Toward a Formal Theory of Information Structure

Jerry R. Hobbs and Rutu Mulkar-Mehta
USC Information Sciences Institute
Marina del Rey, California
hobbs@isi.edu, rutu@isi.edu

1 Introduction

Search engines today are very good at finding information in text resources. But very often the best answer to a question is in a diagram, a map, a photograph, or a video. For example, consider the questions

> What is the Krebs cycle?
> How has the average height of adult American males varied over the last 100 years?
> How did the Native Americans get to America?
> What does Silvio Berlusconi look like?
> What happened on September 11, 2001?
> When will the various tasks on this project be completed?

The answer to the first should be a process diagram, the second a graph, the third a map with routes indicated, and the fourth a photograph. For the fifth, a news video clip might be the best answer. The answer to the last might best be presented in a Gantt chart.

Search engines are very much poorer at finding this kind of information, and generally they do so by looking at the associated text. We would like to have this kind of information encoded in a fashion that makes it more retrievable, in part by describing it better, and in part by expressing "coreference" relations among material presented in different media. For example, a diagram may bear a kind of coreference relation with a text segment in the document it is a part of. There may be a coreference relation between an object in a photograph and a noun phrase in the caption. In a news video clip we have analyzed, a woman refers to sending her "four children" to local schools, as she is standing by a wall covered with family photographs; there is also a coreference relation here.

The aim of this paper is to present a formal theory of the structure of information that will support a variety of statements about documents in various media, their internal structure, and how they function in the world at large. We begin by sketching an approach to anchoring symbolic systems in human cognition and discuss various levels of intentionality that occur. We then consider compositionality in different symbolic systems, and the sometimes complex coreference relations that arise from that. This theory is the basis of a program for translating natural language into logical form, and this is described. Then the theory is applied to the specific case of diagrams as information-bearing objects, and a logical theory of Gantt charts is constructed as an illustration. Finally there is a discussion of issues raised with respect to various modalities and various manifestations of symbolic artifacts.

Thus, Sections 2 and 3 on Gricean nonnatural meaning captures how single words or other atomic symbols convey information. Sections 4, 6 and 7 describe how multiple words or other symbols combine to convey more complex information.

Symbol systems evolved. One of the key tasks in explaining the evolution of something is hypothesizing incremental steps in its development where each new advancement confers an advantage. Hobbs (2006) describes just such a plausible sequence of steps for Gricean nonnatural meaning and natural language syntax. Here we will sketch this account very briefly.

2 Grounding Symbols in Cognition

In this paper we will assume that we have a coherent notion of causality, as in Hobbs (2005), and change of state, as in Hobbs et al. (1987), as well as a theory of commonsense psychology at least rich enough to account for perception, planning and intentional behavior, and what we here call "cognizing", that is, taking some cognitive stance toward, such as belief, thinking of, wondering about, and so on. We will refer to the contents of thoughts and beliefs as "concepts", a general notion that subsumes propositions (Gordon and Hobbs, 2003), but also includes nonpropositional concepts like "dog" and "near", images, vague feelings of apprehension, and so on. We will assume the "ontologically promiscuous" notation of Hobbs (1985a), but for typographical convenience, we will abuse it by using propositions as arguments of other propositions, where a proper treatment would reify the corresponding eventualities and use those as arguments. Some of the inferences below are defeasible, and thus the underlying logic must support a treatment of defeasible inference. There are many frameworks for this, e.g., McCarthy (1980) and Hobbs et al. (1993). To minimize notational complexity, defeasibility is not made explicit in the axioms in this paper.

The basic pattern that symbols rest on is the perception of some external stimulus causing an agent to cognize a concept.

(1) $cause(perceive(a, x), cognize(a, c))$

where a is an agent, x is some entity, and c is a concept. x can be any kind of perceptible entity, including physical objects and physical properties, states, events and processes, and, as we will see later, more abstract entities as well. That is, we can perceive a ball, its roundness, and the event of someone throwing it. Among the states that can be perceived are absences. Seeing that someone's car is not in his garage can cause someone to believe he is not at home. Silence, or absence of speech, can often carry very significant meaning.

The concept c may often be hard to put into words, such as the concepts triggered by music or an aesthetic design.

This pattern covers the case of a cloud reminding someone of a dog, where there is no external causal connection between the stimulus and the concept, and the case of smoke making one think of fire, where there is a causal connection, and the intermediate case of an association that has been established by practice, as in a dinner bell making one think of food.

Some concepts are tied in such a way to the entity perceived that they can be called the "concept of" the entity. We could introduce *conceptOf* as a function mapping from the entity to the concept, but since the predicate *cognize* always takes a concept as its second argument, it is simpler to build the coercion into the predicate *cognize*. Thus, if e is an entity, $cognize(a, e)$ says that agent a cognizes the concept of e. The key relation between entities and their concepts is that perceiving the entity causes the agent to cognize the concept of the entity.

(2) $cause(perceive(a, e), cognize(a, e))$

It is important to note, however, that perception can trigger many concepts and that not everything that is cognized needs to be what is perceived. Perceiving a bell can cause an agent to cognize food (as well as the bell). This makes symbols possible.

Where the concept cognized is propositional, we could talk about its truth in the world. That is, it is not only true that e is cognized; it also holds in the real world—$holds(e, w)$. However, this will play no role in this paper. The meanings of symbols will be strictly in terms of the concepts they invoke in the recipient.

Communication begins when another agent presents an entity causing the first agent to perceive it.

(3) $cause(present(b, x, a), perceive(a, x))$

For an agent b to present something to a is for b to cause it to be within the range of a's perception, and this causes a to perceive it.

The recipient agent a must of course be capable of cognition. A greater range of sending agents b is possible. A car that beeps when you don't fasten your seatbelt is an agent b that is presenting a signal x for the driver to cognize. It is also possible for collectives to be the sending agent, as in jointly authored documents such as the Constitution of the United States. The agents may or may not exhibit intentionality. Humans do, as do organizations of humans, whereas simple artifacts merely reflect the intentionality of their designer. Sufficiently complex artifacts may exhibit intentionality.

Causality is defeasibly transitive, so Rules (1) and (3) can be combined into the defeasible causal pattern for appropriate x's and c's:

(4) $cause(present(b, x, a), cognize(a, c))$

That is, if b presents x to a, it will cause a to cognize the corresponding concept c. For example, a car beeps and that causes the driver to hear the beep; hearing the beep causes the driver to remember to fasten his seatbelt. So the beep reminds the driver to fasten his seat belt.

We will refer to the entity presented (x) as the symbol and to the concept evoked (c) as the meaning of the symbol.

A car that beeps cannot be said to have beliefs. Monkeys that emit alarm cries at the sight of a snake or a leopard may or may not be usefully described as having beliefs about the threat. Higher primates probably do have such beliefs, and humans certainly do.

Belief introduces another level of complexity to meaning. Someone sees a fire and runs to warn his friends. The friends don't see the fire themselves, but they interpret his presentation of the alarm as caused by his belief that there is fire, and this causes them to believe it. Belief acts as a kind of carrier of information across space and time.

A theory of belief is useful in social animals for independent reasons, but once they have such a theory, it can enrich their communication. It allows them to reason in formula (4) about why b presented this information, normally, because b believes c, then to reason about why b came to believe this, and then to assess whether they ought to believe it too. A theory of belief allows agents to interpret the content of utterances as mistakes.

3 Intention and Convention in Communication

Presentation by an agent can involve several levels of intentionality, and the perception can involve several levels of recognition of intentionality. First, the presentation can be entirely unintentional, as, for example, when someone conveys their nervousness by fidgeting or shaking their leg. In an abductive account of intelligent agents, an agent a interprets the environment by telling the most plausible causal story for the observables in it. Here a knows nervousness causes fidgeting and the most plausible causal story is that b's fidgeting is because b is nervous. When b says "ouch" and a infers that b feels pain, the account is exactly the same.

When the presentation is intentional, the presenter's goal is to cause the perceiver to cognize something. The presenter's intention need not be recognized. For example, a professor may keep the door to his office closed to lead students to believe he is not in, without wanting them to recognize his intention to communicate that.

Intention is recognized when it is part of an observer's explanation that an event occurs because the agent of the event had the goal that it occur. Defeasibly, agents do what they want to, when they can.

(5) $goal(g, b) \wedge executable(g, b) \supset cause(goal(g, b), g)$

(All axioms are universally quantified on the variables in the antecedents of the highest-level implication). If g is a goal of b's and is executable by b (or achievable by an executable action), then its being a goal will cause it to actually occur. We won't explicate *executable* here, but it means that g is (achievable by) an action of which b is the agent, and all the preconditions for the action are satisfied.

When an observer a uses this causal rule, he is recognizing the intention that lies behind the occurrence of the event.

It is most common in human communication that the intention is recognized. Agent b knows that presenting x causes a to perceive x, which causes a cognize concept c. b has the goal that a cognize c. So that causes b to present x. Agent a comes up with exactly this causal explanation of b's action of presentation, so not only does a cognize c; a also recognizes b's goal that a cognize c.

This recognition relies on agents' knowing a defeasible rule that says that

(6) $goal(g_1, b) \wedge cause(g_2, g_1) \supset goal(g_2, b)$

That is, if an agent b has a goal g_1 and g_2 tends to cause g_1, then a may have g_2 as a goal as well.

In the case of communication, g_1 is $cognize(a, c)$ and g_2 is $present(b, x, a)$. The recipient observes the event of the presenting, uses axiom (5) to infer abductively that it is intentional, and uses axiom (6) together with schema (4) to recognize that b intends for a to cognize c.

We can get to full Gricean nonnatural meaning (Grice, 1989) by decomposing Rule (6) into two rules:

(7) $goal(g_1, b) \wedge cause(g_2, g_1) \supset goal(cause(g_2, g_1), b)$
(8) $goal(cause(g_2, g_1), b) \supset goal(g_2, b)$

That is, if an agent b has a goal g_1 and g_2 tends to cause g_1, then b may have as a goal that g_2 cause g_1. Moreover, if an agent b has as a goal that g_2 cause g_1, then b has the goal g_2.

When g_1 is $cognize(a, c)$ and g_2 is $present(b, x, a)$, a uses axioms (7) and (8) to explain the presentation; then a will recognize not only b's intention to have a cognize c, but also b's intention that a do so *by virtue of* the causal relation between perceiving x and cognizing c. This is the definition of Gricean nonnatural meaning.

In order for this sort of communication to work, it must be mutually known between a and b that perceiving x causes cognizing c. Mutual belief can be characterized by three properties. First, if a group mutually believes something, each of the members believes it. Second, if a group mutually believes something, then it mutually believes it mutually believes it; this allows one to step up to arbitrary levels of embedding of belief inside belief. Third, mutual belief can be successively approximated by shared beliefs. That is, if we both believe something, we probably but not necessarily mutually believe it; if we both believe we both believe it, it is even more likely that we mutually believe it. The more levels of embedding we add on, the more difficult it is to construct examples where mutual belief fails, and thus, the more likely it is that mutual belief holds.

Communicative conventions are causal rules having the form of (7) and (8), where g_1 is the presentation of a symbol and g_2 is the act of cognizing its meaning. Communicative conventions grow up in different social groups and become mutually believed within the groups. Thus, the structure of a communicative convention is

$$mb(s, cause(present(b, x, a), cognize(a, c))) \wedge member(a, s) \wedge member(b, s)$$

for a specific x and a specific c. That is, a social group s that agents a and b are members of mutually believe the causal relation between presenting x and cognizing c. For example, x might be a red flag with a white diagonal, s might be the community of boaters, and c the concept that there is a scuba diver below.

These communicative conventions can originate and take hold in a group in many different ways. The culture of a group consists in large part of a number of such rules.

Note that there is nothing particularly advanced about the arbitrariness of the symbol x. That is already there in the most primitive stage, in the connection between the bell and the food.

This completes the sketch of how the meaning of atomic symbols can be grounded in a theory of cognition: in our scheme, x is a symbol that means or represents c to a group of agents s. In an elaboration of Pease and Niles (2001) we can express this as

$$means(x, c, s)$$

We will leave out the third argument in the development of the theory of diagrams below; the community is simply the set of people expected to be able to understand the diagrams.

Like a folk theory of belief, a folk theory of intention is a useful thing for a social animal to have. It allows individuals to predict the behavior of other individuals with some degree of accuracy. A stark example of this point is the difference in our reaction to walking down a sidewalk a few feet from a bunch of cars hurtling past at 50 miles an hour and our reaction to standing on a mountain slope a few feet from a landslide. The difference is entirely a matter of intention. Once a theory of intention has evolved, it can be deployed in the construction of a much richer theory of meaning, as described here, and for example allows agents to interpret utterances as deliberately deceptive.

We next turn to how more complex symbolic objects convey more complex meanings in different modalities.

4 Composition in Symbol Systems

An atomic symbol, i.e., one that does not have interpretable parts, corresponds to some concept. Atomic symbols can be composed in various ways, depending on the type of symbol system, and the meaning of the composition is determined by meaning of the parts and the mode of composition. These composite elements can then be components in larger structures, giving us symbolic structures of arbitrary complexity. This is illustrated in the commuting diagram of Figure 1.

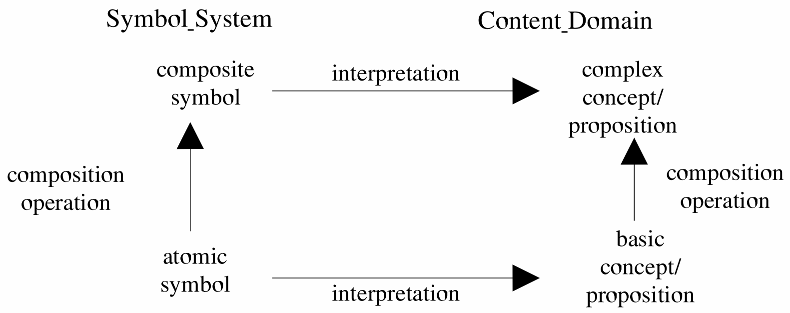

Figure 1: Composition in symbol systems

Composition in symbol systems occurs when entities x and y, meaning c_1 and c_2 respectively, are presented with a relation R_1 between them, where R_1 conveys the relation R_2 in the target domain. Thus, we have three causal relations.

$$
\begin{aligned}
&\mathit{cause}(\mathit{present}(b, x, a), \mathit{cognize}(a, c_1))\\
&\mathit{cause}(\mathit{present}(b, y, a), \mathit{cognize}(a, c_2))\\
&\mathit{cause}(\mathit{present}(b, R_1(x, y), a), \mathit{cognize}(a, R_2(c_1, c_2)))
\end{aligned}
$$

The relation $R_1(x, y)$ can be thought of as just another entity in the symbol system, so it is subject to full Gricean interpretation just as atomic symbols are, and it can similarly be involved in the conventions of some community.

With respect to the concepts invoked, we will confine ourselves here to propositional concepts. The advantage of having a flat notation in which anything can be reified is that when composite concepts are constructed, we can view this as simply a conjunction of what is already cognized with the new relations conveyed by the composition.

Speech (and text as spoken) takes place in time, so the only compositional relation possible on the left side of the diagram is concatenation. In discourse beyond the sentence, concatenation generally conveys a coherence relation based on causality, similarity, interlocking change of state, or figure-ground (Hobbs, 1985b). In a sense, the adjacency of the segments of discourse says that the states or events described in each segment are related somehow, and the hearer has to figure out how. The particular coherence relations that occur are the most common kinds of relations that typically obtain among states and/or events.

Within sentences, the composition of smaller units into larger units conveys primarily a predicate-argument relation between the meanings of the components. Thus, when we concatenate "men" and "work" into "men work", we are indicating that the referrent of "men" is an argument or role-filler in the event denoted by "work". This view of syntax as conveying predicate-argument (and modification) relations through adjacency of constituents is elaborated in Hobbs (2005), in which an extensive grammar of English is developed in a manner similar to the Head-driven Phrase Structure Grammar of Pollard and Sag (1994). Section 6 describes an implementation of this idea for generating logical forms for natural language sentences.

Protolanguage (Bickerton, 1990) is more like discourse. Two words, phrases, or larger segments of natural language are adjacent, and it is up to the hearer to figure out the relation between them. Children's language in the two-word phase is an example of this; "mommy sock" might mean that this sock belongs to mommy, or that mommy should put this sock on or take it off the child, or any number of other relations. The language of panic is another example of protolanguage, for example, when someone runs out of his office shouting, "Help! Heart attack! John! My office! CPR! Just sitting there! 911! Help! Floor!" we understand the parts and hypothesize the relations that build them into a coherent scenario. The nominal compound can be seen as a relic of protolanguage in modern English; the relation between the two nouns might be a predicate-argument relation, as in "language origin", but it might be any other relation as well, as in "turpentine jar".

Hobbs (2006) tells an incremental story of the evolution of syntax that begins with protolanguage. Then constraints are added to the effect that adjacency is interpreted as predicate-argument relations. Further steps extend this to discontinuous elements, as in "Pat is likely to go," where "Pat" is the subject of "go." Long-distance dependencies are seen as language-specific constraints on possible predicate-argument relations between noun phrases and ajacent clauses.

Tables are another kind of complex symbolic object. In tables, the elements in individual cells refer to some concept. The manner of composition is placement of these cells in a vertical and horizontal arrangement with other cells. Generally, the aggregate represents a set of relations: The item in a cell that is not the first in its row stands in some relation to the first element in the row. The relation is the same for all elements in that column, and is often explcitly labelled by a header at the top of the column. For example, in a table of United States presidents, we might have the year 1732 in one cell. The label on the row may be "George Washington", and the label on the column "Birth date". This spatial arrangement then conveys the relation $birthdate(GW, 1732)$.

A beep in your car can mean several things—you haven't fastened your seatbelt, the door is open, your lights are still on. When a sequence of beeps is combined with a property of that sequence, namely, a rapid frequency, the composite meaning can be more precise—your lights are still on.

A map is an interesting example of a complex visual symbolic object. There are underlying fields, indicated by colors, that represent regions of various political or geologic types. Icons are overlaid on these fields in a way that bears at least a topological relation to the reality represented, and perhaps a geometric relation. Generally there is a mixture of topological and geometric correspondences; the width of a road on a map is usually not proportional to its width in reality. The icons represent entities of interest. The icons can have internal structure representing different categories of entities; for example, the size of a city may be represented by the number of circles around a center dot. Labels represent names. Information is conveyed by the possible spatial relations of adjacency, distance, and orientation. For example, labels naming cities are placed near the icon representing the city.

In a process diagram, as in maps, icons represent entities of different types. Individual states may be represented by a set of icons standing in particular spatial relationships to each other, representing the relations among the entities that characterize the state. Adjacent states may be connected by arrows, the direction of the arrow indicating temporal order and thus state transitions (Wahlster et al., 1993; Pineda and Garza, 2000).

Documents are composed of blocks of text and other symbolic elements in a particular spatial relationship. Some spatial relationships are tightly constrained. Titles need to be at the top of the document; the relation conveyed is that the title somehow indicates the content of the body of the document. Paragraphs meant to be read in sequence should be adjacent (with exceptions for page and column breaks). Paragraph breaks should also correlate with the coherence structure of the text, although often in a complex way. Other spatial relationships can be looser. A diagram, map, photograph, or sidebar only needs to be relatively near the block of text describing roughly the same thing, and the ordering of such figures should respect the order of references to them in the body of the text.

Web sites and PowerPoint presentations are complex symbolic objects amenable to similar analysis.

The performance of an individual in face-to-face conversation (Allwood, 2002) is also a complex symbolic object with its own compositional principles. The principal component symbolic elements are the speech stream, prosody, facial expressions, gaze direction, body posture, and gestures with the hands and arms. The primary mode of composition is temporal synchrony. The relationship conveyed by temporal synchrony could either be that the content of the two presentations are the same, as when someone makes a rolling gesture while describing Tweety rolling down the hill as in McNeil's experiments (McNeil, 2000), or parallel actions, as when someone freezes her arm in midgesture to hold her turn while beginning a new clause. Lascarides and Stone (2006) have shown that the possible relations among gestures and between gestures and speech are the same as the coherence relations that obtain between adjacent segments of discourse, namely, those based on causality, similarity and contrast, elaboration, interlocking change of state, and figure-ground.

A play is a more complex composite symbolic object, but its structure is equally amenable to a compositional account in terms of adjacency relations among the symbolic components and conjoined relations in the meaning.

In Section 6 we describe a program implementing these ideas to translate natural language into logical form. In Section 7 we develop more fully a theory of diagrams, to illustrate the application of this theory of information structure to a special case of substantial importance. But first some comments about coreference.

5 Coreference

In a complex symbolic object, different component presentations may convey the same concept, or more generally, closely related concepts. Thus there is a coreference relation between them. The most familiar example is between two noun phrases in speech or text. On a map, the dot with a circle around it representing a city and the label with the name of the city are in a sense coreferential, both representing the city. In a process diagram, the identity of shape or color of icons in two state depictions may indicate coreference relations; they are the same entity in different stages of the process. In a document a region of a photograph, a phrase in the caption on the photograph, and a portion of the accompanying text may all be coreferential. An iconic gesture and the accompanying description in a face-to-face conversation may also be coreferential.

It is clear that the coding of coreference relations in Web sites would be very useful for retrieval. For example, we would be able to move beyond the rather haphazard results we now get from image searches.

6 Compositional Semantics

Hobbs (1985a, 2003) defines a logical formalism for representing natural language in which the logical form of a sentence is a flat conjunction of atomic propositions in which all variables are existentially quantified with the widest possible scope. What in more standard notations are represented by scope are represented in this notation by predicate-argument relations, through the liberal use of reification of such things as possible eventualities and type elements of sets.

In such a notation, it is very straightforward to see how compositional semantics can be organized around the idea that concatenation within sentences means predicate-argument relations. Morphemes provide one or more predications, that is, an n-ary predicate applied to n arguments, where the arguments are existentially quantified variables. When two adjacent strings are recognized as part of the same constituent, we have recognized the identity of two of the variables. This effects a recognition of the predicate-argument relation. Conside a simple example.

Tall men succeed.

We will ignore the plural and present tense, although they are not difficult to handle. The word “tall” contributes the proposition $tall(x_1)$. The morpheme “man” contributes $man(x_2)$. The word “succeed” contributes $succeed(x_3)$.

$tall(x_1) \wedge man(x_2) \wedge succeed(x_3)$

When we recognize “tall men” as an NP, this tells us that $x_1 = x_2$. When we recognize “Tall men succeed” as a clause, this tells us that $x_2 = x_3$. With a flat logical form, the function application of lambda expressions that is the staple of most treatments of compositional semantics, from Montague (1974) on, in fact does no more than identify variables. Hobbs (2005) presents a sizeable portion of English syntax formalized along these lines.

Nishit Rathod (2008) developed a tool called LFToolkit for generating logical form from parse trees based on this idea. A very simple example of an LFToolkit rule would be something like the following:

```
<S,e-1,x-1,y-1> -> NP:<NP,x-1> VP:<VP,e-1,x-1,y-1>
```

This says that when you concatenate an NP and a VP to get a clause S, then the entity x-1 referred to by the NP is the same as the subject that the VP needs. Other rules output logical form fragments at the lexical level.

We have employed LFToolkit in a system for generating flat logical forms for natural language text, handling a substantial proportion of English syntax.

We exploit recent advances in statistical parsing by using as our input the parse trees produced by a statistical parser such as the Charniak parser (1997). However, these parsers are trained on the Penn TreeBank (Marcus, 2009), which lacks structure crucial for generating logical form. In particular, there is no structure in the left modifiers of the head noun in an NP. No complement-adjunct distinction is made. Gap-filler relations are unmarked in long-distance dependency constructions. The label SBAR covers questions, relative clauses, subordinate clauses, and “that” clauses, among others, and all of these are realized in different ways in logical form. Finally, there are thousands of patterns because the parse trees are not binary. For example, a transitive verb phrase can have any number of adverbials after the complement.

(9) `VP` → `Verb NP ADVP ADVP ADVP ADVP ADVP ...`

Hence, we would have to write an unlimited number of LFToolkit rules to accomodate all of these supposed constructions.

Consequently, we first preprocess the parse tree to put it into an amenable form that contains all the right information in the places where it is needed. We insert lexical information, e.g., that a verb is transitive, into the lexical nodes, using a very large lexicon that is based on the lexicon that was used in SRI's DIALOGIC project in the 1980s but was augmented from various other resources. Lexical information is passed up the tree to the nodes where it is needed, so that, for example, the VP knows its head verb is transitive. Structural information is passed down the tree to the nodes where it is needed, for example, about what kind of SBAR a clause or verb phrase is embedded in. In addition, the tree is binarized, to eliminate structures like (9).

At this point it is possible to capture a significant portion of English syntax with 200 to 300 LFToolkit rules.

This system has been used in demo systems and in computational linguistics classes on arbitrary English written texts in a number of domains.

When proposing a general theory of information structure, one huge challenge is to show that it can handle the complex meanings of our most elaborated symbol system, natural language. We have done just that by means of the implementation of a system for generating logical form from natural language text, based on the idea that concatenation in sentences conveys predicate-argument relations, or equivalently, the identity of locally posited, existentially quantified variables.

Another challenge for such a theory is whether the meanings of diagrams can be captured. This is the topic of the next section.

7 A Theory of Diagrams

7.1 Background Theories

In this section we develop a theory of diagrams, and apply it to defining Gantt charts. A Gantt chart is a particularly common type of diagram in certain corporate cultures; it graphs tasks and subtasks in a project against a time line. We need to rely on concepts from several background theories, not all of which have been described in published papers.

Set Theory: We need one relation and one function:

$member(x, s)$: x is a member of the set s.
$card(s) = n$: The cardinality of set s is n.

Composite Entities: A composite entity x is something that has a set of components and a set of relations among the components. We will need two relations:

componentOf(x, s): x is a component of s.
relationOf(r, s): r is a relation of s.

This depends on reifying relations (cf. Hobbs, 1985a).

Scales: One-dimensional diagram objects, intervals of time, and, by extension, events are all scales and have beginnings and ends. It will be convenient to have these concepts in both relational and functional form:

$\mathit{begins}(x, s)$:	x is the beginning of s.
$\mathit{ends}(x, s)$:	x is the end of s.
$\mathit{beginningOf}(s) = x$:	x is the beginning of s.
$\mathit{endOf}(s) = x$:	x is the end of s.

Strings: We assume there are strings of characters. They are usually symbolic objects, so they can be the first argument of the predicate *means*.

Time: In the development on Gantt charts, reference is made to concepts in OWL-Time (Hobbs and Pan, 2004). This ontology posits temporal entities (i.e., intervals and instants), the beginnings and ends of intervals, a *before* relation, Allen interval relations like *intMeets*, and temporal aggregates, which are sets of nonoverlapping, ordered temporal entities. It also handles durations and clock and calendar terms. The three predicates we need here are the following:

$\mathit{TimeLine}(t)$:	t is the infinite interval containing all temporal entities.
$\mathit{atTime}(e, t)$:	Event e occurs at instant t.
$\mathit{calInt}(t)$:	t is a calendar interval, i.e., a calendar day, week, month, or year.

In addition, we will need one predicate relating strings to times:

$\mathit{dateStringFor}(s, t)$:	s is a string describing temporal entity t.

The one predicate we need from a theory of causality or processes (Hobbs, 2005) is *enables*:

$\mathit{enables}(e_1, e_2)$:	Event or condition e_1 enables, or is a prerequisite for, event or condition e_2.

For Gantt charts we need a simple ontology of projects, with the following three predicates.

$\mathit{Project}(p)$:	p is a project.
$\mathit{taskIn}(z, p)$:	z is one of the tasks of project p.
$\mathit{milestoneIn}(m, p)$:	m is one of the milestones of project p.

A project is a composite entity among whose components are tasks and milestones, which are events. The project and its parts can have names.

$name(s, z)$: The string s is the name of z.

Space: The actual drawing of a diagram will involve mapping the ontology of diagrams to an ontology of space. Some portion of space will have to be chosen as the ground. This will define the vertical and horizontal directions and the *above* and *rightOf* relations. In addition, the articulation between the theory of diagrams and the theory of space would have to specify the kinds of spatial regions that realize different kinds of diagram objects.

7.2 Diagram Objects

A diagram consists of various diagram objects placed against a ground, where each diagram object has a meaning. We can take the ground to be a planar surface, which thus has points. Diagram objects can have labels placed near them, and generally they indicate something about the meaning. Diagram objects, points, frameworks, meanings, and labels are discussed in turn, and then it is shown how these can be used to define Gantt charts.

Diagram objects can be classified in terms of their dimensionality. In a spatial ontology in general, we would have to specify both a dimension of the object and the dimension of the embedding space, but in this theory of diagrams, we will take our embedding space to be a two-dimensional plane. Thus, there are three types of diagram objects:

$$0DObject(x) \supset DiagramObject(x)$$

$$1DObject(x) \supset DiagramObject(x)$$

$$2DObject(x) \supset DiagramObject(x)$$

Zero-dimensional diagram objects in diagrams are the class of diagram objects that are treated as having zero dimensions in the context of the diagram. Of course, in a spatial ontology they would actually be small regions generally with some symmetries. Three types of zero-dimensional diagram objects are dots, tickmarks, and diamonds.

$$Dot(x) \supset 0DObject(x)$$

$$Tickmark(x) \supset 0DObject(x)$$

$$Diamond(x) \supset 0DObject(x)$$

One-dimensional diagram objects in diagrams include curves.

$$Curve(x) \supset 1DObject$$

Three important kinds of curves are lines, rays (half-lines), and line segments.

$$Line(x) \supset Curve(x)$$

$$Ray(x) \supset Curve(x)$$

$$LineSegment(x) \supset Curve(x)$$

A line has no beginning or end. A ray has a unique beginning but no end. A line segment has both a unique beginning and a unique end.

$$Line(x) \supset [\neg(\exists\, p_1)[begins(p_1, x)] \wedge \neg(\exists\, p_2)[ends(p_2, x)]]$$

$$Ray(x) \supset [[(\exists\, !p_1)[begins(p_1, x)] \wedge \neg(\exists\, p_2)[ends(p_2, x)]]]$$

$$LineSegment(x) \supset (\exists\, !p_1, p_2)[begins(p_1, x)] \wedge ends(p_2, x)]$$

Beginnings and ends are points, in the sense described below.

Occasionally below it will be convenient to have functions for line segments corresponding to the *begins* and *ends* relations.

$$LineSegment(x) \supset (\forall\, p)[beginningOf(x) = p \equiv begins(p, x)]$$

$$LineSegment(x) \supset (\forall\, p)[endOf(x) = p \equiv ends(p, x)]$$

It will be useful to have a term *Linear* that covers all three types of linear diagram objects.

$$Linear(x) \equiv [Line(x) \vee Ray(x) \vee LineSegment(x)]$$

A line segment "in" a linear diagram object is one that is wholly contained in it.

$$\begin{aligned}&lineSegmentIn(x, y)\\&\quad \equiv LineSegment(x) \wedge Linear(y) \wedge (\forall\, p)[pointIn(p, x) \supset pointIn(p, y)]\end{aligned}$$

Another kind of curve is an arrow.

$$Arrow(x) \supset Curve(x)$$

An arrow has a specific unique beginning and end.

$$Arrow(x) \supset (\exists\, !p_1, p_2)[begins(p_1, x) \wedge ends(p_2, x)]$$

An arrow may be straight or curved. It may overlap in part with other arrows. An arrow is similar to a line segment, but is realized in the underlying spatial ontology differently, e.g., with a small triangle for its end.

It will be useful to talk about an arrow as a ternary relation.

$$arrow(x, p_1, p_2) \equiv [Arrow(x) \wedge begins(p_1, x) \wedge ends(p_2, x)]$$

Diagrams are composite entities whose components are diagram objects.

$$Diagram(d) \wedge componentOf(x, d) \supset DiagramObject(x)$$

7.3 Points and the *at* Relation

A ground consists of *points* and any diagram object consists of points, in some loose sense of "consist of"; that is, for any ground and any diagram object there is a corresponding set of points.

$$[Ground(x) \vee DiagramObject(x)] \supset (\exists\, s)(\forall\, p)[member(p, s) \supset pointIn(p, x)]$$

A zero-dimensional object has exactly one point in it.

$$0DObject(x) \supset (\exists\, !p)pointIn(p, x)$$

For convenience we will say that the single point in a zero-dimensional object both begins and ends it.

$$0DObject(x) \supset (\forall\, p)[pointIn(p, x) \equiv [begins(p, x) \wedge ends(p, x)]]$$

Points are not diagram objects.

The beginnings and ends of linear objects are points.

$$begins(p, x) \supset Linear(x) \wedge pointIn(p, x)$$
$$ends(p, x) \supset Linear(x) \wedge pointIn(p, x)$$

Points in the ground are partially ordered by an *above* relation and a $rightOf$ relation.

$$above(p_1, p_2, g) \supset Ground(g) \wedge pointIn(p_1, g) \wedge pointIn(p_2, g)$$
$$rightOf(p_1, p_2, g) \supset Ground(g) \wedge pointIn(p_1, g) \wedge pointIn(p_2, g)$$

A linear object is horizontal if no point in it is above any other. Similarly, vertical.

$$\begin{aligned} &horizontal(x, g) \equiv Linear(x) \\ &\quad \wedge \neg(\exists\, p_1, p_2)[pointIn(p_1, x) \wedge pointIn(p_2, x) \wedge above(p_1, p_2, g)] \end{aligned}$$
$$\begin{aligned} &vertical(x, g) \equiv Linear(x) \\ &\quad \wedge \neg(\exists\, p_1, p_2)[pointIn(p_1, x) \wedge pointIn(p_2, x) \wedge rightOf(p_1, p_2, g)] \end{aligned}$$

A horizontal ray all of whose points are to the right of its beginning is a rightward positive ray.

$$\begin{aligned} &[ray(x) \wedge horizontal(x, g) \wedge begins(p_0, x) \\ &\qquad \wedge (\forall\, p)[pointIn(p, x) \supset [p = p_0 \vee rightOf(p, p_0, g)]] \\ &\quad \supset rtPositive(x, g) \end{aligned}$$

A vertical ray all of whose points are above its beginning is a upwardly positive ray. A vertical ray all of whose points are below its beginning is a downwardly positive ray.

$$
\begin{array}{l}
[\mathit{ray}(x) \wedge \mathit{vertical}(x,g) \wedge \mathit{begins}(p_0,x) \\
\qquad \wedge (\forall p)[\mathit{pointIn}(p,x) \supset [p = p_0 \vee \mathit{above}(p,p_0,g)]] \\
\quad \supset \mathit{upPositive}(x,g)
\end{array}
$$

$$
\begin{array}{l}
[\mathit{ray}(x) \wedge \mathit{vertical}(x,g) \wedge \mathit{begins}(p_0,x) \\
\qquad \wedge (\forall p)[\mathit{pointIn}(p,x) \supset [p = p_0 \vee \mathit{above}(p_0,p,g)]] \\
\quad \supset \mathit{dnPositive}(x,g)
\end{array}
$$

$$\mathit{rtPositive}(x,g) \supset \mathit{ray}(x)$$

$$\mathit{upPositive}(x,g) \supset \mathit{ray}(x)$$

$$\mathit{dnPositive}(x,g) \supset \mathit{ray}(x)$$

A special kind of line segment needed for Gantt charts is a horizontal bar.

$$\mathit{HBar}(x) \supset (\exists g)[\mathit{horizontal}(x,g) \wedge \mathit{LineSegment}(x)]$$

When realized spatially, it will generally be thicker than other line segments.

Diagrams are constructed by placing points in diagram objects at points in the ground or in another diagram object. The *at* relation expresses this.

$$
\begin{array}{l}
\mathit{at}(p_1,p_2) \\
\quad \supset (\exists x_1,x_2)[\mathit{pointIn}(p_1,x_1) \wedge \mathit{pointIn}(p_2,x_2) \wedge \mathit{DiagramObject}(x_1) \\
\qquad \wedge [\mathit{Ground}(x_2) \vee \mathit{DiagramObject}(x_2)] \wedge x_1 \neq x_2]
\end{array}
$$

The relation *at* can be extended to zero-dimensional objects in the obvious way.

$$0\mathit{DObject}(x) \supset (\forall p)[\mathit{at}(x,p) \equiv (\exists p_1)[\mathit{pointIn}(p_1,x) \wedge \mathit{at}(p_1,p)]]$$

Typically, frameworks (see below) will be placed with respect to some points in the ground, and other diagram objects will be placed with respect to the framework or other diagram objects.

The relations of a diagram as a composite entity include its *at* relations. To say this formally we can reify the *at* relation. Thus, $\mathit{at}'(r,p_1,p_2)$ means that r is the *at* relation between p_1 and p_2. We can then say that r is a member of the relations of the diagram.

$$\mathit{at}'(r,p_1,p_2) \wedge \mathit{relationOf}(r,d)$$

7.4 Frameworks

Many diagrams have an underlying framework with respect to which diagram objects are then located, e.g., the lat-long framework on maps. A framework is a set of objects in a particular relationship to each other.

$$
\begin{array}{l}
\mathit{Framework}(f) \\
\quad \supset (\exists s)(\forall x)[\mathit{member}(x,s) \supset \mathit{DiagramObject}(x) \wedge \mathit{componentOf}(x,f)]
\end{array}
$$

One very important kind of framework is a coordinate system. Here I will characterize only a rectilinear cooordinate system.

$$RCoordinateSystem(f) \supset Framework(f)$$

$$RCoordinateSystem(f) \supset (\exists\, g)[Ground(g) \wedge groundFor(g,f)]$$

$$\begin{aligned}&RCoordinateSystem(f) \wedge groundFor(g,f)\\ &\quad\supset (\exists\, x,y)[xAxisOf(x,f) \wedge yAxisOf(y,f) \wedge rtPositive(x,g)\\ &\qquad \wedge [upPositive(y,g) \vee dnPositive(y,g)]]\end{aligned}$$

$$xAxisOf(x,f) \supset RCoordinateSystem(f) \wedge componentOf(x,f)$$

$$yAxisOf(x,f) \supset RCoordinateSystem(f) \wedge componentOf(y,f)$$

An x-axis and a y-axis are both axes.

$$xAxisOf(x,f) \supset axis(x)$$

$$yAxisOf(y,f) \supset axis(y)$$

Two points have the same x-coordinate if there is a vertical line that contains both of them. Similarly, same y-coordinate.

$$\begin{aligned}&sameX(p_1,p_2,f)\\ &\quad\equiv (\exists\, l,g)[groundFor(g,f) \wedge vertical(l,g) \wedge pointIn(p_1,l) \wedge pointIn(p_2,l)]\end{aligned}$$

$$\begin{aligned}&sameY(p_1,p_2,f)\\ &\quad\equiv (\exists\, l,g)[groundFor(g,f) \wedge horizontal(l,g) \wedge pointIn(p_1,l) \wedge pointIn(p_2,l)]\end{aligned}$$

The x-value of a point p is a point p_1 in the x axis with the same x-coordinate. Similarly for the y-value.

$$x\text{-}value(p_1,p,f) \equiv (\exists\, x)[sameX(p_1,p,f) \wedge pointIn(p_1,x) \wedge xAxisOf(x,f)]$$

$$y\text{-}value(p_2,p,f) \equiv (\exists\, y)[sameY(p_2,p,f) \wedge pointIn(p_2,y) \wedge yAxisOf(y,f)]$$

It will be convenient below to talk about the y-value of a horizontal line segment.

$$\begin{aligned}&LineSegment(h) \wedge horizontal(h,g) \wedge groundFor(g,f)\\ &\quad\supset (\forall\, p_2)[y\text{-}value(p_2,h,f) \equiv (\exists\, p)[pointIn(p,h) \wedge y\text{-}value(p_2,p,f)]]\end{aligned}$$

7.5 Meanings

Associated with every object in a diagram is its meaning. Meaning for diagrams is thus a function mapping diagram objects into entities provided by some other ontology. Meaning is conveyed by the predication $means(x,c)$ introduced above, where x is a diagram object. There are no constraints on the second argument of *means*; it just has to be an entity in some ontology.

$$DiagramObject(x) \supset (\exists\, c)means(x,c)$$

The meanings of the *at* relations in a diagram will be specified by means of axioms having the following form:

$$at(x,y) \wedge p(x) \wedge q(y) \wedge means(x,a) \wedge means(y,b) \supset r(a,b)$$

That is, if a p-type diagram object x is at a q-type diagram object y in a diagram, then if x means a and y means b, then there is an r relation between a and b.

Axes in a coordinate system generally mean some set in another ontology. That set may be unordered (a set of tasks), discrete and linearly ordered (months), or continuous (time).

7.6 Labels

A label is a textual object that can be associated with objects in a diagram. The two basic facts about labels cannot be defined with precision without making reference to the cognition of the reader of the diagram.

1. A label is placed near the object it labels, in a way that allows the reader of the diagram to uniquely identify that object.
2. The content of the label as a string bears some relation to the meaning of the object that it labels, in that perceiving the string causes one to think of the meaning.

Specifying the first of these completely is a very hard technical problem (Edmondson et al., 1997). For example, often on a map one cannot correctly associate the name of a town with a dot on the map without doing the same for all nearby towns, and associating a curve on a map with the name of a road often requires abductive inferences about shortest paths and consistency of line thickness. Here we will simply say that a label can be placed *at* an object, and leave it to component-specific computation to determine what *at* means in some context.

$$label(l,x) \supset string(l) \wedge DiagramObject(x) \wedge at(l,x)$$

The second property of labels is also a difficult technical, or even artistic, problem. But a very common subcase is where the label is a name. The whole purpose of a name is to cause one to think of the object when one perceives the name, so it serves well for this property of labels.

$$label(l,x) \supset (\exists\, c)[means(l,c) \wedge means(x,c)]$$

7.7 Gantt Charts

A Gantt chart g for a project p is a diagram that consists of several types of components. It has a rectilinear coordinate system f where the x-axis is rightward positive and the y-axis is upward or downward positive. (The x-axis can appear at the top or the bottom of the chart.) The meaning of the x-axis is the time line or some other periodic temporal aggregate, and the meaning of the y-axis is a set of tasks in the project.

$$
\begin{aligned}
&\mathit{GanttChart}(g,p)\\
&\quad \supset\ \mathit{Diagram}(p)\ \wedge\ \mathit{Project}(p)\\
&\qquad \wedge (\exists\, f,x,y,t,s)[\mathit{RCoordinateSystem}(f) \wedge \mathit{componentOf}(f,g) \wedge \mathit{xAxisOf}(x,f)\\
&\qquad\quad \wedge \mathit{rtPositive}(x)\ \wedge\ \mathit{means}(x,t)\ \wedge\ \mathit{TimeLine}(t) \wedge \mathit{yAxisOf}(y,f)\\
&\qquad\quad \wedge [\mathit{upPositive}(y)\ \vee\ \mathit{dnPositive}(y)]\ \wedge\ \mathit{means}(y,s)\\
&\qquad\quad \wedge (\forall\, z)[\mathit{member}(z,s)\ \supset\ \mathit{taskIn}(z,p)]]
\end{aligned}
$$

A Gantt chart has horizontal bars representing the interval during which a task is executed.

$$
\begin{aligned}
&\mathit{GanttChart}(g,p)\ \wedge\ \mathit{RCoordinateSystem}(f) \wedge \mathit{componentOf}(f,g)\\
&\quad \supset\ (\exists\, s)(\forall\, b)[\mathit{member}(b,s)\ \supset \mathit{componentOf}(b,g)\ \wedge\ \mathit{HBar}(b)\\
&\qquad \wedge (\exists\, r_1,z,p_1,t_1,q_2,t_2)[\textit{y-value}(r_1,b,f)\ \wedge\ \mathit{means}(r_1,z)\ \wedge\ \mathit{taskIn}(z,p)\\
&\qquad\quad \wedge \textit{x-value}(p_1,\mathit{beginningOf}(b),f)\ \wedge\ \mathit{means}(p_1,t_1)\ \wedge\ \mathit{begins}(t_1,z)\\
&\qquad\quad \wedge \textit{x-value}(q_1,\mathit{endOf}(b),f)\ \wedge\ \mathit{means}(q_1,t_2)\ \wedge\ \mathit{ends}(t_2,z)]]
\end{aligned}
$$

Because a task is an event, OWL-Time allows instants as the beginnings and ends of tasks. This axiom says that a Gantt chart has a set of components which are horizontal bars representing tasks and the beginning of the bar represents the starting time of the task and the end of the bar represents the finishing time of the task.

Similarly, a Gantt chart has diamonds representing milestones.

$$
\begin{aligned}
&\mathit{GanttChart}(g,p)\ \wedge\ \mathit{RCoordinateSystem}(f) \wedge \mathit{componentOf}(f,g)\\
&\quad \supset\ (\exists\, s)(\forall\, d)[\mathit{member}(d,s)\ \supset \mathit{componentOf}(d,g)\ \wedge\ \mathit{Diamond}(d)\\
&\qquad \wedge (\exists\, m,r_1,r_2,t_1)[\textit{y-value}(r_2,d,f)\ \wedge\ \mathit{means}(r_2,m)\ \wedge\ \mathit{milestone}(m,p)\\
&\qquad\quad \wedge \textit{x-value}(r_1,d,f)\ \wedge\ \mathit{means}(r_1,t_1)\ \wedge\ \mathit{atTime}(m,t_1)]]
\end{aligned}
$$

We can call bars and diamonds “task icons”.

$$
\mathit{taskIcon}(x)\ \equiv\ [\mathit{HBar}(x)\ \vee\ \mathit{Diamond}(x)]
$$

A Gantt chart often has arrows going from the end of one bar to the beginning of another, indicating the the first bar’s task is a prerequisite for the second bar’s task. A diamond can also be the source and/or target of an arrow.

$$
\begin{aligned}
&\mathit{GanttChart}(g,p)\ \wedge\ \mathit{RCoordinateSystem}(f) \wedge \mathit{componentOf}(f,g)\\
&\quad \supset\ (\exists\, s)(\forall\, a)[\mathit{member}(a,s)\ \supset \mathit{componentOf}(a,g)\\
&\qquad \wedge (\exists\, x,z_1,p_1,y,z_2,p_2)[\mathit{arrow}(a,p_1,p_2) \wedge \mathit{taskIcon}(x) \wedge \mathit{componentOf}(x,g)\\
&\qquad\quad \wedge \mathit{means}(x,z_1)\ \wedge\ \mathit{ends}(p_1,x)\ \wedge\ \mathit{taskIcon}(y) \wedge \mathit{componentOf}(y,g)\\
&\qquad\quad \wedge \mathit{means}(y,z_2)\ \wedge\ \mathit{begins}(p_2,y)\ \wedge\ \mathit{enables}(z_1,z_2)]]
\end{aligned}
$$

A bar in a Gantt chart may have labels for the date at its beginning and end.

$$
\begin{aligned}
&\mathit{GanttChart}(g,p)\ \wedge\ \mathit{HBar}(b) \wedge \mathit{ComponentOf}(b,g)\\
&\quad \supset\ [(\exists\, s_1)(\forall\, l_1)[\mathit{member}(l_1,s_1)\ \equiv\ (\exists\, p_1,q_1,t_1)[\mathit{begins}(p_1,b)\ \wedge\ \mathit{label}(l_1,p_1)\\
&\qquad\quad \wedge \textit{x-value}(q_1,p_1)\ \wedge\ \mathit{means}(q_1,t_1)\ \wedge\ \mathit{dateStringFor}(l_1,t_1)]\\
&\qquad \wedge \mathit{card}(s_1) < 2]
\end{aligned}
$$

$$\begin{aligned}&\wedge (\exists\, s_2)(\forall\, l_2)[member(l_2, s_2) \;\equiv\; (\exists\, p_2, q_2, t_2)[ends(p_2, b) \;\wedge\; label(l_2, p_2)\\&\quad \wedge\, x\text{-}value(q_2, p_2) \;\wedge\; means(q_2, t_2) \;\wedge\; dateStringFor(l_2, t_2)]\\&\wedge\, card(s_1) < 2]]\end{aligned}$$

The cardinality statement is a way of saying there is either zero or one label.

Similarly, a diamond in a Gantt chart may have a label for a date.

$$\begin{aligned}&GanttChart(g, p) \;\wedge\; Diamond(d) \wedge componentOf(d, g)\\&\quad\supset\; (\exists\, s)[(\forall\, l)[member(l, s) \;\equiv\; (\exists\, q, t)[label(l, d) \wedge x\text{-}value(q, d) \wedge means(q, t)\\&\qquad \wedge\, dateStringFor(l, t)]]\\&\qquad \wedge\, card(s) < 2]\end{aligned}$$

Points on the y-axis of a Gantt chart can be labelled with task names.

$$\begin{aligned}&[GanttChart(g, p) \wedge RCoordinateSystem(f) \wedge componentOf(f, g) \wedge yAxisOf(y, f)\\&\qquad \wedge\, pointIn(p_1, y)]\\&\quad\supset\; (\exists\, s)[(\forall\, l)[member(l, s) \;\equiv\; (\exists\, z)[means(p_1, z) \;\wedge\; name(l, z)]]\\&\qquad \wedge\, card(s) < 2]\end{aligned}$$

Points in the x-axis of a Gantt chart can be labelled with dates or times.

$$\begin{aligned}&[GanttChart(g, p) \wedge RCoordinateSystem(f) \wedge componentOf(f, g) \wedge xAxisOf(x, f)\\&\qquad \wedge\, pointIn(p_1, x)]\\&\quad\supset\; (\exists\, s)[(\forall\, l)[member(l, s) \;\equiv\; (\exists\, t)[label(l, p_1) \;\wedge\; means(p_1, t)\\&\qquad \wedge\, dateStringFor(l, t)]]\\&\qquad \wedge\, card(s) < 2]\end{aligned}$$

Line segments in the x-axis of a Gantt chart can be labelled with calendar intervals.

$$\begin{aligned}&[GanttChart(g, p) \wedge RCoordinateSystem(f) \wedge componentOf(f, g) \wedge xAxisOf(x, f)\\&\qquad \wedge\, lineSegmentIn(s_1, x)]\\&\quad\supset\; (\exists\, s)[(\forall\, l)[member(l, s) \;\equiv\; (\exists\, t)[means(s_1, t) \;\wedge\; label(l, s_1)\\&\qquad \wedge\, calInt(t) \;\wedge\; dateStringFor(l, t)]]\\&\qquad \wedge\, card(s) < 2]\end{aligned}$$

Further elaborations are possible. The labels can have internal structure. For example, labels for subtasks may be indented. Labels for time intervals may be broken into a line for months, a line below for weeks, and so on.

8 Modalities, Media, and Manifestations

In order for communication to work, perception of the symbol must occur. Humans are able to perceive optical, acoustic, and chemical phenomena, as well as pressure and temperature. Of these modalities the optical and acoustic are by far the most important, because they offer the richest possibilities for composition. Artifact agents of course have other modalities.

Communication requires one or more devices. There must be a manner in which the presentation is carried out. Allwood (2002) categorizes these into primary, secondary, and tertiary. The primary devices or media are the ones that are human body parts and processes. The voice is used for grunts, speech and song. The hands, arms, body, face and head are used for gesture. Even at this level some encoding must be done; we need to find words for the concepts we wish to convey, and these must be mapped into sequences of articulatory gestures.

The secondary media are those involving devices external to the human body, such as marks on paper as in writing and drawings, computer terminals, telephones, videotape, and so on. These typically require multiple encodings, where the final code is known to the intended audience. The various media have different advantages and disadvantages that can be exploited for different kinds of represented content. For example, visual spatial representations can exploit more dimensions for conveying relationships than can auditory temporal representations. Hovy and Arens (1990) catalog many of these correlations.

Allwood also mentions tertiary media, including paintings, sculptures, and aesthetic designs of artifacts such as chairs. These are probably just secondary media where the content that is represented is much more difficult to capture in words.

We have a strong tendency to group together classes of symbolic entities that share the same property, especially their content, and think of the aggregates as individuals in their own right. It is probably better in an ontology of symbolic entities to view these as first-class individuals that themselves represent a particular content. Other symbolic entities may be manifestations of these individuals. The predicate *manifest* is a transitive relation whose principal property is that it preserves content.

$$manifest(x_1, x) \wedge means(x, c, s) \supset means(x_1, c, s)$$

(This does not take into account translations, where the s's differ.)

Thus, to use the example of Pease and Niles (2001), there is an entity called *Hamlet*. The performance of *Hamlet* manifests *Hamlet*. The performance of *Hamlet* in a particular season by a particular company manifests that, and a performance on a particular night may manifest that. A videotape of that particular performance manifests the performance, and every copy of that videotape manifests the videotape. A similar story can be told about the script of the play, a particular edition of the script, and a particular physical book with that edition of the script as its content.

The above rule should be thought of as defeasible, because variations exist, lines can be dropped, and printer's errors occur. More precisely, if some proposition occurs in the content of one symbolic entity then defeasibly it occurs in the content of symbolic entities that manifest it.

9 Summary

Information structure is one of the most basic domains in an ontology of the everyday world, along with such domains as space and time. It should be anchored in an ontology of commonsense psychology, as we have tried to sketch here, and there should be an account of how complex symbolic entities can be composed out of simpler symbolic entities in various modalities and combinations of modalities. We have demonstrated the utility of this framework with respect to two of the most complex symbolic systems, natural language and diagrams. Because Web sites are an especially complex sort of symbolic entity, we can expect this kind of theory to be very significant in the development of the modes of access to Web resources.

Acknowledgements

We have profited from discussions with and work by Eduard Hovy, Nishit Rathod, Midhun Ittychariah, and Paul Martin. The opinions expressed here, however, are entirely our own. This material is based in part upon work supported by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under Contract No. NBCHD030010. It was also supported in part by DARPA's "Learning by Reading: The Möbius Project" (NBCHD030010 TO #0007), and in part under the IARPA (DTO) AQUAINT program, contract N61339-06-C-0160.

References

[1] Allwood, Jens. 2002. "Bodily Communication—Dimensions of Expression and Content", in B. Grandström, D. House, and I. Karlsson (Eds.), *Multimodality in Language and Speech Systems*, Kluwer Academic Publishers. Dordrecht, The Netherlands.

[2] Bickerton, Derek, 1990. *Language and Species*, University of Chicago Press.

[3] Charniak, Eugene, 1997. "Statistical Parsing with a Context-Free Grammar and Word Statistics", *Proceedings, AAAI-97, Fourteenth National Conference on Artificial Intelligence*, Providence, Rhode Island, pp. 598-603.

[4] Edmondson, Shawn, Jon Christensen, Joe Marks, and Stuart M. Shieber. (1997) "A General Cartographic Labeling Algorithm", *Cartographica*, Vol. 33, No. 4, pp. 13-23, Winter 1997.

[5] Gordon, Andrew, and Jerry R. Hobbs, 2004. "Formalizations of Commonsense Psychology", *AI Magazine*, Vol. 25, pp. 49-62.

[6] Grice, Paul, 1989. "Meaning", in *Studies in the Way of Words*, Harvard University Press, Cambridge, Massachusetts, 1989.

[7] Hobbs, Jerry R. 1985a. “Ontological Promiscuity.” *Proceedings, 23rd Annual Meeting of the Association for Computational Linguistics*, pp. 61-69. Chicago, Illinois, July 1985.

[8] Hobbs, Jerry R., 1985b. “On the Coherence and Structure of Discourse”, Report No. CSLI-85-37, Center for the Study of Language and Information, Stanford University.

[9] Hobbs, Jerry R. 2003. “The Logical Notation: Ontological Promiscuity”, Available at `http://www.isi.edu/ hobbs/disinf-tc.html`

[10] Hobbs, Jerry R., 2005. “The Syntax of English in an Abductive Framework”, available at `http://www.isi.edu/ hobbs/disinf-chap4.pdf`

[11] Hobbs, Jerry R., 2005. “Toward a Useful Concept of Causality for Lexical Semantics”, *Journal of Semantics*, Vol. 22, pp. 181-209.

[12] Hobbs, Jerry R., 2006. “The Origin and Evolution of Language: A Plausible, Strong AI Account”, in Michael A. Arbib, editor, *Action to Language via the Mirror Neuron System*, Cambridge University Press, Cambridge, England, pp. 48-88.

[13] Hobbs, Jerry R., William Croft, Todd Davies, Douglas Edwards, and Kenneth Laws, 1987. “Commonsense Metaphysics and Lexical Semantics”, *Computational Linguistics*, vol. 13, nos. 3-4, July-December 1987, pp. 241-250.

[14] Hobbs, Jerry R. and Feng Pan, 2004. “An Ontology of Time for the Semantic Web”, *ACM Transactions on Asian Language Information Processing*, Vol. 3, No. 1, March 2004, pp. 66-85.

[15] Hobbs, Jerry R., Mark Stickel, Douglas Appelt, and Paul Martin, 1993. “Interpretation as Abduction”, *Artificial Intelligence*, Vol. 63, Nos. 1-2, pp. 69-142.

[16] Hovy, Eduard H. and Yigal Arens. 1990. “When is a Picture Worth a Thousand Words?—Allocation of Modalities in Multimedia Communication”, AAAI Spring Symposium on Human-Computer Interactions. Palo Alto, CA.

[17] Lascarides, Alex, and Matthew Stone, 2006. “Formal Semantics of Iconic Gesture”. in David Schlangen and Raquel Fernandez, eds., *Proceedings*, BRANDIAL 06: 10th Workshop on the Semantics and Pragmatics of Dialogue, pp. 64-71.

[18] Marcus, Mitchell, 2009. *The Penn TreeBank Project*, available at `http://www.cis.upenn.edu/ treebank/`

[19] McCarthy, John, 1980. “Circumscription: A Form of Nonmonotonic Reasoning”, *Artificial Intelligence*, Vol. 13, pp. 27-39. Reprinted in M. Ginsberg, ed., *Readings in Nonmonotonic Reasoning*, pp. 145-152, Morgan Kaufmann Publishers, Inc., Los Altos, California.

[20] McNeil, David, ed. 2000. *Language and Gesture*, Cambridge University Press, Cambridge, England.

[21] Montague, Richard, 1974. "The Proper Treatment of Quantification in Ordinary English", in R. H. Thomason, ed., *Formal Philosophy: Selected Papers of Richard Montague*, pp. 247-270, Yale University Press, New Haven, Connecticut.

[22] Pease, Adam, and Ian Niles. 2001. "Practical Semiotics: A Formal Theory", available at `citeseer.nj.nec.com/531589.html`.

[23] Pineda, Luis, and Gabriela Garza, 2000. "A Model for Multimodal Reference Resolution", *Computational Linguistics*, Vol. 26, No. 2, June 2000, pp. 139-194.

[24] Pollard, Carl, and Ivan A. Sag, 1994. *Head-Driven Phrase Structure Grammar*, University of Chicago Press and CSLI Publications.

[25] Rathod, Nishit, 2008. "LFToolkit", available at `http://www.isi.edu/ hobbs/LFToolkit/index.html`.

[26] Wahlster, Wolfgang, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, and Thomas Rist, 1993. "Plan-based Integration of Natural Language and Graphics Generation", *Artificial Intelligence*, Vol. 63, Nos. 1-2, pp. 387-427.