[1]Semantics and Logical Form:

* The representation of context-independent meaning is called the **logical form.**

* The **process** of mapping a sentence to its **logical form** is called **semantic interpretation**

* The process of mapping the logical form to the final **knowledge representation (KR)** language is called **contextual interpretation.**

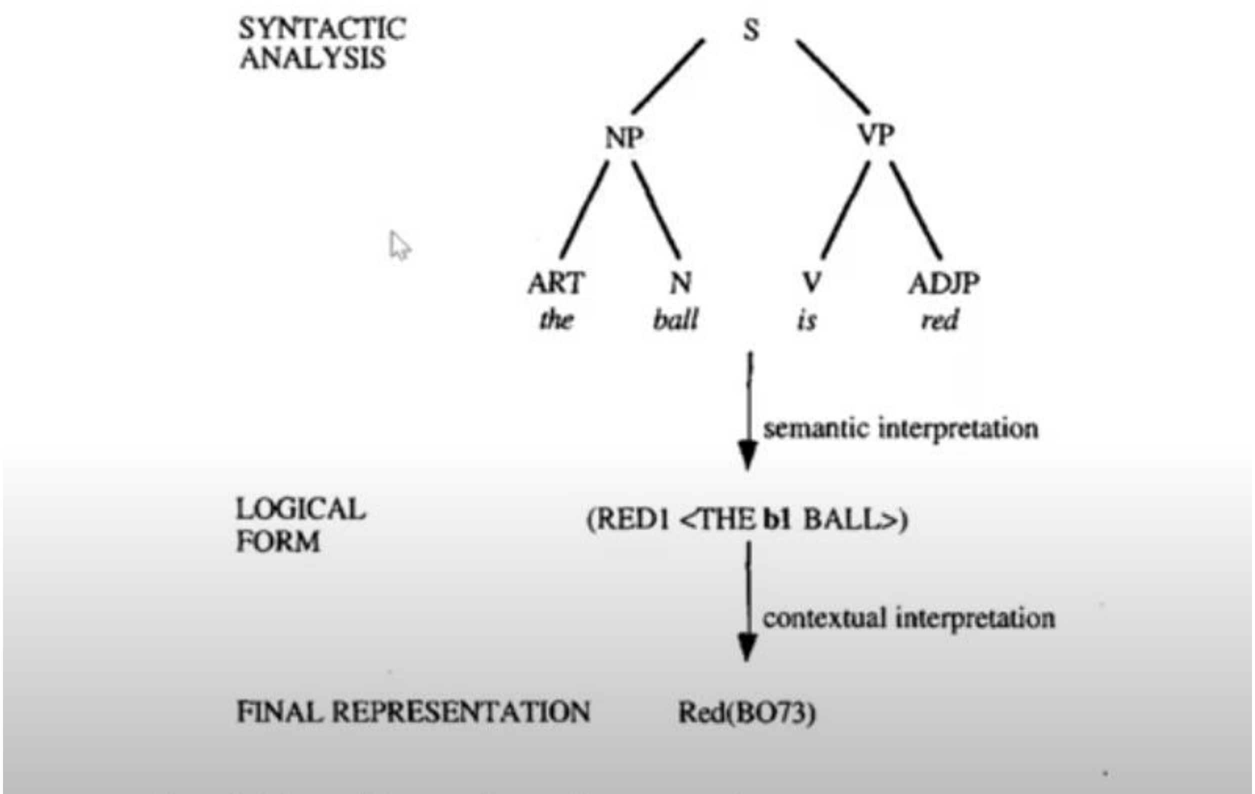

Figure 8.1 Logical form as an intermediate representation

* Indexical terms: terms that are defined by context.

* The **pronouns** "I" and "you" are indexical because their **interpretation** depends on the context of who is **speaking and listening**.

+ Different aspects of language:

— **interpretation** of tense and determining the **scope of quantifiers**, depend on context
— **Can't be uniquely** determined at the logical form level.

Logical Form:

* Itis possible to **consider** meaning in the **absence of context.**
* The **representation** of meaning in the **absence of context** is termed **logical form.**

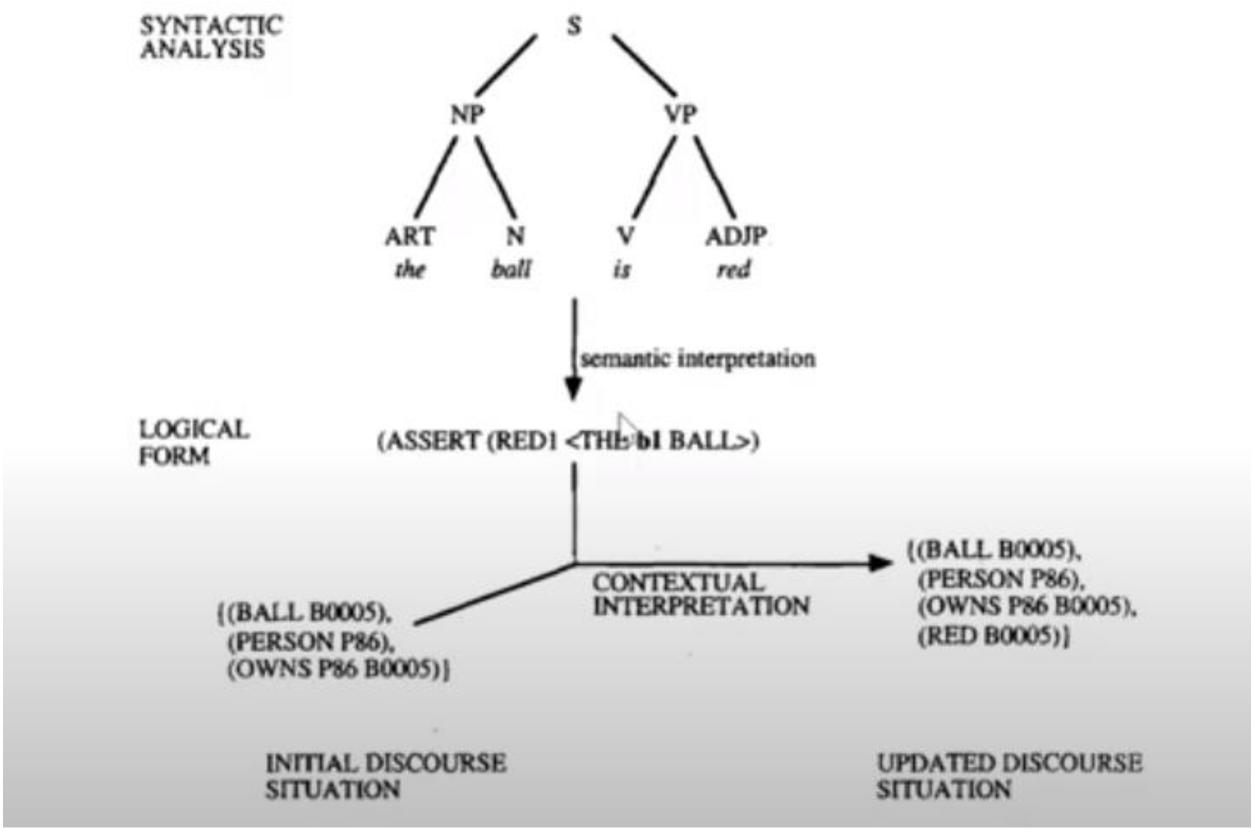

Fig:8.2 Logical form as a function

Word Senses and Ambiguity:

+ Words may have more than **one sense.**

+ For **example**, ball could mean a social **event (SOCIAL-BALL),** or a spherical toy (TOYBALL)

+ For go you might **find** synonyms such as **move, depart, pass, vanish, reach, extend, and set out.** Many of these highlight a different sense of the verb go.

+ **one of the senses** of go will be identical to one of the **senses of depart**

+ Some words, like take, have many different senses (some say 57 senses for take). 3

+ If the intended sense is not clear, we have "word sense ambiguity".

+ The **different senses** can be organized into a set of broad **classes of objects** by which we classify the world.
+ The set of **different classes** of objects in a representation is called its **ontology**

«+ The major classes were
— substance (physical objects),
— quantity (such as numbers),
— quality (such as bright red),
— relation,
— place,
— time,
— position,
— state, v
— action,
~ and affection.
+ other classes such as events, ideas, concepts, and plans.

Lexical ambiguity:

+ **Two** of the most influential classes are actions and events.

+ **Like all objects** in the ontology, actions and events can be referred to by **pronouns**, as in the discourse fragment

We lifted the box. It was hard work.

+ The pronoun "it" refers to the action of lifting the box.

We laughed and sang at the football game

« describes a set of activities performed at a particular time and location, described as the **situation** "the football game".

Semantic ambiguity:

* The word **"kid"** seems to be ambiguous between **a baby goat** and **a human child**: semantically ambiguous.

*** verb senses:** | ran last year and George did too" could mean that we **both** were candidates in an **election** or that we both ran some race, but it would be difficult to read it as a **mixture of the two.**

* Thus the word run is ambiguous between the senses.

* At the logical form level, words are **quite often ambiguous**, as it may take context to decide the intended sense.

[2]Word Senses and Ambiguity:

- There are other types of ambiguity:

referential ambiguity:	*Jon is angry with Jim. He bites him.*	
structural ambiguity:	*Happy cats and dogs live on the farm.*	syntactic structure
	Every man loves a woman.	semantic structure
there can be mixes:	*The major exhibits age.*	

* Important tasks of **semantic interpretation** is to utilize constraints such as to help reduce the number of possible senses for each word.

[3]The basic Logical Form Language:

The Basic Logical Form Language

This section defines a formal language of logical forms, resembling FOPC (first order predicate calculus):

terms

constants or expressions that describe objects: FIDO1, JACK1

predicates

constants or expressions that describe relations or properties: BITES1.

Each predicate has an associated number of arguments - BITES1 is binary (unary = 1 argument; ternary = 3 arguments; *n*-ary = *n* arguments).

Logical Form Language 2

propositions

a predicate followed by the appropriate number of arguments: (BITES1 FIDO1 JACK1), (DOG1 FIDO1) - *Fido is a dog.*

More complex propositions can be constructed using **logical operators**:

(NOT (LOVES1 SUE1 JACK1))
(& (BITES1 FIDO1 JACK1) (DOG1 FIDO1))

Note that *and* does not always "translate" as logical & - e.g. it may suggest temporal sequence: *I went home and had a drink* compared to *I had a drink and went home.*

The Basic Logical Form Language

«+ **Defines a language** in which you can **combine** these elements to form meanings for more complex expressions.

+ The **word senses** will serve as the **atoms or constants** of the representation.

+ These constants can be classified by the types of things they describe.

+ **Ex:** constants that describe objects in the world, including abstract objects such as events and situations, are called **terms.**

+ Constants that describe relations and properties are called **predicates.**

+ A **proposition** in the language is formed from a predicate followed by an appropriate number of terms to serve as its arguments.

+ For **Example,** the proposition corresponding to the sentence **"Fido is a dog"** would be constructed from the term FIDO1 and the predicate constant DOG1 and is written as

(DOG1 FIDO1)

* Predicates that take a single argument are called **unary predicates or properties;**

* Those that take two arguments, such as LOVES], are called **binary predicates**; and

* Those that take n arguments are called **n-ary predicates**. The proposition corresponding to the sentence Sue loves Jack would involve a binary predicate LOVES1 and would be written as

(LOVES1 SUE1 JACK1)

« Different **word classes in English** correspond to different types of constants in the **logical form.**

* Proper names, such as Jack, have word senses that are **terms;**

* Common nouns, such as dog, have word senses that are **unary predicates;**

* and verbs, such as run, love, and put, have word senses that correspond to **n-ary predicates,** where **n depends** on how many terms the **verb subcategorizes** for.

Logical Form Language

* **Logical operators:** for complex propositions constructions

= NOT allows you to construct a proposition that says that some proposition is not true

* The **proposition** corresponding to the sentence **Sue does not love Jack** would be

(NOT (LOVES1 SUE1 JACK1))

* Connective proposition:

* For **example**, the logical form of the sentence "Jack loves Sue or Jack loves Mary" would be **(OR1 (LOVES1 JACK1 SUE1) (LOVES1 JACK1 MARY1)).**

* The logical form language will allow both operators corresponding to word senses.

Quantifier:

+ "A man entered the room. He walked over to the table.”

« The first sentence introduces a new object to the discussion, namely some man.

+ The man introduced existentially in the first sentence is referred to by the pronoun "He" in the second sentence.

* So variables appear to continue their existence after being introduced.

+ To allow this, each time a discourse variable is introduced, it is given a unique name not used before.

+ Under the right circumstances, a subsequent sentence can then refer back to this term.

Quantifier	Use	Example
THE	definite reference	the dog
A	indefinite reference	a dog
BARE	bare singular NP (mass term) or	water, food
BARE	bare plural NP (generics)	dogs

Figure 8.4 Some common quantifiers

+ (quantifier variable : restriction-proposition body-proposition)
+ For instance, the sentence "Most dogs bark" would have the logical form
(MOST1 d1 (DOG d1) (BARKS1. d1)
+ This means that most of the objects d1 that satisfy (00G1.d1) also satisfy.
(BARKS1 d1).
+ (MOST d2 : (BARKS1 d2) (DOG1 d2))

* The meaning of the sentence "Most barking things are dogs". (Different than previous one)

* Important class of generalized quantifiers corresponds to the articles "the" and "a".

* The sentence "The dog barks" would have a logical form

(THE x: (DOG1 x) (BARKS1 x))

* "The happy dog barks" will involve a restriction that is a conjunction, namely
(THE x (& (DOG1 x) (HAPPY x)) (BARKS1 x))

* This will be true only if there is a contextually unique x such that (& (DOG1 x) (HAPPY x)) is true, and this x barks.

:

* Predicate operator is introduced that takes a predicate as an argument and produces a new predicate.

* For plurals the predicate operator PLUR will be used.

+ If DOG1 is a predicate that is true of any dog, then (PLUR DOG1) is a predicate that is true of any set of dogs.

* Thus the representation of the meaning of the sentence "The dogs bark" would be

{THF x: ((PLUR DOG1) x) (BARKS1 x))

+ Plural noun phrases introduce the possibility of a new form of ambiguity.

+ The dogs bark is that there is a specific set of dogs, and each one of them barks. This is called the distributive reading, since the predicate BARKS is distributed over each element of the set.

+ In contrast, consider the sentence The dogs met at the corner.

~ In this case, t makes no sense to say that each individual dog met; rather the meeting is true of the entire set of dogs.

— This s called the collective reading.

+ Some sentences allow both interpretations and hence are ambiguous.

~ For instance, the sentence Two men bought a stereo can mean that two men each bought a stereo (the distributive reading), or that two men bought a stereo together (the collective reading).

Predicative and model:

Logical Form Language Part 4

predicate operator

We also need a way to handle plurals as in *the dogs bark*.

A new type of thing called a **predicate operator**is introduced that takes a predicate as an argument and produces a new predicate.

For plurals, PLUR: if DOG1 is true of any dog, then (PLUR DOG1) is true of any set of dogs with more than one member:

(THE **x** : ((PLUR DOG1) **x**) (BARKS1 **x**)).

Logical Form Language Part 5

modal operator

used for verbs like *believe, know, want*, for tense, and other purposes. *Sue believes Jack is happy* becomes

(BELIEVE SUE1 (HAPPY JACK1)).

Modal operators may exhibit **failure of substitutivity:**

JACK1 may = JOHN22 (i.e. the individual known as Jack may also be called John, e.g. by other people)

However, *Sue believes John is happy* may not be true, e.g. because Sue may not know that JACK1 = JOHN22.

Modal:

+ These are needed to represent the meaning of verbs such as believe and want, for representing tense, and for many other constructs.

+ Modal operators look similar to logical operators but have some important differences.

+ Specifically, terms within the scope of a modal operator may have an interpretation that differs from the normal one.

+ This affects what conclusions you can draw from a proposition.

+ For example, assume that Jack is also known as John to some people.

+ There are two word senses that are equal; that is, JACKL = JOHN22.

+ With a simple proposition, it doesn't matter which of these two constants is used: If (HAPPY JOHN22] is true then (HAPPY JACK1) is true, and vice versa.

«This is true even in complex propositions formed from the logical operators.

+_f((OR (HAPPY JOHN) (SAD JOHNL)) is true, then so is (OR (HAPPY JACK1) (SAD JACK1)), and vice versa.

Modal Operator:

+ The same propositions within the scope of a modal operators such as BELIEVEL, however, are not interchangeable.
«For instance, if Sue believes that Jack is happy, that is, **(BELIEVE SUE1 (HAPPY JACK1))**
. then it does not necessarily follow that Sue believes John is happy, thats, **(BELIEVE SUE (HAPPY JOHN22))**
+ because Sue might not know that JACK1 and JOHN22 are the same person.
+ Thus you cannot freely substitute equal terms when they occur within the scope of a modal operator.
* This is often referred to as the **failure of substitutivity** in modal contents.

Tense:

* An important class of modal operators for natural language are the tense operators, PAST, PRES, and FUT.
* Represent the difference in meaning between John sees Fido, John saw Fido, and John will see Fido, namely as the propositions.
Tense

* **(PRES (SEES1 JOHN1 FIDO1))**

« **(PASY (SEES1 JOHN1 FIDO1))**

+ (FUT (SEES1 JOHN1 FIDO1))

+ They exhibit the **failure of substitutivity.**

+ For example, consider the operator PAST, and assume two onstants, say JOHN1 and PRESIDENT, that are equal now, indicating that John is currently the president. But in the past, Johnwas not the president, so JOHN. did not equal PRESIDENTI.

+ Given this and the fact that John saw Fido in the past, (PAST (SEES1 JOHN1 FIDO 1)), you cannot conclude that the president saw Fido in the past, that is, (PAST (SEES1 PRESIDENT1 FIDO1)), since John was not the president at that time.

Tense:

+ A proposition and its negation can both be true in the past (but at different times).

* Thus it is possible for both the sentences John was happy and John was not happy to be true; that is, (PAST (HAPPY JOHN!) and (PAST (NOT (HAPPY JOHN1))) are both true.

Generalized Quantifier:

+ Constructs such as generalized quantifiers are treated syntactically like terms and appear in the position indicated by the syntactic structure of the sentence.

+ They are marked using angle brackets to indicate the scoping abbreviation.

+ For example, the logical forms for the sentence “Every boy loves

+ a dog" are captured by a single ambiguous form

+ (LOVES1 <EVERY bl (BOY1 b1)> <A d1 (DOG1 d1)>)

+ This abbreviates an **ambiguity** between the **logical form**

+ (EVERY bl : (BOY1b1) (A d1 (DOG1 d1) (LOVES1 b1 d1)))

+ and

+ (Ad1:(DOG1 d1) (EVERY bl : (BOY1 bl) (LOVES bl d1)))

+ <EVERY bi (BOY b1)>
will often be abbreviated as

+ <EVERY bl BOY>

[4] Encoding Ambiguity in the Logical Form:

* The greatest source of ambiguity in the **logical form** comes from the fact that most words have **multiple senses.**

* Some of these **senses** have different **structural properties,** so they can be eliminated given the **context of the surrounding sentence.**

* But often words have different senses that have **identical structural constraints.**

* Build a separate logical form for each possible combination of senses for the words in the Sentence:**solution.**

+ For **example,** the noun "ball" has at least two senses:

BALLI, the object used in **games,** and **BALL2**, the social event involving **dancing.**

+ Thus the sentence **"Sue watched the ball"** is ambiguous out of context.

+ A **single logical form** can represent these **two possibilities**, however:

1. (THE **b1** : ({BALL1 BALL2} **b1**) (PAST (WATCH1 SUE1 **b1**)))

This abbreviates two possible logical forms, namely

2. (THE b1 : (BALLL ba) (PAST (WATCH1 SUE1 b1)))

and

3 (THE bl : (BALL? b1) (PAST (WATCH1 SUE1 b1)))

« "Every boy didn't run" is ambiguous between the reading in which some boys didn't run and some did, that is,

(NOT (EVERY bl : (BOY1 bl) (RUN1 b1)))

«+ and the reading where no boys ran, that is,

(EVERY bl : (BOY1 b1) (NOT (RUN1 b1)))

« These two readings are captured by the single logical form

(<NOT RUN1> <EVERY b1 BOY1>)

+ where unscoped unary operators (for example, NOT, PAST, PRES, and so on) are wrapped around the predicate.

* **Proper names** must be interpreted in context, and the name John will refer to different people in different situations.

* Use a discourse variable that has the property of having the specified name.

* a special function, namely **(NAME <variable> <name>)**

* Produces the appropriate object with the name in the current context.

* Thus, the logical form of "John ran" would be (<PAST RUN1> (NAME j1 "John")).

+ Similar for pronouns and other indexical words, such as "here" and "yesterday",

Treat them using a special function of the form (PRO <variable> <proposition>).

* For example, the quasi-logical form for "Every man liked him" would be

(<PAST LIKE1> <EVERY **m1** MAN1> (PRO **m2** (HE1 **m2**)))

* HE1 is the sense for "he" and "him", and formally is a predicate true of objects that satisfy the restrictions on any antecedent, that is, being animate-male in this case.

* As with generalized quantifiers, when the restriction is a simple unary predicate, the pro forms are often abbreviated.

* For example, the logical form for "he" will often be written as (PRO m2 HE1).

* Similar for pronouns and other indexical words, such as "here" and "yesterday",

Treat them using a special function of the form (PRO <variable> <proposition>).

* For example, the quasi-logical form for "Every man liked him" would be (<PAST LIKE1> <EVERY m1 MAN1> (PRO m2 (HE1 m2))) b

Encoding Ambiguity in the Logical Form

- Because ambiguity often cannot be resolved at the logical form level, it must be possible to represent ambiguities in the logical form language.
- *Sue watched the ball* becomes

  (THE **b1** : ({BALL-DANCE BALL-SPHERE} **b1**) (PAST (WATCH1 SUE1 **b1**)))

Encoding Ambiguity in the Logical Form 2

- *Every child loves a pet* becomes

  (LOVES <EVERY **c1** (CHILD1 **c1**)> <A **p1** (PET1 **p1**)>)

- The quantifiers look like terms. This represents the two possibilities:

  (EVERY **c1** : (CHILD1 **c1**) (A **p1** : (PET1 **p1**) (LOVES1 **c1** **p1**)))
  (A **p1** : (PET1 **p1**) (EVERY **c1** : (CHILD1 **c1**) (LOVES1 **c1** **p1**)))

Abbreviation: <EVERY **c1** CHILD1>= (EVERY **c1** : (CHILD1 **c1**))

[5] Verbs and States in Logical Form:

« Consider the following sentences, all using the verb "break":

« John broke the window with the hammer.

+ The hammer broke the window.

+ The window broke.

+ All these sentences describe the same type of event but in varying detail.

« The verb **"break"** should be mapped to the same sense in each case.

« But there is a problem, as these three uses of the verb seem to indicate verb senses of differing arity.
« The first seems to be a ternary relation between John, the window, and the hammer,

« The second a binary relation between the hammer and the window, and

« The third a unary relation involving the window.

* Three different senses of break,

1. (<PAST BREAK1> (NAME **j1** "John") <THE **wl** WINDOW1> <THE **h1** HAMMER1>),

2. (<PAST BREAK2> <THE **h1** HAMMER1> <THE **w1** WINDOW1>), and

3. (<PAST BREAK3> <THE **wl** WINDOW1>)

+ Introducing events into the ontology, and treating the meaning of a sentence like "John broke it" :
(jel: (BREAK e1 (NAME j1 "John") (PRO i1 IT1)))

+ which asserts that el is an event of John breaking the indicated window.
* Now the meaning of "John broke it with the hammer" would be
+ (j**el**: (& (BREAK **el** (NAME **j1** "John") (PRO **i1** IT1))
(INSTR **e1** <THE h1 HAMMER>)))

« The logical form of "Mary sees John" will sometimes be written as

+ (PRES (SEES1 I1 [AGENT (NAME j1 "Mary")]

[THEME (NAME m1 "John")]))

* which is equivalent to

+ (PRES (j 1 (& (SEES1 11) (AGENT 11 (NAME j1 "Mary"))
(THEME I1 (NAME m1 "John")))))

« Other times it will be written in predicate argument form:

(PRES (SEES1 (NAME j1 "Mary") (NAME m1 "John")))

+ There is a limited set of abstract semantic relationships that can hold between a verb and its arguments.

+ These are often called thematic roles or case roles, and while different researchers have used different sets of roles, the number required has always remained small.

+ The intuition is that "John", "the hammer", and "the window" play the same semantic roles in each of these sentences.

+ "John" is the actor (the agent role), "the window" is the object (the theme role), and "the hammer" is the instrument (the instrument role) used in the act of breaking.

+ This suggests a representation of sentence meaning where everything is expressed in terms of unary and binary relations.

+ Using the three thematic roles just mentioned, the meaning of "John broke the window" would be

+ (je (& (BREAK e) (AGENT e (NAME j1 "John")) (THEME e <THE wl WINDOW>)))

* The logical form of "Mary sees John" will sometimes be written as

+ (PRES (SEES1 I1 [AGENT (NAME j1 "Mary")]

[THEME (NAME m1 "John")]))

« which is equivalent to

+ (PRES (j 1 (& (SEES1 **11**) (AGENT I1 (NAME **j1** "Mary"))
(THEME **I1** (NAME **m1** "John")))))

« Other times it will be written in predicate argument form:

(PRES (SEES1 (NAME **j1** "Mary") (NAME **m1** "John")))

* The abbreviated form for an assertion of the form
+ (je: (& (Event-p e) (Relation e obj1) ... (Relationn e Objn))) will be
(Event-p e [Relation obj1] ... [Relation objn))

«In particular, the quasi-logical form for the sentence "John broke the window" using this abbreviation is

«+ (<PAST BREAK!> ei [AGENT (NAME j1 "John")] [THEME <THE wl WINDOW1>])

+ It turns out that similar arguments can be made for verbs other than event verbs.

+ Consider the sentence "Mary was unhappy". If it is represented using a unary predicate as

+ (<PAST UNHAPPY> (NAME j1 "Mary"))

Verbs and States in Logical Form 4

- Because the pattern in 1 and 2 above is common, it is usual to leave out the "(exists **e1** :" and the "& (" and write 2', for example, as

  2''. (BREAK **e1** [AGENT (NAME **j1** "John"]
  [THEME (PRO **i1** IT1)]
  [INSTR <THE **h1** HAMMER>])

  Strictly the BREAK should be <PAST BREAK>.

Verbs and States in Logical Form 5

- Similarly, *Mary was unhappy* could initially be

  (<PAST UNHAPPY> (NAME **m1** "Mary"))

  but how do we handle modifiers like *in the meeting*?

- **Answer:** Introduce states s and use

  (<PAST UNHAPPY> s [EXPERIENCER (NAME m1 "Mary")]
  [AT-LOC <THE m2 MEETING>])

- Allen tends to switch between various notations as convenient. In particular, the angle brackets < > vs round parentheses ( ) seem to be a bit random

Conclusion:

+ This chapter presented a context-independent semantic representation called the logical form.

+ Such a representation is desirable to simplify the process of computing semantic structures from the syntactic structure, and more generally to modularize the processing of sentences by separating out contextual effects.

* The logical form language can encode many common forms of ambiguity in an efficient manner by allowing alternative senses to be listed wherever a single sense is allowed.

* An important aspect of the logical form language is its use of event and state variables on predicates.

* This allows you to represent additional adverbial modifiers without having to introduce different predicates for each possible combination of arguments to the verb.

Facets of Meaning:

Relationships between objects, entities, etc.

« Referents for phrases like you, him, the red gate - i.e. objects that correspond to the words in the phrases;

* Common sense inferences from basic facts conveyed.

* Speaker's intent: the door is open could be a bare statement, a complaint, a tacit request to close it.

[6] THEMATIC ROLES:

The notion of thematic roles, or cases. One motivating example from the last section included the sentences.

John broke the window with the hammer.
The hammer broke the window.
The window broke.

"John", " the hammer", and **"the window"** play the same **semantic roles** in each of these sentences. **"John"** is the **actor, "the window"** is the **object**, and **"the hammer"** is the instrument used in the act of **breaking of the window.**

We introduced **relations** such as **AGENT**, **THEME**, and **INSTR** to capture these intuitions. But can we define these relations more precisely, and what **other thematic roles** have proved useful in **natural language systems**? These issues are explored in this section.

Perhaps the **easiest thematic role** to define is the **AGENT role.** A **noun phrase** fills the **AGENT role** if it describes the instigator of the action described by the **sentence.**

One test for **AGENT**-hood involves **adding phrases like "intentionally**" or "in order to" to active voice sentences. If the resulting sentence is well formed, the **subject NP can fill the AGENT role**. The following sentences are acceptable:

John intentionally broke the window.
John broke the window in order to let in some air.

But these **sentences are not acceptable:**

*** The hammer intentionally broke the window.**
*** The window broke in order to let in some air.**

Thus the **NP "John"** fills the AGENT role only in the first **two sentences.**

Not all animate NPs, even in the subject **position,** fill the **AGENT role**. For instance, you **cannot normally** say

*** John intentionally died.**
*** Mary remembered her birthday in order to get some presents.**

NPs that describe something undergoing some change or being acted upon will fill a role called **THEME.**

Thus in **"The clouds appeared over the horizon",** the NP **"the clouds"** fills the THEME role. More **examples** follow, with the **THEME NP in italics:**

The rock broke.
John broke the rock.
I gave John the book.

A **range of roles** has to do with **locations,** or abstract locations. **First,** we must make the distinction mentioned earlier between relations that indicate a **location or place** and those that indicate **motion or paths.** The **AT-LOC** relation indicates where an object is or where an event takes place, as in

Harry walked on the road.
The chair is by the door.

On the road describes where the walking took place, while by the door describes where the chair is located.

Other phrases describe **changes in location**, **direction of motion**, or paths:

I walked from here to school yesterday.
It fell to the ground.
The birds flew from the lake along the river gorge.

There are at least **three different** types of phrases here: those that describe where something came from (**the FROM-LOC role),** such as "from here"; those that describe the **destination** (the TO-LOC role), such as "to the ground"; and those that describe the **trajectory or path** (the PATH-LOC role), such as "along the gorge".

These **location roles** can be generalized into roles over **arbitrary state values**, called the **AT role**, and roles for arbitrary state change (**the FROM, TO, and PATH roles**). Thus AT-LOC is a specialization of the AT role, and so on. You can see other specializations of these roles when you consider the abstract relation of possession:

I threw the ball to John.	(the TO-LOC role)
I gave a book to John.	(the TO-POSS role)
I caught the ball from John.	(the FROM-LOC role)
I borrowed a book from John.	(the FROM-POSS role)
The box contains a ball.	(the AT LOC role)
John owns a book.	(the AT POSS role)

Similarly, you might define **AT-TIME, TO-TIME**, and **FROM-TIME roles**, as in

I saw the car at 3 o'clock.	(the AT-TIME role)
I worked from one until three.	(the FROM-TIME and TO-TIME role)

The **roles apply** to general state change as well, as with **temperature** in

The temperature **remains** at zero. (AT VALUE)
The temperature **rose** from zero. (FROM-VALUE)

Another role is motivated by the **problem** that, given the **present taxonomy,** you cannot easily classify the role of the **NP in a sentence** such as

John believed that it was raining.

The **THEME role** is filled with the clause "that it was raining", since this is what is believed. "John" cannot be an AGENT because there is no intentionality in believing something. Thus you must introduce a **new role,** called **EXPERIENCER,** which is filled by **animate objects** that are in a described **psychological state**, or that undergo some **psychological process**, such as **perception**, as in the preceding sentence and as in

John saw the unicorn.

Another role is the **BENEFICIARY role,** which is filled by the **animate person** for whom a certain event is performed, as in

I rolled on the floor for Lucy.
Find me the papers!
I gave the book to Jack for Susan.

The last **example demonstrates** the need to distinguish the TO-POSS role (that is, to Jack) from the BENEFICIARY role.

The **INSTR role** describes a **tool, material**, or force used to perform some event, as in
Harry broke the glass with the telescope.
The telescope broke the glass.
I used some flour to make a cake.
I made a cake with some flour.

Natural forces are also included in the **INSTR category** here, although you could argue for a different analysis. Thus the following are also **examples of the INSTR role:**

The sun dried the apples.
Jack used the sun to dry the apples.

The **AGENT and INSTR roles** could be combined into a more general **role named CAUSAL-AGENT.**

Other roles need to be identified before **certain sentences** can be **analyzed**. For **example**, some sentences describe situations where **two people perform** an act together:

Henry lifted the piano with Jack.

To handle this, you must introduce a **role CO-AGENT** to account for the **PP with Jack.**

A more **complicated case** occurs in sentences involving exchanges or other **complex interactions.**

For **example,** consider the sentences

Jack paid $1 to the man for the book.
Jack bought the book from the man for $1.

These **sentences both** describe a situation where **Jack** gives the **man $1** and receives the book in exchange.

In the **first sentence,** however, the **$1 is the THEME** and there is no role to account for the book. In the **second sentence** the **situation is reversed**: the book is the **THEME and $1** is unaccounted for.
To **handle** these cases you must add a **role CO-THEME** for the **second object in an exchange.**

[7] SPEECH ACTS AND EMBEDDED SENTENCES:

speaker and the **propositional** content of the utterance.

. For now the **logical form language** is extended to **capture the distinctions.**

Each of the **major sentence** types has a corresponding **operator** that takes the **sentence interpretation** as an argument and produces what is called a **surface speech act.**

These indicate how the **proposition** described is intended to be used to **update the discourse situation**. They are indicated by **new operators** as follows:

ASSERT - the proposition is being **asserted**.
Y/N-QUERY - the proposition is being **queried.**
COMMAND - the proposition describes an **action to perform**.
WH-QUERY - the proposition describes an **object to be identified**.

For **declarative sentences,** such as **"The man ate a peach",** the complete LF is

(**ASSERT** (<PAST EAT> **e1** [**AGENT** <THE **m1** MAN1]
[THEME <A **p1** PEACH1>]))

For **yes/no** questions, such as **"Did the man eat a peach?",** the LF is

(Y/N-QUERY (<PAST EAT> **e1** [AGENT <THE **m1** MAN1>]
[THEME <A **p1** PEACH1>]))

For commands, such as ***"Eat the peach",*** the LF is

(COMMAND (EAT **e1** [THEME <THE **p1** PEACH1>]))

For **wh-questions**, such as **"What did the man eat?",** several additions need to be made to the logical form language.

First, you need a way to represent the meaning of **noun phrases** involving **wh-terms.**

A **new quantifier WH** is defined that indicates that the **term stands for an object** or objects under question.

Thus a **noun phrase** such as **"what"** would be represented **<WH o1 ANYTHING>, "which man"** as **<WH m1 MAN1>,** and **"who"** as **<WH p1 PERSON>. Finally,** for question forms such as **"how many" and "how much"**, we introduce the quantifiers **HOW-MANY and HOW-MUCH.**

Note that **wh-terms** are **scope sensitive** and thus treating them as **quantifiers** makes sense. The question **"Who is the leader of every group?"** is ambiguous between asking for a single person who leads every group, and asking for the leader of each of the groups.

Thus, the **logical form** of the sentence ***"What did the man eat?"*** is

(WH-QUERY (<PAST EAT> e1 (AGENT <THE m1 MAN1>]
[THEME <WH w1 PHYSOBJ>]))

Embedded sentences, such as **relative clauses**, end up as complex restrictions within the **noun phrase** construction and thus do not need any **new notation**. For example, the **logical form** of the sentence ***"The man who ate a peach left"*** would be

```
(ASSERT
 (<PAST LEAVE> l1
     [AGENT <THE m1 (& (MAN1 m1)
          (<PAST EAT1> e2
               [AGENT m1]
               [THEME <A p1 PEACH>]))>]))
```

[8] DEFINING SEMANTIC STRUCTURE: MODEL THEORY:

The term **semantics** has been used to **reflect** the representation of **meanings for sentences**.

This was **expressed** in the logical form language. There is another **meaning of semantics** used in **formal language theory** that provides a meaning of the **logical form language itself**.

It concentrates on **distinguishing** the different classes of semantic objects by exploring their model-**theoretic properties,** that is, by defining **semantic units** in terms of their **mapping to set theory.**

UTTERANCE → (ASSERT *PROPOSITION*) |
(Y/N-QUERY *PROPOSITION*) |
(COMMAND *PROPOSITION*) |
(WH-QUERY *PROPOSITION*)

PROPOSITION → (*n-ARY-OPERATOR* *PROPOSITION*$_1$ ... *PROPOSITION*$_n$) |
(*QUANTIFIER VARIABLE : PROPOSITION PROPOSITION*) |
(*n-ARY-PREDICATE TERM*$_1$ ... *TERM*$_n$) |
(*EVENT-STATE-PRED VARIABLE* [*ROLE-NAME TERM*]$_1$ ... [*ROLE-NAME TERM*]$_n$)

TERM → *VARIABLE* |
(NAME *VARIABLE NAME-STRING*) |
(PRO *VARIABLE PROPOSITION*)

1-ARY-OPERATOR → NOT | PAST | PERF | PROG | ...

2-ARY-OPERATOR → AND | BUT | IF-THEN | ...

QUANTIFIER → THE | SOME | WH | ∃ | ..

VARIABLE → **b1** | **man3** | ...

1-ARY-PREDICATE → *TYPE-PREDICATE* | HAPPY1 | RED1 | ...

TYPE-PREDICATE → *EVENT-STATE-PRED* | (PLUR *TYPE-PREDICATE*) | MAN1 | ...

EVENT-STATE-PRED → RUN1 | LOVE3 | GIVE1 | HAPPY | ...

2-ARY-PREDICATE → *ROLE-NAME* | ABOVE1 | ...

ROLE-NAME → AGENT | THEME | AT-LOC | INSTR | ...

NAME-STRING → "John" | "The New York Times" | ...

Figure 8.7 A formal definition of the syntax of the logical form language

TERM → <*QUANTIFIER VARIABLE PROPOSITION*>

TERM → <*n-ARY-OPERATOR TERM*$_1$ ... *TERM*$_n$>

n-ARY-PREDICATE
→ <*m-ARY-OPERATOR n-ARY-PREDICATE*$_1$ ... *n-ARY-PREDICATE*$_m$>

n-ARY-OPERATOR → {*n-ARY-OPERATOR*$_1$ ... *n-ARY-OPERATOR*$_m$}

QUANTIFIER → {*QUANTIFIER*$_1$ ... *QUANTIFIER*$_m$}

n-ARY-PREDICATE → {*n-ARY-PREDICATE*$_1$ ... *n-ARY-PREDICATE*$_m$}

TYPE-PREDICATE → {*TYPE-PREDICATE*$_1$ ... *TYPE-PREDICATE*$_m$}

EVENT-STATE-PRED → {*EVENT-STATE-PRED*$_1$ ... *EVENT-STATE-PRED*$_m$}

ROLE-NAME → {*ROLE-NAME*$_1$ ... *ROLE-NAME*$_m$}

Figure 8.8 Additional rules defining the quasi-logical form

Figure 8.8 Additional rules defining the quasi-logical form

The rest of the book. If you are not familiar with FOPC and its **model-theoretic semantics**, you should read Appendix B before reading this section.

The basic building block for defining **semantic properties is the idea** of a model, which informally can be thought of as a **set of objects and their properties and relationships**, together with a specification of how the language being studied relates to **those objects and relationships.**

A **model** can be thought of as representing a particular context in which a **sentence is to be evaluated.**

Model theory is an excellent method for studying **context-independent meaning**, because the meanings of sentences are not defined with respect to one **specific model** but rather by how they relate to any possible model.

In **other words**, the meaning of a **sentence** is defined in **terms of the properties** it has with respect to an **arbitrary model.**

Formally, a model m is a tuple $\langle D_m, I_m \rangle$, where D_m is the **domain of interpretation** (that is, a set of primitive objects), and **I** is the **interpretation function**.

To **handle natural language**, the domain of interpretation would have to **allow objects** of all the **different types** of things that can be referred to, including **physical objects, times, Locations, events, and situations.**

Senses of noun phrases - refer to specific objects; the interpretation function maps each to an element of D_m.

Senses of singular common nouns (such as "*dog", "idea", "party")* - **identify classes of objects** in the domain; the **interpretation function maps** them to sets of elements from **Dm** (that is, subsets of **Dm**).

Senses of verbs - identify sets of **n-ary relations** between objects in **D**. The arity depends on the verb. For instance, the exercising sense of "run", RUN1, might map to a set of unary relations (<**X**>, where **X** runs), and the usual sense of loves, LOVES 1, to a set of binary relations (<**X**, **Y**>, where **X** loves **Y**), and the usual sense of "put", PUT1, to a set of ternary relations (<**X**, **Y**, **L**>, where **X** puts **Y** in location **L**).

We can now define a notion of **truth** of a proposition in the logical form language, again relative to an arbitrary model **m**.

A proposition of the form $(V_n\ a_1 \ldots a_n)$ is true with **respect to a model** in if and only if the tuple consisting of the **interpretations** of the **ai's** is in the set that is the **interpretation of Vn**. i.e., the

tuple $\langle I_m(a_1),\ldots, I_m(a_n)\rangle$ is in the set $I_m(V_n)$.

Following conventional notation, we will write the fact that a proposition P is true with respect to a model **m** by **m|= P.**

This is sometimes also read as **"m** supports P".
For **example, m** supports (RUN1 JACK1) only if **Im(JACK1)** is in the set **Im(RUN1)**, and **m** supports **(LOVES1 JACK1 SUE1)** only if **< Im(JACK1), Im(SUE1)>** is in the set **Im (LOVES1)**.

The semantics of the **quantifiers in FOPC** is fairly simple.

For **example,** the proposition **Žx .** ***P(x)*** is true with respect to a model **m** if and only if **P(x)** is true for any value of **x** in **Dm**.

The proposition **jx . P(x)**, on the other hand, is true with respect to a model **m** if and only if **P(x)** is true for at least one value of **x** in **Dm**.

For **natural languages** we have generalized quantifiers. The **truth conditions** for each quantifier specify the required relationship between the objects satisfying the **two propositions**.
For **example,** consider the proposition **(MOST1 x (P x) (Q x))**.

Semantic Relations Among Sentences:

With a **semantic theory** in hand, it is now possible to be more precise about certain **inferential relationships among sentences.**

For instance, when you know that some **sentence S is true**, then some **other sentences** must also be **true.**

For instance, if you know **"A red box is on the table"** then you also know that **"A box is on the table"**. This **relationship** between sentences is called **entailment,** and we say the sentence **"A red box is on the table"** entails the sentence **"A box is on the table".** Formally, entailment can be defined in terms of the models that support the sentences. In particular, sentence S entails sentence S' if and only if every model that supports S also supports S'; that is,

S entails S' if and only if for any model m, if m |= S then m |= S'

Conversely, if there are **no models** that support **both sentences simultaneously**, the sentences are said to **contradict** each other; that is, there is no **possible situation** in which **both statements** can be **true. Slightly modifying** the previous example, we know that the sentences **"A red box is on the table"** and **"There is no box on the table"** are **contradictory** because there is no model that can support these **two sentences simultaneously.**