The TRIPS Logical Form

The logical form language is an encoding of the semantic content of a sentence or text that can be mapped to a traditional knowledge representation only after contextual interpretation. It is roughly equivalent to a modal logic with underspecified scoping relationships. In addition, is if a "flat" representation, without the nested expressions one would see in a logic. The connection between the expressions is captured by the logical form variables, which serve as the links between different aspects of the formula.

In a typical application, the LF might pass through reference resolution to identify the intended referents of referring expressions, undergo some scope disambiguation to identify the intended order or quantifiers and operators. See Allen (1995) for an early discussion of our approach to logical form, and Manshadi et al (2008) for an exploration of the LF as a underspecified constraint-based representation.

Another consideration is the support of robust parsing and interpretation. The LF is designed so that the correct representation of fragments extracted from an utterance will be identical in form to the same phrases if we had produced a full parse. The key technique that enables this is the use of a "flat" unscoped representations.

The word senses and semantic relations used in the logical form are specified by the TRIPS ontology. We have [a browser for the LF ontology here].

The logical form of a sentence consists of a set of terms describing objects and relationships evoked by the utterance. One key term is speech act that was performed. For example, the logical form of

But the man wants to eat it

is as follows:

(ONT::SPEECHACT V12 ONT::SA_TELL :CONTENT w1 :MODS (b1))	s1 is a TELL speech act with content w1 and (discourse) modifier b1
(ONT::F w1 (:* ONT::WANT W::WANT) :ACTION e1 :EXPERIENCER m1 :TMA ((W::TENSE W::PRES)))	w1 is a wanting relation between m1 and e1, that holds at a time indicated by the present tense
(ONT::THE m1 (:* ONT::MALE W::MAN))	m1 is some man identifiable in context
(ONT::PRO i1 (:* ONT::REFERENTIAL-SEM W::IT) :Proform W::IT)	i1 is a some object identifiable in context by pronoun "it"
(ONT::F e1 (:* ONT::CONSUME W::EAT) :THEME i1 :AGENT m1))	e1 is an eating relation between m1 and p1
(ONT::F V176949 (:* ONT::CONJUNCT W::BUT) :ONT::OF w1))	w1 is related by a "but" relationship to previous context

Figure 1: The content of *But the man wants to eat it*

This document describes this output representation in detail.

1. Capturing Semantic Content

Building Blocks and Terms

The basic atoms of the logical form language consist of the following, each which will be described in more detail as we go along:

ATOMIC TYPES, atoms that denote classes of objects, be they physical objects, situations, abstractions, and so on (e.g., ONT::PERSON, ONT::SEND, ONT::NUMBER). These are organized into a hierarchy in the LF Ontology.

ROLE NAMES, which can be thought of as "slots" in a frame, labeled arguments to a predicate, or functions from one object to another.

VARIABLES, written in lower case, x, y, c

LOGICAL OPERATORS, operators, such as ONT::AND and ONT::OR

TERM CONSTRUCTORS: ONT::THE, ONT::A, ONT::PRO, ONT::IMPRO, etc. as described below.

Terms in the LF all have the exact same format, namely

(<term constructor> <var-name> <type> [<role> <value>])*

where <*value*> can be

A variable

A list of variables

A list of pairs of form (<role> <value>), which is currently used for the value of the TMA slot as described below.

2. Speech Acts

Utterances are represented at the top level by a surface speech act captures the literal or surface speech act of the utterance1. Using the example above with *But the man wants to eat it*, the top level form is defined as a surface speech act, which in this case is ONT::SA_TELL, the speech act that corresponds to most declarative sentences, i.e.,

(ONT::SPEECHACT V12 ONT::SA_TELL :CONTENT w1 :MODS (b1))

The content w1 is defined by other terms. Discourse connectives such as but are treated as modifiers on the speech act. In this case, the term b1 defines the modifier as is defined as shown in Figure 1.

Surface Acts

The surface speech acts are listed below, together with the slots that can occur. Most of these acts allow :CONTENT and :MODS slots. In few cases, other slots are possible. The associated roles (except for :MODS which can occur with any term) are listed for each act.

---

[1] Even when subsequently interpreted indirectly, the surface act influences the allowable forms of appropriate responses. For example, in response to the invitation *can you come to my party,* one can accept by saying *yes, i can*. But this acceptance would be inappropriate in response to the invitations *Please come to my party* or *Let me invite you to my party*. Each of these invitations would have a different surface act form. The fact that they are invitations is inferred by contextual interpretation.

<table>
<tr><td>ONT::SA_Wh-Question :content :focus</td><td>“Wh” questions</td><td colspan="2">Where is the knife? When? What?</td></tr>
<tr><td>ONT::SA_YN-Question :content</td><td>Yes no question</td><td colspan="2">Is the knife in the kitchen?</td></tr>
<tr><td>ONT::SA_Request :content</td><td>Imperative, typically a command</td><td colspan="2">Get the knife.</td></tr>
<tr><td rowspan="6">ONT::SA_Response :content</td><td rowspan="6">Responses to yes-no questions</td><td>Content</td><td>Example</td></tr>
<tr><td>POS</td><td>Yes</td></tr>
<tr><td>NEG</td><td>No</td></tr>
<tr><td>UNSURE-POS</td><td>Maybe</td></tr>
<tr><td>UNSURE-NEG</td><td>I don’t think so</td></tr>
<tr><td>UNSURE</td><td>I don’t know</td></tr>
<tr><td>ONT::SA_Tell :content</td><td>Assertions</td><td colspan="2">The knife is in the kitchen</td></tr>
<tr><td>ONT::SA_Evaluate :content</td><td>Acts that express an opinion about something</td><td colspan="2">Good, bad, excellent, OK, so-so, …</td></tr>
<tr><td>ONT::SA_Ack :content</td><td>Acts that acknowledge or confirm</td><td colspan="2">OK</td></tr>
<tr><td>ONT::SA_Request-comment :content</td><td>Acts that suggest an object or action</td><td colspan="2">How about coming to my party. What about a beer. How about Toronto.</td></tr>
<tr><td>ONT::SA_Greet :content</td><td>Greetings and Goodbyes</td><td colspan="2">Hello, Hi, Bye,</td></tr>
<tr><td>ONT::SA_Thank</td><td>Thanks</td><td colspan="2">Thanks, thank-you,</td></tr>
<tr><td>ONT::SA_Welcome</td><td>Responses to thanks</td><td colspan="2">You’re welcome, not at all</td></tr>
<tr><td>ONT::SA_Discourse-Manage</td><td>Acts that help manage the conversation, grounding, etc</td><td colspan="2">Just a second, uh-huh,</td></tr>
</table>

Figure 2: The Surface Speech Act Types

Discourse Adverbials

As mentioned above, the discourse adverbials relate the current utterance to the discourse context. Currently the parser does not analyze these into some deeper representation. Rather it just passes on a general classification (e.g., ONT::CONJUNCT) and the actual lexical forms for use in contextual interpretation. These connectors can affect many aspects of the analysis, including not only tense, but also the discourse act performed and what collaborative problem solving act is inferred. For example, the discourse adverbial “And” as in “Then it left” is treated as a modifier of type (:* ONT::CONJUNCT ONT::SATHEN) an the full LF for the speech act is

(ONT::SPEECHACT V7263 ONT::SA_TELL :CONTENT V7035 :MODS (V6997))
(ONT::F V7008 (: ONT::CONJUNCT W::THEN) :OF V7263)*

Some of the general classes are shown in Figure 3.

ONT::CONJUNCT	And, and-then, so, but, …
ONT::SEQUENCE-POSITION	first, second, next, last, …
ONT::TOPIC-SIGNAL	by the way, anyways, …
ONT::POLITENESS	Please
ONT::DEGREE-OF-BELIEF	Hopefully, Actually, in fact, …
ONT::INTERJECTION	You know, I guess, …
ONT::QUALIFICATION	Probably, originally, eventually, …

ONT::ADDITIVE	Too, also, …
ONT::REASON	So that, because, since,
ONT::QUALIFICATION	Maybe, …

Figure 3: Some discourse adverbials

3. A Quick Overview of Basic Phrases

Simple Descriptions

Simple descriptions involve term constructors corresponding to definite and indefinite forms. For example:

the train -- (ONT::THE x (:* ONT::VEHICLE W::TRAIN))

the trains -- (ONT::THE-SET x (:* ONT::VEHICLE W::TRAIN))

a train -- (ONT::A x (:* ONT::VEHICLE W::TRAIN))

trains -- (ONT::INDEF-SET x (:* ONT::VEHICLE W::TRAIN))

Simple Events

The meaning of clauses is modeled as a relation between objects that are the arguments to the verb, and indicated by expressions using the formula constructor F. This is a neutral term that is used for any clause, whether it be main clauses, subordinates, complements, and so on. The use of the constructor F is determined by syntax rather than the semantic type of the event. The fact that a relation is claimed to represent the world would be captured in the speech act - the semantic formula describes the content of the sentence but makes no claim about the world. The event described in the sentence *A man loaded the cargo* would be:

(ONT::F l1 (:* ONT::FILL-CONTAINER W::LOAD) :AGENT m1 :THEME c1)
(ONT::A m1 (:* ONT::MALE W::MAN))
(ONT::THE C1 (:* ONT::COMMODITY W::CARGO))

Note that events can occur in referring expressions as well. For instance, consider the following NP.

The loading of the cargo

(ONT::THE l1 (:* ONT::FILL-CONTAINER W::LOAD) :THEME c1)
(ONT::THE C1 (:* ONT::COMMODITY W::CARGO))

Modifiers

The LF of adjectives such as "red" are not treated as role predicates because they are time varying. A block might be red today, and green tomorrow. Furthermore, many adjectives take additional arguments and modifiers (as in "eager as a beaver", "ready to load"). Such propositions will use a named argument representation as we do with verbs. To attach a modifying phrase to a description, we use the :MODS role that takes a list of relation objects. For example:

The red truck

(ONT::THE V53 (:* ONT::LAND-VEHICLE W::TRUCK) :MODS (V57))
(ONT::F V57 (:* ONT::RED W::RED) :OF V53))

Typically, the single argument in unary relations is identified as the role :OF, as with the LF for red above. For binary functional relations, the arguments are typically identified as :OF and :VAL. For example, spatial prepositions use such arguments, as seen in the example:

ONT::THE	a definite singular form (we expect to be able to resolve it from context)
ONT::THE-SET	Definite plural form (we expect to identify a set of objects from context)
ONT::A	an indefinite form (we expect it to be introducing new object into context)
ONT::INDEF-SET	Indefinite plural (we expect to introduce a set of objects into the context)
ONT::SM	Indefinite mass-term, loosely meaning "some quantity of"
ONT::PRO	a pronoun form (we expect it to be resolved in local context
ONT::PRO-SET	a plural pronoun (we expect it to be resolved in local context)
ONT::IMPRO	an implicit anaphoric form (i.e., it is implicit in the text but does not appear; we expect it to be resolved from local context
ONT::BARE	NPs that have no specifier and are typically ambiguous between generic, kind, mass, and indefinite interpretations
ONt::QUANTIFIER	Universally quantified constructions (e.g., each truck, every item)
WH-TERM	"wh" terms as in questions (e.g., which trucks), and complements to verbs like *know* (e.g., *I know where they hid)*

Figure 4: The Noun Phrase Term Constructors

The truck in the city

(ONT::THE V27 (:* ONT::LAND-VEHICLE W::TRUCK) :MODS (V20))
(ONT::F V20 (:* ONT::IN-LOC W::IN) :ONT::OF V20357 :ONT::VAL V240)
(ONT::THE V240 (:* ONT::CITY W::CITY))

4. Noun Phrases in More Detail

There is a fair range of term constructors needed to handle the variety of noun phrases that occur. The ones defined so far are shown in figure 4. Here we discuss the major classes of noun phrases and how they map into the LF forms.

Names

Names are treated as definite descriptions. A special role NAME-OF relates the object to its specified name. For instance, we have

John

(ONT::THE x ONT::PERSON :NAME-OF (W::JOHN))

The NAME-OF slot is a list to accommodate multi-word names such as "The New York Times":

The New York Times

(ONT::THE x ONT::PUBLICATION :NAME-OF (W::THE W::NEW W::YORK W::TIMES))

Although rare, this form can include modifiers as well, as in the phrase,

The other John

(ONT::THE x ONT::PERSON :NAME-OF (John) :MODS f1)
(ONT::F V53605 (:* ONT::IDENTITY OTHER) :ONT::OF V53611)

Pronouns

Pronominal forms use the term constructor PRO. We also have a specially defined type in the ontology, ONT::REFERENTIAL-SEM, that includes all types of objects that can be typically referred to (objects, events, some abstract objects). Pronoun LFs uses a special role called PROFORM that indicates information relevant for how the expression relates to the context. In general, the value of this slot is simply the lexical form of the pronoun that was used. For example, the pronoun *it* would have the LF

(PRO x (:* ONT::REFERENTIAL-SEM W::IT) :PROFORM W::IT)

The pronoun *he* would have the LF

(PRO x (:* ONT::PERSON W::HE) :PROFORM W::HE)

Plural pronouns such as *them* would be as:

(PRO-SET x (:* ONT::REFERENTIAL-SEM W::THEM) :PROFORM W::THEM)

Cardinality Constraints on plurals

With plurals, most modifiers apply to each individual in the set, except for cardinality which refers to the size of the set. The cardinality of sets is captured in the role :SIZE. Other modifiers are attached in the MODS feature, and they point back to the variable identifying the set. (Note if you were mapping to extensional logic interpretation, we'd need another variable ranging over the elements of the set. The TRIPS LF does not commit to such an interpretation).

T*he three red trains --*

(ONT::THE-SET x (:* ONT::VEHICLE W::TRAIN) :MODS (v1) :SIZE 3)
(ONT::F v1 (:* ONT::COLOR_VAL W::RED) :THEME x)

Comparatives and Superlatives

The comparative and superlative adjectives all refer to scalar operations on sequences ordered according to the scalar adjective. Thus, while *cheap* is an adjective that identifies an abstract location on a scale, say ONT::COST*, cheaper* relates to objects on the ONT::COST scale, and *cheapest* identifies the object on with the minimum cost. Note that *expensive* might refer to the same scale ONT::COST, where the superlative, *most expensive*, refers to the object with the maximum cost.

The LF form captures these distinctions using a general ordering and maximizing/minimizing relation types. In addition, it inserts IMPROs for implicit arguments -- e.g., *the cheaper computer* has an implicit argument which is the computer that it is cheaper than,

The cheap computer --

(ONT::THE C1 (:* ONT::COMPUTER W::COMPUTER) :MODS (M1))

(ONT::F (:* ONT::INEXPENSIVE W::CHEAP) :of C1)

The cheaper computer --

(ONT::THE C1 (:* ONT::COMPUTER W::COMPUTER) :MODS (M1))

(ONT::F (:* ONT::MORE-VAL W::CHEAP) :SCALE ONT::MONEY-SCALE :FIGURE C1 :GROUND D1)

(ONT::IMPRO D1 ONT::REFERENTIAL-SEM)

A computer cheaper than $1000 --

(ONT::THE C1 (:* ONT::COMPUTER W::COMPUTER) :MODS (M1))

(ONT::F (:* ONT::MORE-VAL W::CHEAP) :SCALE ONT::MONEY-SCALE :FIGURE C1 :GROUND D2)

(ONT::A D2 (:* ONT::QUANTITY ONT::MONEY-SCALE) :UNIT ONT::DOLLAR :AMOUNT 1000)

The cheapest computer --

(ONT::THE C1 (:* ONT::COMPUTER W::COMPUTER) :MODS (M1))

(ONT::F (:* ONT::MAX-VAL W::CHEAP) :SCALE ONT::MONEY-SCALE :FIGURE C1 :GROUND D1)

(ONT::IMPRO D1 ONT::REFERENTIAL-SEM)

The cheapest of the computers --

(ONT::THE C1 (:* ONT::COMPUTER W::COMPUTER) :MODS (M1))

(ONT::F (:* ONT::MAX-VAL W::CHEAP) :SCALE ONT::MONEY-SCALE :FIGURE C1 :GROUND D1)

(ONT::THE-SET D1 (:* ONT::COMPUTER W::COMPUTER))

Quantifiers

The TRIPS logical form divides quantifiers into two classes. The first are the true quantifiers, which involve some form of universal iteration over a set of objects, such as *each* and *every*, and have singular agreement in English (e.g., *each dog*, not **each dogs*). The second are the cardinality quantifiers, such as *most*, *some, a few, many,* and have plural agreement. The latter are treated as relations defining the referred-to set in terms of another (possibly implicit set). Note that explicit existential quantifiers do not exist in out LF, there are either indefinites (e.g., *I had a dog)*, or arise from expletive constructions (e.g., *there are five cars)*. Expletive constructions such as *it is raining* and *there are five cars* do not have an explicit quantifier in the logical form. Rather the interpretation is capture by a existence predicate, EXISTS, which is a sense of the verb *be* (see Section 5).

True Quantifiers

These use the constructor ONT::QUANTIFIER but take an extra feature QUAN that identifies the specific quantifier used.

Each man ---

(ONT::QUANTIFIER v1 (:* ONT::MALE W::MAN) :QUAN ONT::EACH)

When the domain of quantification is explicitly indicated, it is captured with a :REFSET relation, as in

Each of the men ---

(ONT::QUANTIFIER v1 (:* ONT::MALE W::MAN) :QUAN ONT::EACH :REFSET v2)

(ONT::THE-SET v2 (:* ONT::MALE W::MAN))

Construction	Logical Form	Other quantifiers
Each man	(ONT::QUANTIFIER v1 (:* ONT::MALE W::MAN) :QUAN ONT::EACH)	Every No Some
Each of the men	(ONT::QUANTIFIER v1 (:* ONT::MALE W::MAN) :QUAN ONT::EACH :REFSET v2) (ONT::THE-SET v1 (:* ONT::MALE W::MAN))	None All
All men	(ONT::QUANTIFIER v1 (:* ONT::MALE W::MAN) :QUAN ONT::UNIVERSAL)	No
Some men	(ONT::INDEF-SET v1 (:* ONT::MALE W::MAN) :SIZE ONT::SOME)	A few Many Most Several
Some of the men	(ONT::INDEF-SET v1 (:* ONT::MALE W::MAN) :SIZE ONT::SOME :REFSET v2) (ONT::THE-SET v2 (:* ONT::MALE W::MAN))	
Five men	(ONT::INDEF-SET v1 (:* ONT::MALE W::MAN) :SIZE v2) (ONT::THE v2 ONT::NUMBER :VALUE 5)	At least six More than eight Around four
Seven of the men	(ONT::INDEF-SET v1 (:* ONT::MALE W::MAN) :SIZE v2 :REFSET v3) (ONT::THE v2 ONT::NUMBER :VALUE 7) (ONT::THE-SET v3 (:* ONT::MALE W::MAN))	

Table 1: Quantified count expressions

Cardinality Quantifiers

Cardinality quantifiers produce a logical form that defines a set in terms of some other (possibly implicit) set of objects. The REFSET feature specifies the “reference set” from which the objects are drawn, and the SIZE feature identifies the size of the subset drawn from the reference set. For example, the LF for the phrase *most of the men* would have :SIZE most and :REFSET being the men, e.g.,

Most of the men ----

(ONT::INDEF-SET x (:* ONT::MALE W::MAN) :SIZE ONT::MOST :REFSET V33)
(ONT::THE-SET V33 (:* ONT::MALE W::MAN))

Mass Terms

Mass terms, such as "sand", have different properties than count nouns such as "truck". Whereas the LF type associated with truck, ONT::TRUCK, can be viewed as a predicate that is true of any object that is a truck, it is not clear what the predicate ONT::SAND is true of. A common approach is to view ONT::SAND as being true of some quantity of sand. We’ll take this view for the sake of motivation, but note the logical form doesn’t not constrain what final semantics one might give after contextual interpretation. The indefinite form for mass terms refers to amounts of substances, and we use the term constructor ONT::SM, the mass form of the quantifier *some*. The definite description "The sand" refers to some delineable object that consists of sand, such the beach we are talking about. Note because of this treatment, you have to consider the interpretation of the predicate in order to distinguish count and mass interpretations for definite descriptions.

the water (i.e., a specific delineable amount of water identifiable in context)

(ONT::THE v1 (:* ONT::WATER W::WATER))

some beer (i.e., an indefinite quantity of beer)

(ONT::SM v2 (:* ONT::FOOD ONT::BEER))

The explicit quantity can also be specified, as in

three gallons of water

(ONT::SM v1 (:* ONT::WATER W::WATER) :QUANTITY v2)

(ONT::A v2 (:* ONT::QUANTITY F::VOLUME-SCALE) :SCALE ONT::VOLUME-SCALE :UNIT (:* ONT::VOLUME-MEASURE-UNIT ONT::GALLON) :AMOUNT v3)

(ONT::A v3 ONT::NUMBER :VALUE 3)

The phrase *the three gallons of water* would have the same LF except that the term constructor would be ONT::THE instead of ONT::SM. More details on quantity expressions are given later.

Note that many mass terms, like *beer*, can be coerced into other forms, like objects (as in a bottle of beer). Currently, we use the indefinite count specifiers for these expressions.

a beer

(ONT::A y (:* ONT::FOOD ONT::BEER))

beers

(ONT::INDEF-SET y (:* LF ::FOOD ONT::BEER))

Finally, mass terms may occur without any determiner and generally act as some type of predicate or kind. Rather than committing to a specific interpretation, we encode such forms using the constructor ONT::BARE, leaving the interpretation for discourse processing.

water

(ONT::BARE v1 (:* ONT::WATER W::WATER))

It is also possible to convert most countable objects into a mass term. Here's an example

Much of the truck

(ONT::SM v1 (:* ONT::LAND-VEHICLE W::TRUCK) :QUAN ONT::MUCH
:REFOBJECT v2)

(ONT::THE v2 (:* ONT::LAND-VEHICLE W::TRUCK))

Other Bare Terms

There is a small set of nouns in English that do not require a determiner, yet are not interpreted like bare mass NPs, such as *I was happy in school,* and *Would you come to lunch*. These terms seem to refer to culturally defined events, and vary from dialect to dialect. Depending on their use, these may refer the to temporal/spatial objects defined by an activity (e.g., A*t school we can't speak freely*). We use the BARE term constructor for these as well, leaving these complications for later interpretation:

I laughed at school

(ONT::F g1 ONT::LAUGH :AGENT i1 :MODS (loc1))
(PRO i1 ONT::PERSON :PROFORM I)
(ONT::F loc1 (:* ONT::AT-LOC AT) :OF g1 :VAL s1)
(BARE s1 ONT::SCHOOL)

WH Terms

Wh-terms such as where, when, how, and so on play a central role in questions, and also appear in the complements of verbs like know, as in *I know where the truck is*.

For questions, the wh-terms appear in the LF using the WH-TERM constructor. For example

What's the plan

(ONT::SPEECHACT V12087 ONT::SA_WH-QUESTION :FOCUS V14 :CONTENT V18)

(ONT::F V18 (:* ONT::IN-RELATION W::BE) :CO-THEME V11 :THEME V14
:TMA ((W::TENSE W::PRES)))

(ONT::THE V11 (:* ONT::PLANNING W::PLAN))

(ONT::WH-TERM V14 ONT::REFERENTIAL-SEM :PROFORM W::WHAT)

For some wh-terms, like where, when and how, the question LF is captured with both a modifying relation and a wh-term. Thus "Where can we treat him" has an LF that is equivalent to "At what location can we treat him".

Where was he seen?

(ONT::SPEECHACT V11 ONT::SA_WH-QUESTION :FOCUS V1 :CONTENT V2)

(ONT::WH-TERM V1 ONT::REFERENTIAL-SEM :PROFORM W::WHERE)

(ONT::F V2 (:* ONT::ACTIVE-PERCEPTION W::SEE) :THEME V3 :MODS (V4)
:TMA ((W::TENSE W::PAST) (W::PASSIVE +)))

(ONT::PRO V3 (:* ONT::PERSON W::HE) :PROFORM W::HE)

(ONT::F V4 (:* ONT::WH-LOCATION W::WHERE) :OF V2 :VAL V1)

WH-NP's as Complements

When wh-terms are used as complements to verbs like know or find out, we treat them as wh-terms descriptions with a special role :suchthat

I know what arrived.

(ONT::F V21028 (:* ONT::FAMILIAR W::KNOW) :THEME V2 :COGNIZER V5)

(ONT::WH-TERM V2 (:* ONT::REFERENTIAL-SEM W::WHAT) :PROFORM W::WHAT :SUCHTHAT V6)

(ONT::F V6 (:* ONT::ARRIVE W::ARRIVE) :THEME V2)

(ONT::PRO V5 (:* ONT::PERSON W::I) :PROFORM W::I)

Show me where the car stopped.

(ONT::F V23438 (:* ONT::SHOW W::SHOW) :ADDRESSEE V1 :THEME V7 :AGENT V9)

(ONT::WH-TERM V7 (:* ONT::SPATIAL-LOC W::WHERE) :SUCHTHAT V2)

(ONT::F V2 (:* ONT::STOP-MOVE W::STOP) :THEME V3)

(ONT::THE V3 (:* ONT::LAND-VEHICLE W::CAR))

(ONT::IMPRO V9 ONT::PERSON :PROFORM W::ME)

(ONT::IMPRO V9 ONT::PERSON :PROFORM W::*YOU*)

Possessives

The possessive construction is captured in the logical form using a role relation :ASSOC-POSS, which denotes abstract possession. The exact relations between the possessor and possessed, say ownership, or control, or proximity, can only be determined by contextual interpretation.

The man's cat

(ONT::THE V1 (:* ONT::NON-HUMAN-ANIMAL W::CAT) :ASSOC-POSS V2)

(ONT::THE V2 (:* ONT::MALE MAN))

Demonstratives

Demonstratives are treated as definite descriptions and the lexical item is placed in the PROFORM relation as done with pronouns. This allows reference resolution processes to have strategies specific for each word. For example

These trucks

(ONT::THE-SET x (:* ONT::LAND-VEHICLE W::TRUCK) :PROFORM ONT::THESE)

That truck

(ONT::THE x (:* ONT::LAND-VEHICLE W::TRUCK) :PROFORM ONT::THAT)

One

One used as a head noun indicates no restriction on the LF type, and *one* is placed in the PROFORM to enable special referential processing.

The red one

(ONT::THE x (:* ONT::REFERENTIAL-SEM W::ONE) :MODS c1 :PROFORM ONE)
(ONT::F c1 (:* ONT::RED W::RED) :SCALE ONT::COLOR-SCALE :OF x)

Gerunds and Nominalizations

Most verbs support forms that make various noun phrases. When a specifier is explicit we build the obvious form. For example

The investigation of the crime

(ONT::THE l1 (:* ONT::SCRUTINY W::INVESTIGATION) :THEME c2)

(ONT::THE c2 (:* ONT::ACTIVITY W::CRIME))

Identical in form except for the constructor to the verbal form

They investigated the crime

(ONT::F l1 (:* ONT::SCRUTINY W::INVESTIGATION) :COGNIZER p1 :THEME c2)

(ONT::PRO-SET p1 (:* ONT::REFERENTIAL-SEM W::THEY) :PROFORM W::THEY)

(ONT::THE c2 (:* ONT::ACTIVITY W::CRIME))

Gerunds are treated similarly. For example

The burning of the city

(ONT::THE d1 (:* ONT::BURN W::BURN) :THEME c1)

(ONT::THE c1 ONT::CITY)

Bare gerunds, which might refer to events or to kinds of activities, retain the ONT::BARE constructor, leaving disambiguation to contextual processing

Burning the city was fun

(ONT::F e1 (:* ONT::HAVE-PROPERTY W::BE) :THEME b1 :PROPERTY p1)

(ONT::BARE b1 (:* ONT::BURN W::BURN) :THEME c1)

(ONT::THE c1 ONT::CITY)

(ONT::F p1 (:* ONT::ENTERTAINMENT-VAL W::FUN) :OF b1)

Enumerated Constructions

Conjoined noun phrases require the construction of a set of objects. A set is constructed for the conjoined phrase that uses the special role :MEMBERS to list the items involved. For example.

Dogs and cats

(ONT::THE-SET V44 ONT::NONHUMAN-ANIMAL :OPERATOR ONT::AND :SEQUENCE (V40 V46))

(ONT::THE V40 (:* ONT::NONHUMAN-ANIMAL W::DOG))

(ONT::THE V46 (:* ONT::NONHUMAN-ANIMAL W::CAT))

Neither dogs nor cats

(ONT::QUANTIFIER v1 ONT::NONHUMAN-ANIMAL :OPERATOR ONT::NONE-OF :SEQUENCE (v2 v3))

(ONT::INDEF-SET V40 (:* ONT::NONHUMAN-ANIMAL W::DOG))

(ONT::INDEF-SET V46 (:* ONT::NONHUMAN-ANIMAL W::CAT))

Either the dog or the cat

(ONT::A v1 ONT::NONHUMAN-ANIMAL :OPERATOR ONT::ONE-OF :SEQUENCE (v2 v3)

(ONT::THE v2 (:* ONT::NONHUMAN-ANIMAL W::DOG))

(ONT::THE v3 (:* ONT::NONHUMAN-ANIMAL W::CAT))

4. Time, Numbers, and Locations

Temporal Objects

Temporal expressions fall into two categories, those describing particular times according to some clock system (e.g., the time of day), and those describing durations of time (e.g., the length of a movie). Here we describe the first use, reference to temporal “locations”. Temporal durations are handled in the next section as an example of quantity terms.

Clock-times are often underspecified and require contextual processing. We say Saturday but don't mention which Saturday is meant, or 3 PM without mentioning the day. These terms serve to constrain the range of possible times that could be intended, and contextual interpretation

Slot Name	Onotology Type of value	Typical values	Example text
:YEAR	ONT::NUMBER	2004, 2010	2010, Two thousand ten
:MONTH	ONT::MONTH-NAME	ONT::JULY	July
:DAY	ONT::NUMBER	1,..., 31	July **4th**
:DOW	ONT::DAY-NAME	ONT::MONDAY, ONT::TUESDAY	Monday, Tuesday
:AM-PM	ONT::TIME-OBJECT	ONT::AM, ONT::PM, ONT::MORNING,	AM, A.M., morning, evening
:HOUR	ONT:::NUMBER	1,..., 12	1,..., 12, one, ...,twelve
:MINUTE	ONT::NUMBER	1,...,59	5:**30**, 6:**45**
:CENTURY	ONT::NUMBER	1,2,...	The third century
:ERA	ONT::ERA	ONT::AD, ONT::BC	3rd Century **BC**
:PHASE	ONT::STAGE-VAL	ONT::MID, ONT::EARLY, ONT::LATE	**Mid**-July

Table 2: The slots of ONT::TIME-LOC

would identify the intended one. These are classified in the ontology as ONT::TIME-LOC, which has slots for type of clock-time attribute, as shown in Table 2.

Here are some examples:

Monday July 4

```
(ONT::THE V3 ONT::TIME-LOC
            :DAY 4
            :MONTH  (:* ONT::MONTH-NAME JULY)
            :DOW (:* ONT::DAY-NAME MONDAY)))
```

Five PM

```
(ONT::THE V7039 ONT::TIME-LOC
    :AM-PM (:* ONT::TIME-OBJECT W::PM)
    :HOUR 5)
```

Numbers

Numbers are the most basic quantity terms. Below we will deal with other quantity terms that expressed quantities in terms of units and measures. The LF of an expression like five is not simple an atom such as 5 because we need to handle modifiers, as in at least five, a few hundred, and not more than seven. These express constraints on values and can appear almost anywhere a simple number may. The system simplifies purely numeric expressions into conventional form, as in the following examples.

***Five** dogs*

```
(ONT::A v1 ONT::NUMBER :VALUE 5)
```

***At least five** dogs*

```
(ONT::A v1 ONT::NUMBER :MODS (v2))
(ONT::F v2 (:* ONT::QMODIFIER W::MIN) :OF v1 :IS v3)
```

(ONT::A v3 ONT::NUMBER :VALUE 5)

Approximately five *dogs*

(ONT::A v1 ONT::NUMBER :MODS (v2))

(ONT::F v2 (:* ONT::QMODIFIER W::APPROXIMATE) :OF v1 :IS v3)

(ONT::A v3 ONT::NUMBER :VALUE 5)

Numeric expressions that involve units, like hundred, thousand, etc, have an LF that retains the form of the linguistic expression, rather than collapsing to the mathematical representation. Thus we have expressions like the following

Five hundred

(ONT::A v1 ONT::NUMBER :VALUE 500)

Two thousand three hundred and five

(ONT::A v1 ONT::NUMBER :VALUE 2305)

However, for vague number expressions we have to retain the units explicitly, as in

Several hundred

(ONT::A v1 (:* ONT::NUMBER-UNIT W::HUNDRED) :QUAN v2)

(ONT::A v2 ONT::NUMBER :VALUE W::SEVERAL)

Quantity Terms

There are many expressions of measurement in language, that combine a numerical quantity and some unit on a scale. This includes expressions such as three miles, many liters, several hours, eight days, and so on. These generally map to expressions of type QUANTITY, as in

Three miles

(ONT::A v1 (:* ONT::QUANTITY ONT::LENGTH-SCALE)

:UNIT (:* ONT::LENGTH-UNIT W::MILE) :AMOUNT v2)

(A v2 ONT::NUMBER :VALUE 3)

Several pounds

(ONT::A v1 (:* ONT::QUANTITY ONT::WEIGHT-SCALE)

:UNIT (:* ONT::WEIGHT-UNIT W::POUND) :AMOUNT ONT::SEVERAL)

At least ten dollars

(ONT::A v1 (:* ONT::QUANTITY ONT::MONEY-SCALE)

:UNIT (:* ONT::MONEY-UNIT W::DOLLAR) :AMOUNT v2)

(ONT::A v2 ONT::NUMBER :MODS (v3))

(ONT::F v3 (:* ONT::QMODIFIER W::MIN) :OF v2 :IS v4)

(ONT::A v4 ONT::NUMBER :VALUE 10)

5. Verbs

Tense, Aspect and Modality

There is information expressed in the lexical and structural forms that identify tense, aspect and modality, that have not yet been discussed in the logical form language. All this information is important for contextual interpretation, and meaning so we package it up in a special structure that is the value of a new role :TMA, which can only appear in "F" structures. These TMA structures are themselves sets of feature values as shown in table 5.

Feature	Values	Realization	Example
:TENSE	:PRES	Present tense	I like ...
	:PAST	Past tense	I liked ...
	:FUT	Future construction	I will like ...
:PROGR	+/-	Progressive	I am liking ...
:PERF	+/-	Perfective	I had liked ...
:PASSIVE	+/-	Passive	I was liked
:MODALITY	ONT::ABILITY	Can auxiliary	I can run
	ONT::EMPHASIS	do	I did run
	ONT::MUST	must	I must run
	ONT::SHOULD	should	I should run
	ONT::FUTURE	will, shall	I will run
	ONT::POSSIBILITY	may, might	I might run
	ONT::CONDITIONAL	could, would	I could run
	ONT::GOING-TO	going to, gonna	I'm going to run

Figure 5: The TMA Auxiliaries

Some examples of full logical forms follow:

I had not seen the ice

(ONT::F v0 (:* ONT::ACTIVE-PERCEPTION W::SEE) :THEME v1 :EXPERIENCER v2 :TMA ((TENSE PAST) (PERF +) (NEGATION +))))

(PRO v1 (:* ONT::PERSON W::I) :PROFORM W::I)

(ONT::THE v2 (:* ONT::SUBSTANCE W::ICE))

I should have gone

(ONT::F v1 (:* ONT::MOVE W::GO) :AGENT v2 :TMA ((W::TENSE W::PRES) (W::MODALITY (:* ONT::OBLIGATION W::SHOULD)) (W::PERF +)))

(ONT::PRO v2 (:* ONT::PERSON W::I) :PROFORM W::I)

I can't see it

(ONT::F v1 (:* ONT::ACTIVE-PERCEPTION SEE) :THEME v1 :EXPERIENCER v2 :TMA ((TENSE PRES) (MODALITY (:* ONT::ABILITY CAN)) (NEGATION +)))

(ONT::PRO v2 (:* ONT::PERSON W::I) :PROFORM W::I)

(ONT::PRO v1 (:* ONT::REFERENTIAL-SEM W::IT) :PROFORM W::IT)

Ellided Forms

Because we treat auxiliaries as augmentations to main verbs, we need a special treatment for elided forms. We introduce a special LF for the elided verb phrase. This would serve as a signal to

I can't

(ONT::F V6 ONT::ELLIPSIS :THEME V2 :TMA ((TENSE PRES) (MODALITY (:* ONT::ABILITY W::CAN)) (NEGATION +))))

Semantic Roles

The TRIPS ontology has a fixed set of semantic roles that are critical for defining the ontology and the structure of verbal forms. Rather than viewing semantic roles as merely signaling different argument positions for predicates, the semantic roles have inferential import in their own right. In addition, the TRIPS ontology makes a key distinction between what we call **essential roles**, those that identify various ways objects may relate to a core event, such as AGENT, THEME, etc, and the various other relations that are actually treated as prepositional/ adverbial meanings, such as GOAL, AT-LOC, INSTRUMENT, and so on. The latter set is open ended, corresponding to prepositions and adverbs, while the essential roles are limited and fixed. The roles are organized hierarchically into three abstract classes, indicating three key ways arguments relate to their event: CAUSE, designating a causal type relationship; AFFECTED, designating an object controlled or changed by the event; and, THEME, an abstraction that is not changed by the event but it critical to the event's meaning. Each of these have specializations that cover many of the roles that are typically seen in various role taxonomies. Table 6 summarizes the roles.

	Definition	Example	Verbnet	Lirics
CAUSE	Object that plays a causal and/or controlling role wrt the event	***The storm*** tore the flag.	Cause	Cause
AGENT	object capable of self-initiated causation, and is aware of the causal attribution	***He*** broke the window	Agent, Actor, Actor1	Agent
COGNIZER	and event describes some mental process	***He*** calculated the total	Agent	
INSTRUMENT	is an object incapable of self-initiated causation	***The hammer*** broke the window	Instrument	Instrument
STIMULUS	and triggers a response through perception	***The song*** saddened John	Stimulus	
AFFECTED	Object that is controlled /changed in some way or plays a passive role in the event	I sanded ***the board;*** **The dollar** is worthless	Patient, Patient1, Agent, Topic, Attribute, Patient2, Benefactive, Theme	Patient + (Theme) Pivot
EXPERIENCER	and must be aware of process described by the verb	***He*** saw the pelican	Experiencer, Patient1	(Patient)
ADDRESSEE	and is intended recipient of communication	I told ***him*** the story	Recipient	
PRODUCT	and is a new object created by the event	He created ***a masterpiece,*** They baked ***a cake,***	Product	Result
THEME	Object is an abstraction that is not changed by the event (i.e., AFFECTED) , nor controls the event (CAUSER)		Stimulus,Theme, Theme1, Theme2, topic	(Theme)
PROPOSITION	and is a proposition (i.e., something that is true or false)	I believe ***that the world is flat***	Proposition	
PROPERTY	and is a property ascribed to another participant of the event	It had ***great potential***	Predicate, Attribute	Attribute
AMOUNT	and denotes a quantity that characterizes another participant	It weighs ***seven pounds***	Asset	Amount

Table 6: The Essential Verbal Semantic Roles and Verbnet and Lirics correspondences

Appendix A: Some Key Verbs

Main verb be

The main verb be has three main forms. The first use, ONT::HAVE-PROPERTY, associates an object with a property which is realized by an adjective, PP, or other predicative form. To save space, we will suppress the LF for the various pronominal forms.

It was red

(ONT::F h1 (:* ONT::HAVE-PROPERTY W::BE) :AFFECTED it1 :PROPERTY p1)

(ONT::F p1 (:* ONT::RED W::RED) :ONT::OF it1)

It is in the truck

(ONT::F h1 (:* HAVE-PROPERTY W::BE) :AFFECTED it1 :PROPERTY p1)

(ONT::F p1 (:* ONT::IN-LOC W::IN) :ONT::OF it1 :ONT::VAL a1)

(ONT::THE a1 (:* ONT::LAND-VEHICLE W::TRUCK))

He was late

(ONT::F h2 (*: ONT::HAVE-PROPERTY W::BE) :AFFECTED he1 :PROPERTY p2)

(ONT::F p2 (:* ONT::SCHEDULED-TIME-MODIFIER W::LATE) :OF he1)

The second sense of be indicates a relationship between objects, which often is equality, but also might involve some contextually-defined relations. For instance, the utterance *Three miles is four hours* states that the relation "time to travel" relates three miles to four hours. The predicate ONT::IN-RELATION is used for this, and determining the exact relation is left for contextual processing.

It is the best truck.

(ONT::F be1 (*: ONT::IN-RELATION BE) :AFFECTED it1 :THEME b1)

(ONT::THE b1 (:* ONT::LAND-VEHICLE TRUCK) :MODS (bd1))

(ONT::F bd1 (:* ONT::MAX-VAL W::GOOD) :FIGURE b1 :SCALE ONT::ACCEPTABILITY-VAL)

Three miles is four hours

(ONT::F v0 (:* ONT::IN-RELATION BE) :AFFECTED v1 :THEME v2)

(ONT::A v1 (:* ONT::QUANTITY DURATION-SCALE) :UNIT(:* ONT::DISTANCE-UNIT ONT::HOUR)) :AMOUNT v2)

(ONT::A v2 ONT::NUMBER :VALUE 4)

(ONT::A v3 (:* ONT::QUANTITY LENGTH-SCALE) :UNIT(:* ONT::LENGTH-UNIT ONT::MILE)) :AMOUNT v4)

(ONT::A v4 ONT::NUMBER :VALUE 4)

The third sense is existence, and is typically seen in utterances like "there is the truck".

There is a person there

(ONT::F v0 ONT::EXISTS :AFFECTED v1 :TMA ((TENSE PRES)))

(ONT::A v1 (:* ONT::PERSON PERSON) :MODS (v2))

(ONT::F v2 (:* ONT::THERE THERE) :OF v1 :VAL v3)

(ONT::IMPRO v3 ONT::LOCATION :PROFORM W::THERE)

Main verb have

The verb have is another verb that takes on many different meanings depending on its arguments. We do not attempt to capture these variations in entailments in the sense of the verb, but rather view the entailments arising in later processing, and based on the verb-argument combinations. Thus the senses for have reflect the structural restrictions seen in language. The sense ONT::HAVE takes an AFFECTED and THEME role and asserts some relationship between the two (e.g., in a prototypical case, possession). A second sense, ONT::UNDERGO, has an AFFECTED role and a THEME that is an event/situation, and roughly asserts that the AFFECTED undergoes the event described (e.g., he has a headache). Finally, the sense ONT::MAKE-IT-SO involves a causal agent force an object to undergo some event, as in *He had me open the can*. The senses as shown in figure 6, along with the fine-grained wordnet senses. Only one wordnet sense (the third one), requires splitting examples between our senses, since an idea is an abstract object in our ontology, whereas feelings, emergencies and headaches are situations.

Sense	**Comment**	**Roles**	**Examples**	Sense #
HAVE	AFFECTED in relation to THEME	:AFFECTED :THEME	She has $1000 in the bank. She has two daughters	1
			This restuarant has the best chef in France.	2
			I have three houses in Florida.	4
			He has a postdoc/lover.	7
			I don't have any money left. They have two years before they retire.	9
			She has arthritis	12
			I had an idea	3
			I had a letter from them	15
			I had her (archaic)	19
HAVE-EXPERIENCE	AFFECTED undergos a situation described by THEME	:AFFECTED :THEME	I have a feeling. We have an emergency, I have a headache.	3
			He had a reception/party.	8
			We have a fine mess. What do we have here?	10
			The stocks had a fast runup.	11
			I won't have this dog in the house	14
			She had an accident.	16
			The team had 4 goals.	17
			My wife had twins yesterday.	18
MAKE-IT-SO	AGENT causes AFFECTED to be related is a situation described by THEME	:AGENT :AFFECTED :THEME	He had me in for a big surprise.	5
		:AGENT :AFFECTED :THEME	They had me buy a VCR.	13
CONSUME	An agent consumes some substance	:AGENT :AFFECTED	Have another bowl of soup. We had fish for dinner	6
NECESSITY	An agent is required to do something	:AFFECTED :THEME	I had to go, The files had to be destroyed.	

Figure 6: The senses of "have" and their WordNet mappings

Here are a few examples uses some of the senses.

I have a car

(ONT::F V444588 (:* ONT::HAVE W::HAVE) :AFFECTED V44 :THEME V45 :TMA ((TENSE PRES)))

(PRO v44 (:* ONT::PERSON W:I) :PROFORM W::I)

(ONT::A v45 (:* ONT::VEHICLE W::CAR))

We have an emergency

(ONT::F h1 (:* ONT::HAVE-EXPERIENCE HAVE) :AFFECTED we1 :THEME ha1)

(PRO i1 (SET-OF (:* ONT::PERSON W:WE)) :PROFORM W::we)

(ONT::A ha1 (:* ONT::EVENT W::EMERGENCY))

I had Fred go

(ONT::F v0 (:* ONT::MAKE-IT-SO W::HAVE) :AGENT v1 :AFFECTED v2 :THEME v3 :TMA ((TENSE PAST)))

(PRO v1 ONT::PERSON :PROFORM I)

(ONT::F v3 (:* ONT::MOVE GO) :AGENT v2)

(ONT::THE v2 ONT::PERSON :NAME-OF FRED)

I had some water

(ONT::F v0 (:* ONT::CONSUME HAVE) :AGENT v1 :AFFECTED v2 :TMA ((TENSE PAST)))

(PRO v1 ONT::PERSON :PROFORM I)

(A v2 (:* ONT::FOOD WATER))