

# Montague Grammar

Amy H. Kao  
EECS 595  
Fall 2004

*“There is in my opinion no important theoretical difference between natural languages and the artificial languages of logicians; indeed, I consider it possible to comprehend the syntax and semantics of both kinds of languages within a single natural and mathematically precise theory” (Montague 1970)*

## Abstract

In 1970, Richard Montague's radical theories of how formal logic could be used to describe natural language changed the way logicians and linguists perceived the connections between their fields. Using formal logic and a model-theoretic view, Montague creates a system where the syntactic structure and semantic structure of natural language are connected in a manner that allows for a better understanding of the semantic meanings of sentences. These theories are exemplified in what has been termed **Montague Grammar**. In this paper, we gain a better understanding of Montague Grammar's impact on logicians and linguists by understanding the many components of Montague Grammar, including syntactic and semantic handling, intensional logic, general quantification, and model theory.

## History

Richard Montague (1930 - 1971), both a logician and philosopher of language, was a student of Alfred Tarski, a logician often compared with the likes of Aristotle and Gödel. In 1933, Tarski's paper **The concept of truth in formalized language** began the development of model theory, which provided the foundation for much of Montague's work in semantics. Model theory describes the meanings of formal and natural languages by defining the classes of objects referred to by the expressions. In this manner, model theory examines the truth of an expression (using Tarski's Truth definitions) by examining the truth of the expression's corresponding class elements. Tarski's influence may be seen in much of Montague's approaches semantics, allowing for a logical model theoretic approach that was radical for Montague's time.

In **Universal Grammar** (1970), Montague introduced his theory of formal syntax and semantics as applied to both formal and natural language. **Universal Grammar** is significant because it was the first attempt at applying formal semantics to natural language. Logicians prior to Montague regarded natural language as too ambiguous and unstructured for formal logical analysis while linguists felt that formal languages were unable to capture the structures of natural languages. Montague more explicitly argues for the similarities of natural and formal language in **English as a formal language** (1970) where he writes "I reject the contention that an important theoretical difference exists between formal and natural languages."

Montague demonstrates the application of his theories in **The Proper Treatment of Quantification in Ordinary English** (1973), most commonly referred to as PTQ, where he defines the syntax and semantics for a large fragment of English. In the text, English phrases are translated into logical expressions based on intensional logic, which are then interpreted with Tarski's model theory. The term **Montague grammar** generally refers to the theories outlined in **Universal Grammar**, **English as a formal language**, and **PTQ**, but because so much of Montague's work is explained in **PTQ**, Montague grammar is often also referred to as PTQ.

Montague's work came at a time when linguists were developing different approaches to semantics in relation to Chomsky's generative grammar. The two main approaches, generative semantics (Lakoff, Ross, McCawley, Postal) and interpretive semantics (Jackendoff, Chomsky), were at odds with each other. Generative semantics did not separate semantic and syntactic rules while interpretive semantics contained a distinction between the two types of rules. Montague's approach offered a perspective of how the structures of semantic and syntactic rules can be connected to each other but not required to be the same rules. Linguists hoped that Montague's work would merge "the best aspects of both of the warring approaches, with some added advantages of its own" (Partee 2001).

Montague was not the only linguist aware of the relevant advances in logic and philosophy of language but before Montague's work became popular, linguist and logicians did not actively work together. Many attribute the lack of communication between the fields to the personalities of generative semanticists. The clash of these two fields was demonstrated in August of 1969 during a colloquium of logicians and linguists. The colloquium was aimed at fostering collaborations between the two fields but was deemed a failure by W.V. Quine, who declared it "a fiasco at bridge building" (Abbott, 1999). Though unsuccessful in generating new interactions between people, the papers from the colloquium, including articles by Montague, were read by many linguists and logicians. These articles were key in setting the grounds for future collaborations between the two fields.

One of the important issues brought to the attention of linguists at the 1969 colloquium was the differentiation between semantic representation and semantic meaning. Philosopher David Lewis condemned the existing systems of semantic representation because they lacked the treatment of truth conditions. At the time, expressions were commonly given a corresponding marking to denote the semantic meaning. Lewis claimed that these markings simply translated sentences into an artificial language one could call **Semantic Markers** without regard to the meaning of the sentence. Montague's work in the years following that colloquium could be deemed an answer to Lewis's criticisms with semantic analysis. His model theoretic analysis of semantics in natural language provided a treatment of meanings that Lewis had found lacking in semantic representations.

Montague's work in natural language is described in only three publications (**Universal Grammar**, **English as a formal language**, and **The Proper Treatment of Quantification in Ordinary English**) but most found his writing to be "highly formal and condensed, very difficult for ordinary humans (even logicians!) to read with comprehension." (Abbott 1999) The proliferation of Montague's work is therefore often attributed to Barbara Partee who presented Montague's work in a more understandable fashion.

Partee graduated MIT (1965) with a PhD in Linguistics under Chomsky and became familiar with Montague's work when she began teaching at UCLA (1965-1972) where he was a professor. Partee's **Montague Grammar and Transformational Grammar**

(Linguistic Inquiry 1975) was seen as the first introductory text on Montague Grammar. Dowty, Wall & Peters later published another comprehensive text called “Word Meaning and Montague Grammar” (1981). When Partee became a faculty member at the University of Massachusetts at Amherst, her classes on Montague Grammar became an attraction to many prominent linguists and logicians.

Montague’s groundbreaking work in formal semantics created new areas of research that were of interest to both linguists and logicians, helping to proliferate interactions between the fields. Barbara Partee’s joint appointment in the linguistics and philosophy departments at University of Massachusetts is one of many signs that linguists and logicians are collaborating more closely. In 1977, the journal **Linguistics and Philosophy** published its first issue and to this day, continues to foster communications between the two communities.

## **Montague Grammar**

Montague’s **Universal Grammar** (UG) is a general theory of language developed to encompass the syntax and semantics of known artificial languages as well as natural languages and “unnatural” languages. UG relates syntax and semantics by creating a formal interpretation of Frege’s philosophy that an expression’s meaning is a function of the meaning of its constituents and its syntax. The characteristics of UG are meant to be more general so as to serve as the reference framework for comparing formal and natural languages.

The theories of **Universal Grammar** are applied in Montague Grammar as described in **The Proper Treatment of Quantification in Ordinary English** (PTQ). It is worth noting that the terms Universal Grammar (UG), Montague Grammar, and PTQ may be found interchangeably in various texts regarding Montague’s work. In general, UG refers to Montague’s theories of syntax and semantics found in **Universal Grammar** whereas PTQ is commonly used to denote the application and process of the theory. The term Montague Grammar is more general and is commonly used to refer to anything within Montague’s three main texts concerning syntax and semantics. The following sections of this paper will examine syntax and semantics as handled in Montague Grammar.

### **Syntax**

Syntax in Montague grammar consists of syntactic rules and syntactic operations as described in **Universal Grammar**. The rules are composed of syntactic categories defined as basic or recursive clauses while syntactic operations are functions such as concatenation that define how categories form new phrases. These rules and operations may be applied to a lexicon to define the syntax of a ‘fragment’ of English.

Syntactic categories are based upon two primitive categories: sentence and entity categories as denoted by *t* and *e* respectively. The *t* category does not contain lexical items. It is composed of sentences built by recursive rules. The label *t* represents the fact that all members of the category contain a truth-value. The primitive category *e* does not

actually contain any entities. Its usage lies not in categorization of phrases but in representing semantic information. For example, phrases categorized as *t/e* implies the usage of a function that takes the senses of entities (*e*) into truth-values (*t*).

Using the primitive categories *t* and *e*, an indefinite number of categories may be constructed in the form *X/Y* where *X* and *Y* are categories and *Y* is a phrase that may be used to create an *X* phrase. To represent the fact that more than syntactic category may be represented by the same category type, the syntactic categories may be further divided by added more slashes. For example, two distinct categories that use an *e* phrase to create a *t* phrase may be represented by *t/e* and *t//e*. There is no limit to the number of slashes allowed but the original grammar defined in PTQ only used double slashes.

Category	Abbreviation	PTQ Name	Nearest linguistic equivalent
<i>t</i>	(primitive)	Truth-value expression; or declarative sentence	Sentence
<i>e</i>	(primitive)	Entity expression; or individual expression	(noun phrase)
<i>t/e</i>	IV	Intransitive verb phrase	transitive verb, transitive verb and its object, or other verb phrases
<i>t/IV</i>	T	Term	Noun phrase
<i>IV/T</i>	TV	Transitive verb phrase	Transitive verb
<i>IV/IV</i>	IAV	IV-modifying adverb	VP-adverb and prepositional phrases containing in and about.
<i>t//e</i>	CN	Common noun phrase	Noun or NOM
<i>t/t</i>	None	Sentence-modifying adverb	Sentence-modifying adverb
<i>IAV/T</i>	None	IAV-making preposition	Locative, etc., preposition
<i>IV/t</i>	None	Sentence-taking verb phrase	V which takes that-COMP
<i>IV/IV</i>	None	IV-taking verb phrase	V which takes infinitive COMP

**Table 1: Syntactic Categories (Partee 1976)**

For simplification, Montague created the abbreviations IV, T, TV, IAV, and CN for the first five derived categories. Without abbreviations, category labels would grow increasingly hard to read. For example, TV would need to be denoted as *(t/e)/(t/(t/e))*. These abbreviated categories contain both lexical and derived phrases while the last four unabbreviated categories (*t/t*, *IAV/T*, *IV/t*, and *IV/IV*) contain only lexical members. For large lexicons, it may be necessary to define further syntactic categories.

## Rules

Every non-primitive syntactic category in Montague grammar contains lexical rules and recursive rules. Lexical rules simply state the category of a lexical phrase. To generate a new grammar, one first categorizes the lexical terms in terms of syntactic categories. Each resulting set of lexical members is called a ‘basic expression’ of the category. Below is an example ‘fragment’ of English from Partee 1973 where  $B_A$  represents ‘basic expression of category A’. The term ‘fragment’ refers to a subset of a language. When working with Montague grammar, one works with ‘fragments’ because the grammar was not intended to define syntax and semantics a whole language. Note that the example does not contain common categories such as determiners. This is because our example is only working with a very small ‘fragment’ of English. Fragments that require determiners would have a category for them. The ability to define new syntactic categories allows for extending the category set to accommodate larger and more complex ‘fragments’ of English.

- 
- 1)  $B_{IV} = \{\text{run, walk, talk, rise, change}\}$
  - 2)  $B_T = \{\text{John, Mary, Bill, ninety, he}_0, \text{he}_1, \text{he}_2, \dots\}$
  - 3)  $B_{TV} = \{\text{find lose, eat, love, date, be, seek, conceive}\}$
  - 4)  $B_{IAV} = \{\text{rapidly, slowly, voluntarily, allegedly}\}$
  - 5)  $B_{CN} = \{\text{man, woman, park, fish, pen, unicorn, price, temperature}\}$
  - 6)  $B_{t/t} = \{\text{necessarily}\}$
  - 7)  $B_{IAV/T} = \{\text{in, about}\}$
  - 8)  $B_{IV/t} = \{\text{believe that, assert that}\}$
  - 9)  $B_{IV/IV} = \{\text{try to, wish to}\}$
- 

**Figure 1: Example Expressions for a Fragment**

Every complex syntactic category of form  $X/Y$  has a corresponding recursive syntactic rule. The rule defines a function  $F_i(x,y)$  that creates a phrase in  $X$  from a phrase in  $X/Y$  and a phrase in  $Y$ . Abbreviations  $IV$  and  $CN$  represent categories  $t/e$  and  $t/e$  respectively. The category  $e$  contains no phrases, so a function on  $IV$  or  $CN$  would have no effect on the phrases. Therefore,  $IV$  and  $CN$  do not follow the recursive rule shown in Equation 1.

---

$$\text{If } \alpha \in X/Y \text{ and } \beta \in Y \text{ then } F_i(\alpha, \beta) \in X$$

---

**Equation 1: Syntactic Rule**

Most recursive rules are simply concatenations such that  $F_i(\alpha, \beta) = \alpha \beta$  but they may be as complicated as required. The following example rule for  $TV$  (from Thomason p.81) defines a more complex rule that checks if  $\alpha$  is a  $TV$  or  $TV/T$  in order to determine the accurate position of  $\beta$ . The rule also transforms  $he_i$  to  $him_i$  if  $\beta$  is  $he_i$ . Note that the

subscript 3 from  $F_3(\alpha, \beta)$  refers to the expression number of its corresponding category,  $B_{TV}$ , from Figure 1.

---


$$F_3(\alpha, \beta) = \begin{array}{ll} \text{If first word of } \alpha \text{ is a TV:} & \\ \alpha \beta & \text{if } \beta \text{ is not a variable} \\ \alpha \text{ him}_i & \text{if } \beta \text{ is he}_i \end{array}$$
  

$$\begin{array}{ll} \text{If } \alpha \text{ is } \alpha_1 \alpha_2 \text{ where } \alpha_1 \text{ is a TV/T:} & \\ \alpha_1 \beta \alpha_2 & \text{if } \beta \text{ is not a variable} \\ \alpha \text{ him}_i \alpha_2 & \text{if } \beta \text{ is he}_i \end{array}$$


---

**Figure 2: Example Syntactic Rule**

Examples:

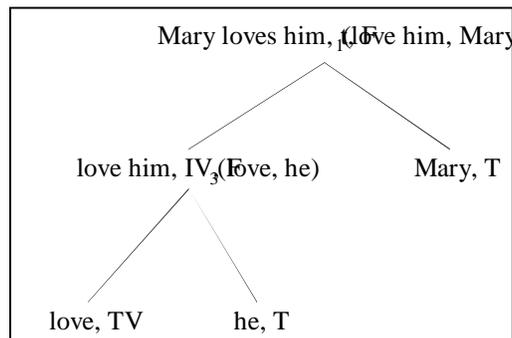
$F_3(\text{shave, a fish}) = \text{shave a fish}$

$F_3(\text{seek, he}_1) = \text{seek him}_1$

$F_3(\text{read a large book, Mary}) = \text{read Mary a large book}$

**Figure 3: Examples of Rule  $F_3$**

Syntactic recursive and lexical rules may be applied to generate basic expressions and sentences. A tree created from the rules display how the syntactic rules generate a sentence. Such a tree, known as an analysis trees, illustrates the syntactic structure of a sentence. The following analysis tree uses the lexicon from Figure 1 to generate the phrase **Mary loves him** from the lexical terms **love he Mary**. Note that the example function  $F_3$  is used to generate **love him** from **love** and **he**.



**Figure 4: Analysis Tree for "Mary loves him"**

## Semantics

In PTQ, semantic interpretations are derived from the formal intentional logic translation of sentences. The process begins by translating sentences into a formal intentional logic that specifies a one-to-one correlation between syntactic and semantic rules. The formulas generated may then be analyzed to determine the sentence's truth with respect to a given model.

### Intensional Logic and Types

An intension is a function that generates extensions from a set of inputs. Extensions refer to things in one of the following three categories: the truth-value of a sentence, the thing named by a name, or the set of objects a common noun applies to. Inputs, known as an index, to an intension consist of variables about the context of the expression.

Extension Category	Appropriate Intension Function
Truth-value of a sentence	From indices to truth-values
Thing named by a name	From indices to things
Set of objects a common noun or intransitive verb phrase applies to	From indices to sets

Table 2: Extension Categories and Intension Functions

To understand extensions more thoroughly, we first examine each extension category. The first category defines extensions of names where an extension simply refers to the object being named. In this category, the extension of the name **John** would be the individual named **John**. In the next category, an extension refers to the set of objects represented by a common noun phrase or intransitive verb phrase. For example, the extension of common noun **student** is the set of all students whereas the extension of the intransitive verb phrase **reads quickly** is the set of all individuals who read quickly.

The last category makes use of extensions of names and common nouns to generate extensions of sentences as a truth-value. The truth-value of a sentence refers to whether or not it is true in a particular possible world or worlds identified. In this context, the truth-value or extension is based upon the extensions within the sentence. A simple example would be **John reads quickly** where the sentence's extension is based upon the extensions of **John** and **reads quickly**. This concept of the meaning of a sentence as a function of the meanings of its constituents is based upon Frege's functionality principle.

One of the problems with extensions is that alone, they are not able to accurately analyze sentences with intensional contexts. An example is the sentence **Necessarily the University of Michigan is identical with the University of Michigan** where "necessarily" is understood to mean "in all possible worlds". The extension of this sentence is its truth-value, which is true. Replacing the second occurrence of **University of Michigan** with **Columbia University** would replace a constituent with a new

constituent containing the same extension but the resulting new sentence would no longer contain the same extension. Its truth-value would not be true because there exists a possible world in which the University of Michigan and Columbia University are not identical.

A more complicated example of the failings of extensions is the phrase **classmate of John** where **classmate** is defined as a student at the same institution and level of study. If John is currently a graduate student at the University of Michigan, then the phrase has the same extension as the phrase **graduate student at the University of Michigan**.

We now examine the compound phrase **former classmate of John** whose extension is the set of individuals who have at some point been at the same institution and level of study as John but are currently not a graduate student at the University of Michigan. If we replace **classmate of John** with the phrase **graduate student at the University of Michigan**, the resulting phrase is **former graduate student at the University of Michigan**. The extension of this new phrase is the set of individuals who were once a graduate student at the University of Michigan but currently are not. Though we replaced a phrase with a new phrase of the same extension, the resulting compound phrase does not contain the same extension as the original compound phrase.

Because of these problems with extension, Frege suggested that in oblique contexts, an expression's extension is the intension of the expression. The logic for determining intensions (or the sense) of an expression was partially developed by Church (1951) and based on Carnap's suggestion that an intension is a function of the expression and its possible state of affairs. Montague developed these ideas into a new form of intensional logic (IL) in which the possible state of affairs was represented by the context of the expression, defined as an index. The index consists of facts about the world, the time of utterance, place of utterance, the surrounding discourse, as well as other relevant variables.

We now examine how intensional logic applies to extensions. As previously specified, an expression's extension may be defined as the expression's intension, which is a function of that expression and the indexes of the expression. The symbol  $\wedge$  is prefixed to an expression to indicate that the expression's extension is its intension. For example, if  $\wedge j$  represents the individual **John**, then  $\wedge j$  is a function that finds an individual named **John** at any given world and time.

In terms of phrases such as **classmate of John**, the intension depends on the extension of **classmate of John** as well as context information such as when this phrase was spoken. For the phrase **former classmate of John**, the intension is a function of the extension of the phrase as well as the time at which the phrase was spoken and the set of individuals who were classmates of John at all times previous to the time of the phrase being spoken.

Montague merges intensional logic and Church's theory of types to create his version of intensional logic known as IL. In IL, each rule is a type and every syntactic category in Montague Grammar has a corresponding type in intensional logic. The rule for

translating a syntactic category into IL simply translates each  $F_i(\alpha, \beta)$  from the syntactic rule of Equation 1 into  $\alpha'(\wedge\beta')$ . The variable  $\wedge\beta'$  refers to the intension of  $\beta'$  and  $\alpha, \beta$  translates into  $\alpha'\beta'$ . Formally, this is stated as follows:

$$\begin{array}{l} \text{If } \alpha \in X/Y \text{ and } \beta \in Y \text{ and } \alpha, \beta \text{ translates into } \alpha'\beta', \\ \text{then } F_i(\alpha, \beta) \text{ translates into } \alpha'(\wedge\beta'). \end{array}$$

**Equation 2: Translation of Syntactic Rule to IL**

As in syntactic categories, IL's types are based upon the two primitive types  $t$  and  $e$ . Type  $t$ , sentence, contains rules that define the truth-value of expressions of type  $t$  with respect to a predefined model and the assignment of values to variables. The type  $e$  is composed of entities. All types are recursively generated from primitive types  $t$  and  $e$  using the rules from intensional logic (IL).

Though there are many rules that define IL, for the purpose of this paper, we will only focus on the rules that pertain to the recursive generation of types. Table 3 lists the rules for recursively defining new types. All types are considered with respect to the set of individuals (A), the set of worlds (W), and the set of times (T). These factors represent the indexes of the function.

#	Rule	Semantic Rule	Semantic Set
1	T	$D_t$ is $\{0,1\}$ .	all truth-values
2	E	$D_e$ is A.	all entities
3	If a and b are types, $\langle a, b \rangle$ is a type	$D_{\langle a, b \rangle}$ is $D_b^{D_a}$	all functions from $D_a$ to $D_b$ for $D_a$
4	If a is a type, $\langle s, a \rangle$ is a type	$D_{\langle s, a \rangle}$ is $S_a = S_a^{W \times T}$	all functions from world-time pairs to $D_a$

**Table 3: Semantic Rules**

The third rule in Table 3 defines  $\langle a, b \rangle$  as the representation of a type that is a function from  $a$  to  $b$ . This representation of functions is used in rule 4 to define types with respect to world-time pairs. The expression  $\langle s, a \rangle$  is used to represent a phrase of type  $a$  whose extension is the intension of the phrase. The variable  $s$  may be seen as representing the world-time pairs such that  $\langle s, s \rangle$  is a function from world time pairs to  $a$ .

The recursive part of rule 4 is illustrated when we define the extensions and intensions of lexical members of expressions from Figure 1. A function for an expression of syntactic category X/Y, the semantic function would be a function from the intensions of Y's to extensions of Xs. Semantically, the type X/Y would be represented as  $\langle \langle s, y \rangle, x \rangle$ .

Using the rules in IL, we see that the extension of a sentence would be represented by type  $\langle \langle s, t \rangle, t \rangle$ . The extension of  $t$  would be a function form worlds and times to truth-values. It follows that  $\langle \langle s, t \rangle, t \rangle$  is a function from worlds and times ( $s$ ) to the extension of  $t$ . In other words, the intension of  $\langle \langle s, t \rangle, t \rangle$  is a function from worlds and times to a function of worlds and times to truth-values.

A simple example of IL rules is demonstrated with the IV-phrase walk which is of type  $\langle\langle s, e \rangle, t \rangle$ . The phrase may be seen as a command, hence its classification as a sentence,  $t$ . The extension of the phrase is the set of walk concepts. Using rule 4 from Table 3, the intension of walk,  $\wedge$ walk, is denoted as  $\langle s, \langle\langle s, e \rangle, t \rangle \rangle$ , a function of worlds and times to the phrase's extension.

### Quantification and Compositionality

Intensional logic was used by Montague to address natural language's lack of strong compositionality in terms of noun phrases. Strong compositionality means that a phrase's meaning may be derived from the meaning of its constituents and their syntactic structure. Formal logics are strongly compositional because an expression's components always behave in the same manner and receive the same interpretation as their category dictates. Natural languages on the other hand may not always be interpreted in the same manner as their constituents may indicate.

Before Montague, noun phrases presented linguists with a compositionality problem due to the existence of quantifiers. The problem is exemplified in comparing the interpretations of **John talks** and **Every student talks**. The two sentences have similar syntactic structures and parallel meanings so it is desirable to analyze them in parallel. When the two sentences are translated into first order predicate logic, their translations, as seen in Table 4, are drastically different.

Sentence	First Order Predicate Logic	General Quantification
John talks	Talks(j)	John' ( $\wedge$ talk')
Every student talks	$\forall x$ [Student(x) $\rightarrow$ Talks(x)]	Every man' ( $\wedge$ talk')

Table 4: Sentences and their Translations

To address the compositionality problem, Montague introduced the notion of a generalized quantifier (GQ) in analyzing noun phrases. A generalized quantifier is an expression that denotes the set of subsets of a domain. This would mean that the phrase **John** could be seen as the set of properties (sets) containing the individual concept of **John**. In intensional logic, **John** is represented by the constant  $\mathbf{j}$  of category  $e$  and the intension or individual concept of **John** is  $\wedge \mathbf{j}$ . The general quantification of the phrase **John** is therefore  $\mathbf{P}'\mathbf{P}\{\wedge \mathbf{j}\}$ , the set of all properties of the individual concept of **John**. Similarly, **Every student** could be seen as the set of all supersets of the set of students.

$$\begin{aligned} & \text{If } \alpha \in P_{CN} \text{ then } F_0(\alpha), F_1(\alpha), F_2(\alpha) \in \\ & P_T, \text{ where} \\ & \quad F_0(\alpha) = \text{every } \alpha \\ & \quad F_1(\alpha) = \text{the } \alpha \\ & \quad F_2(\alpha) = \text{a/an } \alpha \end{aligned}$$

Equation 3: Syntactic Rule for quantifier Phrases

Syntactic rules in GQ treat noun phrases with quantifiers **every**, **the**, **a**, or **an** in the same manner. Note from Equation 3 and Equation 4 that phrases with quantifiers are all translated into functions of  $P_T$  regardless of the quantifier used.

---

If  $\alpha$  translates into  $\alpha'$ , then  
 $F_0(\alpha)$  translates into  $P'[(\forall x)(\alpha'(x) \rightarrow P\{x\})]$   
 $F_1(\alpha)$  translates into  $P'[(\exists y) ((\forall x) [\alpha'(x) \leftrightarrow x = y] \wedge P\{y\})]$   
 $F_2(\alpha)$  translates into  $P'[(\exists x)(\alpha'(x) \wedge P\{x\})]$

---

**Equation 4: Translation rule for quantifier phrases**

With the GQ interpretation, **John talks** and **Every student talks** may be represented in similar fashion as seen in Table 4. Note in Table 4 that with GQ, the predicate is no longer **talks**. This is due to the GQ rule that a term-phrase is used as the function when a term-phrase and IV-phrase are combined to form a sentence. Because the phrase **talks** is an IV-phrase, the term-phrases **John** and **Every man** are used as functions. We therefore see **John'(^run')** which states that the individual concept of John contains a set of properties where one of the properties is running. Similarly, **every man'(^run')** means that every man contains a set of properties where one of the properties is running.

One of the limitations of Donkey sentences were unable to be properly analyzed using Montague's generalized quantifiers (GQ). The famous donkey sentence is **Every farmer who owns a donkey beats it**. The first problem presented in the translation to predicate logic occurs with respect to the pronoun **it**. If **a donkey** is represented with an existential quantifier,  $beats(X,Y)$  is not in the scope of variable  $Y$ , meaning the phrase **it** will be outside the scope of the quantifier and not properly bound. **Equation 5** illustrates this problem by underlining the portion of the equation scoped by  $Y$ .

---

Every farmer who owns a donkey beats it.

---

**Figure 5: Donkey Sentence**

If the scope of  $Y$  were extended to the right to encompass  $beats(X,Y)$ ,  $farmer(X)$  is removed from the implication as shown in

**Equation 6.** This interpretation would state that for every  $X$ ,  $X$  is a farmer and there exists some  $Y$  such that if  $Y$  is a donkey and  $X$  owns  $Y$ ,  $X$  beats  $Y$ . To account for the farmer, **Equation 7** extends the scope to encompass  $farmer(X)$  but the result is still incorrect. This third attempt states that for every  $X$ , there is definitely a  $Y$ , and if  $X$  is a farmer and  $Y$  is a donkey, and  $X$  owns  $Y$ , then  $X$  beats  $Y$ . In simpler terms, this would state that every farmer owns a donkey that he beats. **Equation 8** moves  $Y$  out even further to scope the whole statement but this causes further error in the result. Moving  $Y$  out creates the statement that there exists some  $Y$  so that for all  $X$ ,  $X$  is a farmer,  $y$  is a donkey owned by  $X$ , and  $X$  beats  $Y$ .

**Equation 5:**

$$\forall X (\text{farmer}(X) \wedge \exists Y (\underline{\text{donkey}(Y) \wedge \text{owns}(X, Y)}) \rightarrow \text{beats}(X, Y))$$

**Equation 6:**

$$\forall X (\text{farmer}(X) \wedge \exists Y (\underline{\text{donkey}(Y) \wedge \text{owns}(X, Y) \rightarrow \text{beats}(X, Y)}))$$

**Equation 7:**

$$\forall X \exists Y (\underline{\text{farmer}(X) \wedge \text{donkey}(Y) \wedge \text{owns}(X, Y) \rightarrow \text{beats}(X, Y)})$$

**Equation 8:**

$$\exists Y \forall X (\underline{\text{farmer}(X) \wedge \text{donkey}(Y) \wedge \text{owns}(X, Y) \rightarrow \text{beats}(X, Y)})$$

**Equation 9:**

$$\forall X \forall Y (\underline{\text{farmer}(X) \wedge \text{donkey}(Y) \wedge \text{owns}(X, Y)}) \rightarrow \text{beats}(X, Y)$$

The last attempt at translating the donkey sentence replaces the existential quantifier of Y with a universal quantifier as shown in **Equation 9**. The result is the statement that for every X and every Y, if X is a farmer and Y is X's donkey, then X beats Y. This final translation provides the correct logical interpretation of the sentence but there is no logical reason for using universal quantifier instead of a universal quantifier and therefore, there is no logical way to generate the correct translation of the donkey sentence.

### Model-theoretic Semantics

The term Model-theoretic Semantics refers to semantics based on truth conditions as defined in Tarski's model theory developed in 1954. Truth-conditions of a phrase are defined as the truth-values of parameters in the context of the phrase. Parameters consist of variables such as the time the sentence was spoken and the possible worlds where the phrase is true relative to a model. These parameters are represented in Montague's IL as types.

Model theory consists of three levels of symbols: logical constants, variables, and non-logical constants. Logical constants refer to traditional logical symbols such as = and  $\neg$ . The traditional mathematical view of variables is used for variables in model theory. The group of non-logical constants consists of universal and existential quantifiers;  $\forall$  and  $\exists$ , relation symbols, function symbols, and constant individual symbols.

Using the symbols in model theory one may express the truth-value of a sentence in terms of its ability to satisfy a given formula. For example, the truth-value of the sentence (From Partee 1973 Green book) **Every man loves a woman such that she loves him** is dependent on the conditions which cause X and Y to satisfy the equation  $X \text{ loves } Y$ . The resulting equation is satisfied by variables X and Y if and only if Y loves X. Tarski bases his approach to the fact that languages of first-order predicate logic contain the following three characteristics.

- 1) Able to give finite recursive syntactic characterization of the set of all formulas of the language
- 2) Able to give a finite recursive semantic characterization of satisfaction for all the formulas, based on that syntax
- 3) Able to define truth in terms of satisfaction in a way that leads to the correct characterization of truth-conditions for the sentences of the language.

Note that languages of first-order predicate logic recursively define the conditions for satisfaction of a complex formula in terms of its component formulas. The truth-conditions of sentences are simply the conditions under which the sentence is satisfied. The conditions under which sentences are true are based upon the model used. To understand truth conditions, one must first understand the basic definition of a truth. This brings forth the notion of truth-definitions, which define the relationships between variables and models that cause an expression to be true.

Because truth definitions work recursively, a base truth must exist. The base truth definition is  $\alpha$  satisfies  $F$  where setting variables in  $F$  with values in  $\alpha$  causes  $F$  to evaluate to true. The recursive rule for truth definitions defines the satisfaction of a complex formula based on a recursive satisfaction of its constituents. The basic truth definitions in Figure 1 are known as absolute truths.

Base Rule:	$\alpha$ satisfies $F$
Recursive Rule:	$\alpha$ satisfies “ $F$ and $G$ ” iff $\alpha$ satisfies $F$ and $\alpha$ satisfies $G$

**Figure 6: Truth Definitions**

Model theory contains further definitions known as relativized truths, which define truth in relation to a model. Such definitions specify functions as true if they hold true with respect to the truths in a given model under certain conditions. Relative rules allow for the definition of semantic concepts such as consequence, equivalence, truth, and contradiction.

<b>Concept</b>	<b>Truth Definition</b>
Consequence	$X$ is a consequence of sentences in class $K$ iff $X$ is true in every model where every sentence in $K$ is true.
Equivalence	$X$ and $Y$ are logically equivalent if $X$ is a consequence of $Y$ and $Y$ is a consequence of $X$
Truth	$X$ is logically true if for all models, $X$ is true
Contradiction	Class $K$ is contradictory if no model exists where all sentences in $K$ are true

**Table 5: Truth definitions for semantic concepts**

In terms of Montague grammar, the truth definitions described correspond to rules used in interpreting intensions of a sentence fragment. The discussions of intensional logic and quantification have addressed the process of translating sentences into the formal

language IL. In the IL format, intensions of phrases are based upon model theory, defining the meaning as a function of model. This model is defined as indices in intensional logic.

### Controversies

The ideas brought forth in Montague Grammar were important because they went against many traditional linguistic concepts of syntax and semantics. In understanding Montague Grammar's affects, one must first examine the affects its theories had on the linguistic and logistical communities. The controversies quelled and those ignited by Montague Grammar are part of what makes Montague's work interesting.

The three main concepts commonly attacked are Model-theory, formal logic, and the use of truth conditions. It is not Montague's use of these concepts linguists disagree with, but the validity of their application within linguistics. The concepts are discussed in respect to Montague because he was the first to actively apply the theories to natural language.

Montague's use of formal logic in semantics of natural language brought forth questions of what semantics is. The Chomskyan perspective labeled all of linguistics, including semantics, as a branch of psychology while classical formal semanticists differentiated semantics from knowledge of semantics. Linguists such as Jackendoff and Fauconnier have since presented their own forms of semantic analysis but their theories have not proven to be more comprehensive than formal theories. Since the 1970's, formal semantics have become popular and part of mainstream linguistics.

Some linguists question the value of examining truth conditions in natural language semantics because it brings in so many factors that are seemingly not relevant. The argument for truth conditions is that human interpretations of meaning of natural language sentences are based upon the context in which the sentence is spoken. For example, the truth-value of the sentence **Mary is the tallest girl here** depends on the context in which the sentence was spoken. More specifically, the truth-value depends upon the intention of the word **here** and the height of all the girls within the location defined by **here**.

Those who prefer semantic analysis based upon more concrete expressions, such as those used in syntactic analysis dispute the abstract model-theoretic view of Montague grammar. An alternative form of semantic analysis based upon formal logic but not model-theoretic was later proposed by Donald Davidson. In his approach, Davidson uses Tarski's T-sentences to define truth conditions in a manner more familiar to linguists.

Though controversies may have arisen from Montague's work, there is no dispute that he brought to light many new ways of viewing both syntax and semantics. The theories of Montague Grammar have generated new areas of research and development.

## Developments from Montague Grammar

One of the most interesting results of Montague Grammar is the shift in attitude of both linguists and logicians. Montague's semantic theories dramatically influenced the way linguists approach semantics by combating long held beliefs that formal logic and natural language were not compatible. His works also help to foster collaboration between professionals of the two fields who had been at odds with each other

Montague's ability to bring logical semantics into the linguistic community opened new avenues of approaching semantics. His work has resulted in new innovations in computational semantics such as methods for handling ambiguous. Though Montague's influence is mostly apparent in semantics, his work has impacted theories with syntax as well. Montague's theory of categorizing lexicons in a grammar is attributed with influencing Pollard, Sag, and Wasow's development of Head-Driven Phrase Structure Grammar (HDPS). His theory of categorization, known as CAT, refers to the use of syntactic categories to categorize lexical terms.

Much of the work influenced by Montague Grammar has resulted new theories that extend the capabilities of the original theories. Areas of research began in logical parts such as quantification and anaphora gradually expanded to all areas of Montague Grammar. By addressing the limitations of Montague Grammar, more robust and encompassing syntactic and semantic theories bring linguistics closer to Montague's idea of a Universal Grammar able to express any language.

Irene Heim, one of Barbara Partee's students at the University of Massachusetts Amherst, developed a notable approach to semantics that addressed a limitation of Montague's General Quantification (GQ) theory. Recall that GQ is unable to handle donkey sentences. Heim's work, known as file change semantics (FCS), addressed donkey sentences by viewing a sentence's quantification in terms of three parts: the Quantification operator, Restriction, and scope. Her work is similar to that of Hans Kamp's discourse representation semantics (DRS), which appeared independently at the same time as Heim's. With Heim and Kamp's approach, the donkey sentence would be represented as shown in Equation 10.

$$\underline{Q[\text{every}:x,y] R[\text{Farmer}(x), \text{Donkey}(y), \text{Owns}(x,y)] S[\text{Beats}(x,y)]}$$

Equation 10: Donkey Sentence in DRS/FCS

Heim and Kamp's work is noteworthy in this context because their developments led to the use of **context change potential** for the core semantic property of a sentence as opposed to the "truth conditions" used by Montague. These new changes in theory may be seen as results of work inspired by the innovative ideas of Montague Grammar.

Continued research with regard to Montague Grammar has focused on extensions in model theory to handle mass nouns and plural entities. Model theory has also influenced Kratzer's work in **situation semantics**, which focuses on situations as a component of a

world. Other works influenced by Montague Grammar include Extended Categorical Grammar, Generalized Phrase Structure Grammar (GPSG), Head Driven Phrase Structure Grammar (HPSG), and Lexical semantics. It is evident that Montague's legacy is a lasting one as new developments continue to be influenced by his works.

## References

- Abbott, Barbara. "The formal approach to meaning: Formal semantics and its recent developments." Journal of Foreign Languages. 119:1 (January 1999) p2-20.
- Barker Ken. "Donkey Sentence Discussion" CSI 5386 (October 2, 1999) University of Ottawa. <<http://www.csi.uottawa.ca/~kbarker/teach/5386/donkey.html>>
- Carpenter, Bob. Type-Logical Semantics. Cambridge, Massachusetts: The MIT Press, A Bradford Book, 1997.
- Davis, Steven and Mithun, Marianne. Linguistics, Philosophy, and Montague Grammar. Austin, Texas: University of Texas Press, 1977.
- Dowty, David R. Word Meaning and Montague Grammar. Dordrecht, Holland: D. Reidel Publishing Company, 1979.
- Gallin, Daniel. Intensional and Higher-Order Modal Logic with Applications to Montague Semantics. Amsterdam: North-Holland Publishing Company, 1975.
- Hodges, Wilfrid, "Model Theory", The Stanford Encyclopedia of Philosophy (Winter 2001 Edition), Edward N. Zalta (ed.), <<http://plato.stanford.edu/archives/win2001/entries/model-theory/>>.
- Hodges, Wilfrid, "Tarski's Truth Definitions", The Stanford Encyclopedia of Philosophy (Spring 2003 Edition), Edward N. Zalta (ed.), <<http://plato.stanford.edu/archives/spr2003/entries/tarski-truth/>>.
- Kornai, Andras. "Logical types and linguistic types." Tertium Non Datur 3 (1986) p241-255.
- Krifka, Manfred. "Pragmatic Strengthening in Plural Predictions and Donkey Sentences." Proceedings of Semantics and Linguistic Theory 6. Cornell University (1996) p136-153
- Morrill, Glyn V. Type Logical Grammar. Dordrecht, Netherlands: Kluwer Academic Publishers, 1994.
- O'Connor J.J. and Robertson, E.F. "Alfred Tarski" The MacTutor History of Mathematics archive. (October 2003) School of Mathematics and Statistics. University of St Andrews, Scotland. <<http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Tarski.html>>
- Partee, Barbara and Borschev, Vladimir. Integrating Lexical and Formal Semantics, Lecture 1. Kolding. (January 2000) p.3
- Partee, Barbara. Montague Grammar. New York, NY: Academic Press, 1976.

Partee, Barbara H. “Montague grammar” Neil J. Smelser and Paul B. Baltes, eds. International Encyclopedia of the Social and Behavioral Sciences. Oxford, Pergamon/Elsevier Science 2001

Partee, Barbara. “Montague Grammar and Transformational Grammar.” Linguistic Inquiry. Volume VI, Number 2 (Spring 1975) p203-300.

Tarski, Alfred. “The Semantic Conception of Truth and the Foundations of Semantics” Philosophy and Phenomenological Research (1944)

Uzquiano Gabriel “Review of Volker Halbach and Leon Horsten’s (eds), *Principles of Truth*” Notre Dame Philosophical Reviews (April 2003) p. 13

Montague semantics is a theory of natural language semantics and of its relation with syntax. It was originally developed by the logician Richard Montague (1930–1971) and subsequently modified and extended by linguists, philosophers, and logicians. The most important features of the theory are its use of model theoretic semantics which is nowadays commonly used for the semantics of logical languages and its View Montague Grammar Research Papers on Academia.edu for free. Montague Grammar, Logical Analysis of Language, Intensional Type Theory. Montague et les catégories d'Ajdukiewicz. Rôle et enjeux de la notion de catégorie en logique, D. Miñville (éd), Travaux de logique, 1999, n° 13, Neuchâtel : CdRS, pp. 159-175. Save to Library. Download. Montague grammar is an approach to natural language semantics, named after American logician Richard Montague. The Montague grammar is based on formal logic, especially higher-order predicate logic and lambda calculus, and makes use of the notions of intensional logic, via Kripke models. Montague pioneered this approach in the 1960s and early 1970s. Montague's thesis was that natural languages (like English) and formal languages (like programming languages) can be treated in the same way In Universal Grammar (1970), Montague introduced his theory of formal syntax and semantics as applied to both formal and natural language. Universal Grammar is significant because it was the first attempt at applying formal semantics to natural language. Partee's Montague Grammar and Transformational Grammar. 3. (Linguistic Inquiry 1975) was seen as the first introductory text on Montague Grammar.