logic

Graphical knowledge representation systems are not a new phenomenon. Eighteenth century Euler circles and nineteenth century Venn diagrams are still popular for manipulating sets and boolean operations. Although most languages of modern logic are textual, it is worth noting that the pioneer research for the foundation of predicate logic was presented in graphical forms: at the end of the nineteenth century, Frege and Peirce independently introduced two graphical systems of first-order logic in an attempt to formalise mathematical reasoning.

#### 1.1.1 Frege's graphs

The language proposed by Frege in his *Begriffsschrift* [Fre79] represents sentences by trees derived from four graphical primitives:

1. "assert A": \[ \top A \]
2. "not A": \[ \bot A \]
3. "B implies A": \[ B \rightarrow A \]
4. “for every $x$, $Px$”:
\[ \exists x \neg Px \]

For instance, “assert that for every $x$, there exists $y$ such that $Pxy$ and $Pyx$” or equivalently, “$\forall x \forall y (Pxy \rightarrow \neg Pyx)$” is represented by

\[ \exists x \forall y \neg Pyx \]

Similar to a tree presentation of a tableau calculus proof, different parts of the representation are distinguished by a disposition on branches. In this particular language, the premiss and the conclusion of an implication occur on different branches. Despite this graphical feature, a representation reads in a linear fashion that clearly resembles its textual counterpart: from left to right and in a depth-first way such that at a branching point, the lowest path—i.e. the premiss of an implication—is first explored.

For predicate logic, Frege’s graphical language has long since been replaced by Peano’s textual notation. Nevertheless, the importance of visual information to computers and robots has brought back another logical system of this period to the research agenda, Peirce’s graphs.

### 1.1.2 Peirce’s existential graphs

Peirce’s languages and calculi have been studied extensively; see e.g., [Pei58], [Rob73], [Thi75] [Shi93] or [Ham98]. It is not the aim to describe these logical systems in detail, but to point out some features of Peirce’s existential graphs that have been adopted in conceptual graphs.

A first feature of Peirce’s graphs, that is fundamental to conceptual graphs, is the role of a primary surface. The sheet of assertion fixes the bounds of the space on which the representations of the different pieces of information that are asserted are disposed. Furthermore, the two dimensionality of the plane is used to represent the conjunction of all drawn components.

For instance, \[ A \cap B \cap C \] represents the conjunction of $A$, $B$ and $C$. The symmetry of conjunction is induced by the fact that there is no predefined order of the conjuncts, as opposed to a textual formula read from left to right.

In Peirce’s graphs, existentially quantified variables are represented as lines connecting the predicate occurrences of which they are arguments.

For instance, \[ A \rightarrow B \rightarrow C \] is equivalent to $\exists x (Ax \wedge Bx \wedge Cx)$. Direct connections through edges will similarly be exploited by conceptual graphs to represent the relationship between predicate occurrences and their arguments.
Finally, negations are represented as closed lines cutting off the negated part from the rest of the assertion.

For instance, \[ A \quad \bar{\quad} \quad B \] is equivalent to \[ A \land \neg B. \]

The place where existential quantification occurs is defined by the outermost zone in which a line that represents the quantified variable in question appears.

For example, \[ A \quad \bar{\quad} \quad B \] represents \[ \exists x (Ax \rightarrow Bx). \]

The interaction between existential quantification and negation will be elaborated upon in Chapter 1.4, where some structures occurring in discourse are highlighted in conceptual graph representations.

Peirce [Pei58] proposed some calculi for propositional (alpha system) and predicate (beta system) logic and ideas of a modal framework (gamma systems). In Peirce’s systems, a conclusion graph follows from a premiss one if and only if the later can be transformed into the former using an appropriate set of graph transformation rules. Although interesting in themselves, these calculi are not particularly adapted to automatised reasoning. Indeed, they are not analytical in the sense that they do not systematically decompose a problem into subproblems, but rest on non-guided rules such as “any graph may be added into a zone enclosed in an odd number of negation lines”. In the light of automated theorem proving, analytical calculi based on graph homomorphisms and analytic tableaux will be studied in this thesis.

1.1.3 Conceptual graphs

Since the late sixties, a graphical knowledge representation formalism equivalent to first-order logic has been developed: conceptual graphs; see e.g., [Sow84, Sow99] for detailed expositions of Sowa’s original systems. The syntax and layout were influenced by a combination of Peirce’s graphs, linguistic dependency graphs and computer science flow charts. On the semantic and deductive side, order-sorted predicate logic and Peirce’s calculi were adopted.

1.1.3.1 Positive information

Departing from the whole first-order language, a sub-formalism for representing positive existential-conjunctive information, simple conceptual graphs, has been carefully studied since Sowa’s book [Sow84]. The language is expressive enough to describe factual information with a slight touch of indeterminacy provided by existential quantification. We may distinguish two graphical aspects related to
the fragment: the representation by graph diagrams and a proof method based on labelled graph homomorphism.

**Representation** Textual symbols of the vocabulary for a conceptual graph language are partially ordered in a predefined classification, called a support in [CM92] or canon in [Sow84].

![Figure 1.2: A support](image)

For instance, the tree in Figure 1.2 represents the information that “every order-sorted logic is a logic and that every logic or AI ontology is a formalism” or in FOL notation:

$$\Phi_0 = \forall x[\text{OrderSortedLogic}(x) \rightarrow \text{Logic}(x)]$$

$$\land \forall x[\text{Logic}(x) \rightarrow \text{Formalism}(x)]$$

$$\land \forall x[\text{AI Ontology}(x) \rightarrow \text{Formalism}(x)]$$

Simple conceptual graphs are bipartite node-edge diagrams, in which square nodes, representing term occurrences, alternate with rounded nodes, representing predicate occurrences. Labelled edges linking a round node (or relation node) to a set of square nodes (or concept nodes) symbolise the ordered relationship between a predicate occurrence and its arguments. Concept nodes are labelled with a concept type and either a constant or a star (standing for an unnamed existentially quantified variable).

![Figure 1.3: A simple conceptual graph diagram](image)

For instance, the graph in Figure 1.3 is a representation of “The CG formalism combines Peirce’s EG logic to an order-sorted logic, which itself combines an AI
ontology to the FOL logic" or the (positive existentially quantified) FOL formula:

$$\Phi_1 = \exists x [ \text{Formalism}(CG) \land \text{Logic}(EG) \land \text{OrderSortedLogic}(x)$$
$$\land \text{combines}(CG, EG, x)$$
$$\land \exists y [\text{AIOntology}(y) \land \text{Logic}(FOL) \land \text{combines}(x, y, FOL)]]$$

Computation Consequence proofs in the simple conceptual graph formalism correspond to labelled graph homomorphisms, called projections (e.g., [CM92]).

The possibility of basing deduction on graph operations has strengthened interest in this alternative to classical calculi of predicate logic.

As in order-sorted logics [SW90], the classification of concepts and relations is exploited in logical consequence. For instance, given the information that “every order-sorted logic is a logic”, represented in the support in Figure 1.2, the information that “the CG formalism combines two (not necessarily different) logics” can be derived from the graph in Figure 1.3 or in FOL notation:

$$\Phi_0 \land \Phi_1 \vdash \exists x \exists y [\text{formalism}(CG) \land \text{logic}(x) \land \text{logic}(y) \land \text{combines}(CG, x, y)]$$

Figure 1.4: A projection from a simple conceptual graph to another one

A proof of this logical consequence is provided by a mapping, pictured in Figure 1.4, preserving both the structure of the source graph (i.e., the conclusion of the logical consequence) and the ordering of labels conveyed by the underlying support.

In subsequent chapters, the computational efficiency of this calculus will be explored for different structural fragments of simple conceptual graphs and extensions to negations and modalities.

1.1.3.2 Negation

For a full predicate logic language, Peirce’s closed negation lines are used to enclose negated zones. For instance, the graph in Figure 1.5 is a representation
1.1. Diagrams in logic

Figure 1.5: Negated regions in a conceptual graph

of "there is a surface such that every negation line delimits a zone which is part of that surface" or

\[ \exists x [\text{surface}(x) \land \neg \exists y [\text{negation line}(y) \land \neg (\exists z [\text{zone}(z) \land \text{delimits}(y, z) \land \text{part of}(z, x)])]] \]

In Chapter 4, we will explore some possibilities and limitations of adapting the projection calculus to the representation of negation in conceptual graphs. In particular, an interlacing of projections and semantic tableaux will be proposed as a predicate logic calculus.

1.1.3.3 Nested conceptual graphs

Figure 1.6: A nested conceptual graph

An additional structural level is obtained by nesting a description (that is itself a nested graph) in concept nodes. To set the ground of the recurrence, the empty
graph that corresponds to the logical constant True, is considered as a nested graph (in order not to overload the picture, empty descriptions of concept nodes in Figure 1.6 have not been represented). The nested conceptual graph formalism, which has a modal flavour, can be exploited to distinguish different groups of localised pieces of information or different levels of knowledge. A "zooming in effect" enables to focus on one local description.

For instance, the nested diagram in Figure 1.6 illustrates a boot failure occurring in the context of the open of my car.

The study of several semantics that can be associated to these nested drawings will be the subject of Chapter 4.3.

1.1.4 Concluding remarks

As graph theory is an extensively studied field in computer science, it is not surprising that many other logical formalisms have chosen graphical features.

Kripke models of modal logics are often represented as labelled graphs and model comparisons, such as bisimulations (e.g., [Ben96]), are naturally defined in terms of graph homomorphisms. Applied modal logics, such as attribute value logics (e.g., [Spa93]) or feature logics (e.g., [Rou97]) exploit the tree structures of their frame languages.

Research by the team of Barwise and Etchemendy at the Visual Inference Laboratory\(^1\) has concentrated on the process of learning logical reasoning by graphical model construction (Hyperproof project\(^2\) and the pieces of software Turing's world and Tarski's world) and on the formalisation of the graphical systems of Euler, Venn and Peirce; see, e.g., [Shi93, Shi95, Ham95, Ham98] and [BE98] for a collection of articles on different aspects of learning and practicing diagrammatic logic.

This introduction to conceptual graphs has exemplified the fact that there is not a single conceptual graph formalism, but a multitude of possible ways to combine and interpret a chosen group of primitive graphical artefacts. Therefore, it is important to identify some criteria, that may guide us in favouring one system over another. In the remainder of this chapter and in the next chapter too, facets of the conceptual graph paradigm are explored under the light of several fields related to logical reasoning, such as artificial intelligence, cognitive science, linguistics and computational logic.

\(^1\)http://www-vil.cs.indiana.edu
\(^2\)http://www-vil.cs.indiana.edu/Projects/hyperproof.html
1.2 Conceptual graph diagrams and artificial intelligence

Diagrams have gained an indisputable importance in computer science and artificial intelligence (AI). They occur in almost every field related to computers, ranging from the actual chips to the abstract representation of knowledge. For example, circuit designs, data structures, algorithms, human-machine interfaces, inheritance in object programming languages or knowledge bases can be represented as trees, graphs, flow-charts or other specific diagrammatic forms.

If research in logic has long been concerned with the distinction between what is provable or not, the application to AI has somehow shifted the focus to determining what kind of reasoning can feasibly be carried out by a computer in a “reasonable” amount of time. Following this line of thinking, adapted representation languages and deductive systems have been invented for automated reasoning. Semantic networks are one example of this.

1.2.1 Semantic networks

Semantic networks, a family of node-edge graphs in AI, have been popular for trying to represent knowledge in a way that is as close to natural language as possible. The proliferation of graphical systems lacking formal semantics has lead to criticism such as McDermott’s “Artificial intelligence meets natural stupidity” [McD76], but also to the development of a family of formal semantic networks originated by Brachman’s KL-ONE system.

Besides the fact that conceptual graph formalisms belong to the class of (formal) semantic networks, they have also borrowed a central notion of classification from artificial intelligence.

- On one hand, the ordering of archived representations of pieces of information, with respect to logical entailment, is relevant to efficient information retrieving from conceptual graph knowledge bases. The logical consequence relationship is sometimes called subsumption and, its symmetrical counterpart, generalisation.

- On the other hand, in AI the classification of the basic terms of a language to describe a particular application domain is called an ontology. Conceptual graph languages exploit such ontologies for efficiency purposes by restraining search spaces to subdomains smaller than the whole domain of a given knowledge base.

1.2.2 Description logics

When artificial intelligence and logic meet, description logics are successful logical formalisms applied to the representation of knowledge. They inherited the two
notions of classification, ontologies and knowledge classification, from semantic network and terminological logic ancestors.

By adopting the semantics of a modal formalism, called hybrid logic (see [Are00] for a detailed analysis of hybrid logics and their relation to description logics), description logics benefit from the efficient computational behaviour of modal logics.

Building on the tree characteristics of models for modal logics, there has been a recent return to graphical features in the syntactic and deductive side of description logics: Baader et al. [BKM99, BMT99] propose a translation of some description logics into a language of trees that is exploited in homomorphism calculi. We shall return to description logics with the complexity study of logical reasoning in Chapters 2 and 3 and with the modal direction taken for nested conceptual graphs in Chapter 4.3.

1.2.3 Contexts in AI

Many researches in AI have questioned the context dependency of information. Giunchiglia and Bouquet[GB97] metaphorically present a context in AI as "a sort of box which is part of the structure of an individual’s representation of the world and which draws a sort of boundary between what is in and what is out". In J. McCarthy’s pioneering work on the formalisation of context (see e.g. [McC87] and [MB97] for a recent survey), such a box is a rich object (a collection of parameters) upon which a representation depends. Typically, a representation can be true in some contexts and false in others. For instance, the piece of information “It is raining” calls for a context of utterance to be interpreted and that context can include among the parameters the time and place of utterance (In the context of Amsterdam, that sentence is often true and particularly on Sunday April 4, 1999). A context, as part of the cognitive state of an agent (the hearer), is used in the interpretation process.

The box metaphor resembles the two kinds of closed lines of the conceptual graph syntax: negation lines and nested boxes. Indeed, from a linguistic point of view, negations play the role of a border line for anaphoric bindings by surrounding a context of discourse interpretation and being permeable in specific conditions. We will elaborate this linguistic argument by examining some structural properties of discourse in Chapter 1.4.

For nested graphs, the meaning of enclosing information into a box can be captured by adapting an applied modal logic: the context logic of Buvac [Buv98]. This point of view will be undertaken in the study of nested conceptual graphs (Chapter 4.3).
1.3. The visual impact of CG drawings

1.2.4 Conclusion

Artificial intelligence is at the crossroad of logic, linguistics, computer and cognitive sciences. Therefore, it is almost impossible to avoid such pluridisciplinary references. Conceptual graphs also dwell at this multicultural crossroads.

Returning to ontologies and without disputing terms, Peirce's graphs, semantic networks and a fortiori, conceptual graphs, make an ontological commitment to graphical items. We now turn to the cognitive impact of these primitive graphical artefacts that, when combined, form conceptual graphs.

1.3 The visual impact of CG drawings

Drawings have many visual properties. Three properties that are particularly pertinent to this study of knowledge representation by conceptual graphs have been chosen to be elaborated upon.

The gestalt feature of diagrams, their faculty to provide an overview of what is represented, is the first visual subject. The perception of the global shape of the information represented results from the different uses of the two dimensionality of drawings. In particular, we distinguish the spatial disposition of pieces of information and the agglomeration of lines to form skeletal structures on which some components hang.

The second visual feature of drawings, that will be discussed in Chapter 1.3.2, is their faithfulness to what they represent. Drawings are often easy to grasp because they are somehow close to what they depict. This property is linked to the expressive power of the drawings, which is relatively limited compared to the high level of abstraction conveyed in sentences of classical linear textual logic languages.

Finally, Chapter 1.3.3 examines how some drawings can provide additional information to the semantic conventions.

![Figure 1.7: The evolution of stock-quotes over time](image)

For example, these three themes appear in a kind of diagram that is com-
monly found in the economic pages of newspapers; stock-charts. Figure 1.7 is a space economical presentation of a large matrix of numbers (i.e., 859 bidimensional coordinates). The diagram stresses the overall characteristics of the data, such as a price that globally follows a downward slope over a three month period. Moreover, the use of conventional scales for price and time facilitates our understanding of the chart. Finally, the intersections of curves are typical pieces of information that are not part of the initial data, but are directly read on the diagram and can be interpreted by investors as signals for changes of tendencies.

We now undertake our first subject in visual matters, the perception of a global perspective of diagrammatic information.

1.3.1 Overview of the information drawn

A generally acknowledged property of diagrams is that they offer a synoptic representation. The possibility of visualising the global structure of a large set of data, takes advantage of our prime perception of visual notions such as density or direction. In particular, the global information perceived in a conceptual graph drawing is a sort of large scale map of the represented relational network. This map has two main components: a partition of the space into areas and a skeletal structure.

1.3.1.1 Partition of the space.

![Figure 1.8: Closed lines and empty-spaces dividing the sheet of assertion](image)

How is the spatial division of the plane on which a conceptual graph is drawn perceived? Outlines of the areas must be found. The most effective symbol to represent a borderline between zones is to draw a line. For example, a Peirce's cut, the representations of a negation in conceptual graphs, is a closed line imprisoning pieces of information into a negated area. Nested boxes in nested conceptual graphs also divide the plane of a drawing into areas symbolising different levels of information. An additional way of defining areas is provided by the perceptual effect of density. In particular, emptiness appears as a discriminating feature
1.3. The visual impact of CG drawings

between zones of high-density. To summarise, the first overall impression of a conceptual graph is some partition of the space into zones containing pieces of information.

1.3.1.2 Spinal structure.

Figure 1.9: Spinal structures

Just as important for the overview is the impact of edges that are perceived as agglomerated into a spine linking different pieces of information. This skeleton does not necessarily have a beginning or end; it is merely central to the different components. The global structure of such a network provides some assistance for navigating the drawing, for moving our point of focus along a path or jumping to an information island. This idea of a support for navigation is reinforced in nested conceptual graphs because they represent different levels of relational structures in one picture: like a road map that includes enlargements for cities provides a representation of a road network at the top level and of some street networks at a lower level. A nested conceptual graph drawing stimulates our visual faculty for discriminating levels and grouping what is connected, in order to safely convey an understandable picture of a complex multi-level network.

To recapitulate, the ingenious human visual machinery capitalises on the perception of density, groups, discontinuities and line structures, to extract, at a glance, the overall information conveyed by drawings. This information can further be employed to guide a search for more details. The overview is a large-scale guide for further in-depth observations. Nevertheless, efficiently using it may require the same kind of training as the reading of a road map does.

Structures formed by lines on the drawings are perceived, but what makes us recognise a shape in a drawing? This is the subject of the next section.

1.3.2 Faithfulness of drawings

Graphics are often labelled as efficient information conveyors because of the facility to understand them. They somehow mirror the information represented. In the case of conceptual graph drawings, two factors influencing this resemblance property can be distinguished: (i) the use of graphical basic components that have a conventional meaning closely related to what they represent and (ii) a deliberately limited level of abstraction.
1.3.2.1 Simple graphical components.

Basic graphical items in conceptual graph drawings are nodes, edges (lines connecting nodes) and closed lines defining frontiers. Different kinds of frontiers can be distinguished by the chosen shape conventions, such as thickness. For instance, lines surrounding negated zones and those defining the outlines of modal worlds are drawn differently.

1.3.2.2 Limited abstraction.

It has been argued that the graphical signs have a standard simple semantics, but what makes a conceptual graph diagram easy to grasp, also lies in limited expressive power of the drawings.

First, there are very few graphical signs used and they all have a clear significance. This fact implies the need for only few simple rules of interpretation, which is certainly to the advantage of the reader.

A second factor of simplicity is the direct nature of the graphical message: what is left unsaid is really not represented. The sole exception to this rule is the use of the indefinite marker *, a place holder for an indefinite object. It corresponds to an existentially quantified variable in a textual logic language. Nevertheless, other connectives commonly used in logic, such as universal quantification, disjunction and implication, are left out of the picture. These connectives have the disadvantage of summarising complex information into single symbols. For example, universal quantification conveys the message that all the individuals living in the represented model have some property. In other words, it abstracts some information to the level of the whole population instead of directly showing facts for each individual. It conveys a large amount of information with very few syntactic items and the expansion of the compacted information is left to the reader. Similarly, disjunctions and implications call upon the reader's interpretation process to build several alternative models at once.

Existential quantification alone does not have these drawbacks. It provides the reader with an unnamed object, but guides the interpretation process by showing a one-on-one correspondence between the syntactic objects and the represented ones. Hammer [Ham95] has studied a similar type of matching for different forms of diagrams; the isomorphism thesis. [AB96] and [CSO94] discuss how the low level of abstraction in some graphical representations of logical sentences can influence a logic learning process.

To summarise this point on the faithfulness of conceptual graph drawings to what they represent, we can relate this advantage to the small amount of graphical signs used, and to their intuitive meaning. A drawing with a relatively low level of abstraction presents a one-on-one correspondence to the represented.

Until this point, it has been argued that drawings have prominent perceptual features. Some of them, like resemblance, have an obvious semantic use. Others,
1.3. The visual impact of CG drawings

like the overview spinal structure, have a less direct meaning. The study of the semantics of these graphical properties that provide extra information, is the subject of the next section.

1.3.3 Additional information

Graphical features can be perceived. Some of them are given meaning according to defined interpretation rules. For instance, an edge between two nodes is known to represent a binary relationship between two objects. Other features fall outside the interpretation conventions, but are nevertheless informative if associated with a meaning.

This section is devoted to the study of the additional information that is perceived from conceptual graph diagrams. Basic semantic conventions give rise to new interpretation rules. Two themes may be distinguished. The first concerns some generalisation of a particular convention to a larger domain. The second is related to the modification of a particular convention by some typically graphical feature, which initially had an obvious intuitive meaning.

1.3.3.1 Generalisation of semantic conventions.

In the previous section, conceptual graph drawings were observed to be composed of a small amount of distinct graphical signs (nodes, edges and closed lines). It has been argued that the small amount of signs is a cognitive strength, as only very few interpretation conventions are required to understand a drawing.

Among these graphical signs, the edge has a preponderant role, one of representing relational information. An edge is a local object. It connects its extremities at a particular place in the representation. However, this role of representing **direct connectedness** is intuitively generalised to the level of the whole representation. Agglomeration or concatenation of edges convey a global notion of **indirect connectedness**.

This is first visible in the graphical representation of the vocabulary classification. Edges correspond to implications and paths provide their transitive closure (e.g., from reading the branch on the right-hand side in Figure 1.2, we can conclude that any order-sorted logic is a formalism).

Similarly, in conceptual graph diagrams, the notion of indirect connectedness has a meaning of relatedness. Relatedness is useful in applications like information retrieval, enabling the connection of objects that are not in direct relation to each other. Interpreting distinct connected compounds as unrelated pieces of information provides a simple guide for breaking down a problem into smaller sub-problems that can be solved independently of each other. Salvat’s experiments [Sal97] in an application of a meta-resolution rule for a language of conceptual graphs, which includes implication, have shown that applying this obvious selection function (i.e., if possible, take a successor in the connected compound at
stake) does often reduce the number of backtracks. Tree structures will also prove essential to efficient calculi.

To summarise, the meaning of edges, as being representations of connectedness, can be generalised in a weaker significance for paths: one of relatedness.

1.3.3.2 Interaction of graphical effects and semantic conventions.

From the basic interpretation convention, the meaning of the occurrence of two distinct pieces of information on the same plane (or area in the presence of Peirce's cuts) is known: the conjunction of the components is represented. However, this significance can be strengthened by a perceptual effect, namely density. Indeed, spatial grouping of pieces of information can corrupt the neutral conjunctive information and represent a second form of relatedness. As noted above, density capitalises on innate human perception to make salient information relevant. This second notion of relatedness can prove useful in order to organise the presentation of information in packets. These are groups either conveying a semantic message or simply being a practical help (for example, a division of the space between multiple users of a knowledge base).

Another use of density occurs in homomorphism proof drawings. The significance of a proof diagram can be enriched by information about the location of the pieces of information that are utilised on the density map.

We have seen that information that is not considered in the basic semantic conventions can be perceived from graphics. The meaning of this information is intuitive because it results either from the generalisation of the semantics of local items to a larger scale, or from the interaction of meaningful graphical effects with basic semantic rules.

In his thesis [Shi95], Shimojima studies a related phenomenon: free-rides. Free-rides are additional information resulting from the matching of graphical constraints with some constraints of the represented. The derived meaning postulate, read from the transitive closure on support paths, would fit this definition, but free-rides are more specific. The additional information can be directly interpreted using the basic semantic rules of the graphical system. The phenomena examined in this section are of a slightly different nature. They concern extra information which is obtained by derived interpretation rules.

1.3.4 Conclusion

In this survey of some visual properties of conceptual graph drawings, which by no means claims to be exhaustive, three main themes that participate in the cognitive efficiency of diagrams have been distinguished.

The first issue concerns the gestalt feature of diagrams and their faculty of offering a synoptic representation of both the partition of the information space and the spinal structure linking pieces of information.
The second theme is an attempt to recognise the features that make conceptual graph drawings faithful representations of relational structures. The nearly iconic nature of the graphic components and the limited abstraction represented in drawings have been identified as reasons for this mirroring property.

Finally, the usefulness of perceptible additional information is linked to the intuitive adaptation of local semantic conventions to large-scale graphical effects.

By using expressions such as *easy perception*, the ingenuity of the human visual machinery is taken for granted. However, it is far from clear how complex basic perception operations would function for an artificial visual machine. Despite the lack of formal visual models for efficiency measures, we are not totally clueless. In the forthcoming chapters, the use of classical complexity theory for textual translations of graphs will provide a first handle in a formal attempt to answer the question.

Closer to the previous cognitive concerns than computation models, the study of relationships between natural language and conceptual graphs is the next subject focused on.

1.4 Conceptual graphs and the structure of discourse

Natural language is the pervasive medium for cognitive activities. Despite the fact that it is transcribed in a linear way with the use of symbols (letters or characters), a discourse shares some characteristics with conceptual graph diagrams.

Primarily, a discourse is structured. First, we will examine some correspondences between the binding of term occurrences in the process of conceptual graph construction and anaphoric phenomena in natural language.

Some linguistic theories also use pictures. In a second section, the features that bring a specific linguistic formalism, discourse representation theory, and conceptual graphs closer will be considered.

Finally, by viewing conceptual graphs and discourse representation structures together, the same innovations may be applied to both. As an illustration, so-called dynamic interpretations will be quickly discussed.

1.4.1 Bindings in conceptual graphs and discourse

It has been observed that the linguistic counterparts of logical connectives behave as structuring items in discourse, with different permeability properties to pronominal coreferences.

For example, the conjunction of two sentences can be expressed in English by the use of the term ‘and’ or just the concatenation of these sentences: “A man entered. He took a chair.” or “A man came in and he took a chair.”. In the second sentence, a pronominal reference to an object introduced in the first
sentence is possible. On the other hand, the use of negation appears to block the possibility of such binding: It seems unacceptable to continue the sentence “It is not the case that a man came in.” with “He took a chair,” as the pronoun ‘he’ cannot be resolved by any object previously introduced in this piece of discourse. Conceptual graph construction rules present similar properties of bindings.

1.4.1.1 Conjunction of conceptual graphs

Conjunction in conceptual graphs obeys two simple rules:

(i) the conjunction of two pieces of information is represented by their juxtaposition on the sheet of assertion

(ii) in the absence of negation line, a concept node can be made coreferent to another concept node occurring in the same graph.

For example, in Figure 1.10, the pronoun ‘he’ is represented by a concept node labelled with the marker ‘?’ symbolising that it needs to be made coreferent to another accessible concept node. After the two graphs have been juxtaposed, the concept node ‘man:*’ becomes available for coreference to the node labelled with the question mark.

The resolution of the anaphoric binding is a problem that is beyond the scope of this thesis. What is important is the fact that, after juxtaposition of the two initial graphs, the representation of the indefinite noun phrase ‘a man’ becomes available for coreference to the representation of the pronoun ‘he’.

According to the second rule, in the absence of negation, the application of coreference is, in principle, free for any pair of concept nodes in a graph. Of course, one can add some additional constraints. For instance, it could be requested that two nodes made coreferent should have concept types sharing a common subtype.
It could also be forbidden to link two nodes labelled with different constants, respecting a common assumption for many AI systems that different constants represent different individuals.

If conjunction in conceptual graphs is, as conjunction is in discourse, permeable to coreferences, what about negation?

1.4.1.2 Negation

Closed lines, representing negations in conceptual graphs, delimit zones that are included in the outermost zone: the sheet of assertion. These zones and frontiers remind the metaphoric image of "context as a sort of box" discussed in Chapter 1.2.3.

By construction, negation lines do not intersect each other. Thus, the nesting of zones has the structure of a tree whose root is the sheet of assertion. This partial order is called domination. A zone is said to dominate another zone if the later is included in the first one. We may now restate the rule for coreference as follows:

(ii') a concept node can be made coreferent to another concept node occurring in a dominating zone of the same graph.

Figure 1.11: Negation boxes are permeable for coreferences from outside-in

Figure 1.10 is an example of concept nodes occurring in the same zone (the sheet of assertion). Let us consider an example with a negation: "A man entered. It is not the case that he took a chair." In Figure 1.11, the concept node representing the pronoun 'he' occurs in a zone dominated by the sheet of assertion, in which the concept node for 'a man' occurs. According to the rule (ii'), we are allowed to bridge those two nodes with a coreference link.

Conversely, the binding of the pronoun 'he' is not resolvable in the (unacceptable) discourse "It is not the case that a man came in. He took a chair.".
Chapter 1. Diagrams and visual information

1.4.2 Discourse representation theory

Discourse representation structures (DRS) combine nested boxes with notations of predicate logic for representing the structure of natural language sentences. We refer the reader to Kamp’s foundation article [Kam81] and to the extended treatment of DRT in [KR93]. [BB98] is a comprehensive introduction to DRT and some background in computational linguistics.

In a language where all connectives are expressed in terms of conjunction, negation and existential quantification, a DRS is a box divided into two parts. These parts are a set of discourse referents and a set of conditions where a condition has either the form of a predicate logic atom or the negation of a DRS. Discourse referents in the first part of a box correspond to existentially quantified variables which are accessible to the conditions in the second part and, by transitivity of nesting, to all conditions occurring deeper in the nesting.

For example, “A man entered. It is not the case that he took a chair.” can be represented by the DRS:

| x | \(\text{man}(x)\) | \(\text{entered}(x)\) | \(\neg\) | \(\text{chair}(y)\) | \(\text{took}(x,y)\) |

There is a notable difference between conceptual graphs and DRSs. By inheriting Peirce’s lines of identity and contrary to DRSs, conceptual graphs provide a notation of equivalent expressive power that is free of variables. This feature is relevant in theorem proving. Indeed, efficient methods for constructing proofs like free-variable tableaux or resolution require pure representations (representations in which a variable is not quantified twice). Hence, a renaming pre-process can be required. By avoiding variable names, CGs are always pure. However, the absence of variables is mostly relevant for an incremental construction of the representations. If the representations of the constituents of a text can be drawn independently, then building a representation of the whole text consists in merging the representations of the constituents.

To illustrate a problem associated to merging, we present an example from [Eij98]. A DRS for “A man entered. A woman entered.” can be obtained by
merging the following two DRSs:

\[
\begin{array}{c}
\text{man}(x) \\
\text{entered}(x)
\end{array}
\oplus
\begin{array}{c}
\text{woman}(y) \\
\text{entered}(y)
\end{array}
\]

But merging is not defined in case of a variable clash:

\[
\begin{array}{c}
\text{man}(x) \\
\text{entered}(x)
\end{array}
\oplus
\begin{array}{c}
\text{woman}(x) \\
\text{entered}(x)
\end{array}
\]

In discourse representation theory, the problem of variable clashes is solved by always building the representation of a new sentence in the context of an existing DRS. Another solution consists in first renaming the variables occurring in different DRSs before merging them. Van Eijck [Eij98] proposes another alternative: the replacement of variable names in DRSs by De Bruijn’s indices. Conceptual graphs do not make use of variables, so variable clashes cannot occur and two graphs can always be merged by only juxtaposing them. The simplicity of this safe merging operation is an attractive feature of Conceptual Graphs.

Syntactically, DRSs and conceptual graphs share the same structure for displaying the representations and the same scoping rules for existential quantifiers. Furthermore, semantically, both formalisms rely on a notion of embedding of pictures into classical models for predicate logic. These similarities may be exploited to bridge the differences. On one hand, discourse representation theory has achieved numerous results in the study of natural language phenomena and being able to adapt these results would be beneficial to conceptual graph theory. On the other hand, conceptual graph theory has achieved some computational results, based on the use of graph calculi, which could serve the interest of the deductive side of DRT.

Finally, there is an alternative semantics to the embedding of DRSs into classical models, which brings us to our next subject, dynamic semantics.

### 1.4.3 Dynamic view on conceptual graph semantics

It is often assumed that discourse interpretation is related to a dynamic process of discourse context evolution. When successively uttered sentences are processed by a hearer, they bring successive changes into the interpretation context of that hearer. When a simple conceptual graph is asserted, it introduces two kinds of information: a relation occurrence between concepts provides some factual information and a concept node introduces in the context an item that is available for further references. A semantics for conceptual graphs could take into account these two kinds of information and follow the way paved by dynamic semantics [GS91].

#### 1.4.1 Example. Imagine the following situation: the vocabulary is composed of three individual markers \(a, b\) and \(c\), and two relations \(\text{entered}\) and \(\text{spoke}\). Let \(M = (D = \{A, B, C\}, F)\) be a classical model such that \(F(a) = A, F(b) = B, F(c) = C, F(\text{entered}) = \{A, B\}\) and \(F(\text{spoke}) = \{(A, B)\}\). We are at the
beginning of a conceptual graph discourse, only aware of our formal vision of the
world, \( M \). Our discourse context is empty:

\[
\langle \emptyset \rangle
\]

Suppose that we are told that “Someone entered.”
We process this information by creating a record for that person in our information state and associating to it all objects provided by the model, which satisfies the graph utterance: \( A \) and \( B \).

Our new discourse context is:

\[
\langle *, A \rangle
\]

If we are now told that “He spoke to b.” We can resolve the pronoun ‘he’ to the sole item in our discourse context and, given our model \( M \), eliminate the possibility that the person at stake is referring to \( B \).

If the last utterance had been “He spoke to a.”, no possible interpretation for the context item would have remained and we, as hearer, would have ended up in an ‘absurd state’. A possible escape of would then be a rejection of (pieces of) the discourse or the revision of previously accepted information.

The meaning of a graph is given by the change its assertion brings into a context. The example only sketches the possibility of storing terms (and their possible interpretation) as contextual information. However far richer contexts could be conceived. For example, a context could also contain relational information between the stored discourse items with the consequence of a shift from the notion of test in a model to a notion of model construction.

The dynamic view of conceptual graph interpretation can be related to the process of constructing a complex representation by successive updates, each of them refining a previous representation with additional knowledge. This procedural view is common to conceptual graph incremental construction rules and to calculi based on successive graph derivations (see e.g., [Sow84] or [CM92]).

1.4.4 Concluding remarks

The aim of this brief jaunt into natural language semantics was twofold. First, to relate the layout of conceptual graph drawings to some structural properties of discourse and in particular, to the interaction of boolean connectives and quantifiers. Secondly, to provide the first steps toward stronger interactions between conceptual graph theory and linguistic theories. Such an exchange could benefit both sides. On one hand, it would strengthen the foundations of conceptual graphs in artificial intelligence, as natural language remains the most common way of communication between humans. On the other hand, such a link could
provide new insights into efficient computation for theories that are aimed at automated natural language processing.

However, despite the promising perspectives of further connections between conceptual graphs and natural language processing, a longer exposition of linguistic features would bring us out of our main trail, which is concerned with the use of conceptual graphs for efficient logical reasoning. We refer the reader to [KR93], [BB98], [GS91], [GSV96] and [Ben96] for deeper insights into discourse representation theory and dynamic semantics and their relation to logic and computer science. For a formal proposition of dynamic interpretation in conceptual graphs, see [Ker99a].

1.5 Conclusions

Languages of modern logics are essentially textual. It comes as no surprise that knowledge representation systems often adopt textual notations similar to those of their underlying logics. However, on the semantic side, sentences of these languages are interpreted with respect to structures. It is somehow paradoxical that these languages describe and refer to structured information by means of linear text, while linking sentences and structures is left to the interpretation process alone. To represent structured knowledge, it seems sensible to import as much as possible the object structure in the layout of the representation language. Conceptual graphs take this path by combining textual labels with node-edge drawings.

Along this introductory chapter, we have described influential graphical ancestors of conceptual graphs at the historical foundations of modern logic.

The importance of graphics and graphs to logic and its application to artificial intelligence has become a fact with the development of robotics, artificial vision and computers that are now provided with great graphical abilities. For instance, almost no operating system would now be commercialised with a single purely textual interface, no web-browser would be limited to the display of textual information. Pictures have found applications in logic and applied logics for educational purposes (e.g., the Hyperproof project), for representational and computational purposes (e.g., semantic networks, description logics or conceptual graphs).

Another reason for this regain of interest in pictures, which, after all, have always been used for every day communication since prehistoric times, might simply come from their cognitive power: perspicuity and efficiency. Such features appear in conceptual graphs under different aspects: primary gestalt view, distinction of structural components, simplicity of interpretation and visual derivative meanings of the representations. With the experience of working with conceptual graphs, acquired in the subsequent chapters, we will come back to this cognitive theme in the concluding chapter of the dissertation.
Finally, we have explored a linguistic aspect of conceptual graphs: their faithfulness to natural language structures. The perspective of rich interaction with fruitful linguistic theories has just been scratched and is promising for further research. In particular, the dynamic turn in semantics does not only confine itself to natural language semantics, it is also of great interest to computer science for which the dynamic notion of process is central. This brings us to the theme of our next chapter: the computational aspect of logical reasoning.