Recursion in Language:
A Critical Discussion

by

Donovan Cowie
941 300 724

Coursework dissertation submitted in partial fulfillment of the requirements for a degree of Master of Arts in Linguistics in the school of Language Literature and Linguistics, University of KwaZulu-Natal, Durban 2015.

Acknowledgements

Jochen Zeller for believing in me,
my Mom for inspiring me
and my wife for enduring with me.

Abstract

This research draws on Chomsky and Everett's opposing views on the role of recursion in language. I will examine the role of recursion in sentences and investigate the hypothesis that it also plays an important role in the structuring of discourse, which would confirm the view that recursion is a fundamental feature of language.

Table of Contents:

Introduction

The linguist M.A.K. Halliday (2003:96-99) identifies two traditions, or approaches to language. The first, Halliday calls "philosophical-logical", and the other "descriptive-ethnographic". Linguists following the “philosophical-logical” view see language as thought, and engage in formal analysis of the rules that govern sentences, while linguists following the “descriptive-ethnographic" view see language as a cultural resource and focus on the semantic interpretation of discourse.

According to Halliday, these two traditions go back to Plato and Aristotle; Plato saw language as a resource, while Aristotle saw language consisting of a set of logical rules. Furthermore, Halliday sides with Plato in seeing language as a resource, and ascribes the Aristotlean position to “Chomskyan” linguists. He says about Chomsky:

"If Chomsky had admitted that he was building on the work of his predecessors, the ensuing dialogue between philosophers and ethnographers of language could have been very fruitful and rewarding. Probably never before in the history of ideas about language have the two views, of language as resource and language as rule, been made to seem so incompatible." (Halliday 2003:96-99)

In his (2003) discussion, M.A.K. Halliday presents a simplified view of two schools of thought in linguistics, with theorists and scholars fitting in somewhere between the two poles of language as resource, or language as rule. This historical view of the two approaches sets the stage for a recent debate between Chomsky and Everett.

In order to best understand the debate, let's take a brief look at the development of Chomsky's program. Halliday's (2003) quoted paragraphs above were originally published in 1977, only three years after the term "biolinguistics" was coined at an international conference by Massimo Piatelli-Palmerini (Chomsky 2007: 1-5).

Chomksy (1979: 81) argues against the empiricist view that the brain is a tabula rasa, empty and unstructured. In the empiricist view, all of language is learned and nothing is innate. Chomsky's view aligns itself with the nativist framework, whereby certain skills or abilities are available to the brain from birth.

The biolinguistic framework arose from this view and is a cross-disciplinary approach whereby the faculty of language (FL) is understood to be a "cognitive organ", shared among humans and unique to the species, and central to Chomsky's linguistic framework.

The aim of the framework was to investigate universal grammar (UG) which Chomsky defines as "the initial state of FL". A human child is born with certain biological and cognitive faculties, which are used for linguistic purposes as the child starts to learn to speak. In the biolinguistic program, universal grammar comprises the tools that the human child is endowed with as a result of the process of evolution (although noting that in Chomsky’s view, language was introduced to the human race in a single mutation, an evolutionary leap, not as a gradual process).

The biolinguistic program gave rise to Chomsky’s Principles and Parameters (P&P) framework, which explains the diversity of human languages as a set of principles, such as that sentences will always have a subject, whether it is overtly pronounced or not, and a series of binary switches, or parameters, whereby for example a language could be head-initial like English, or head-final like Japanese (Chomsky 1981; 1986).

Principles and parameters are a part of an innate Universal Grammar and do not need to be learned by exposure to language. Rather, exposure to language merely triggers the parameters to adopt to the linguistic setting.

The P&P framework accounts for Plato's problem (or the poverty of stimulus problem) in which children acquire language at a much faster rate than can be accounted for by the limited amount of data that they are exposed to in their environment (Chomsky 2007:3):

"The P&P approach largely emerged from intensive study of a range of languages, but it was also suggested by major developments in general biology, specifically Francois Jacob's account of how slight changes in the hierarchy and timing of regulatory mechanisms might yield great superficial differences – a butterfly or an elephant, and so on. The model seemed natural for language as well: slight changes in parameter settings might yield superficial variety, through interaction of invariant principles with parameter choices.

The P&P framework also made it possible to pursue more seriously the recognition, from the earliest days of generative grammar, that acquisition of language involves not only a few years of experience and millions of years of evolution, but also "principles of neural organization that may be even more deeply grounded in physical law". Again, somewhat parallel developments were proceeding in general biology, now sometimes called the 'evo-devo revolution.'" (Chomsky 2007:3)

The second part of this quote refers to what can be considered a central tenet of the Minimalist Program (MP), which emerged in the 1990s and is the most current version of the P&P framework (Chomsky 1995: 2000). Research in the MP is driven by the objective to reduce principles that were previously attributed to UG as much as possible to language-independent "third factor" principles. While earlier versions of the Principles and Parameters theory tended to see UG as rich and detailed, the MP has sought to identify the minimum that needs to be attributed to UG:

"Throughout the modern history of generative grammar, the problem of determining the character of FL has been approached "from top down": How much must be attributed to UG to account for language acquisition? The MP seeks to approach the problem "from bottom up": How little can be attributed to UG while still accounting for the variety of I-languages attained, relying on third factor principles? The two approaches should, of course, converge, and should interact in the course of pursuing a common goal." (Chomsky 2007:4)

In 2002, Hauser, Chomsky and Fitch published an article (hereafter HCF), in which they distinguished between the faculty of language in the broad sense (FLB) and the faculty of language in the narrow sense (FLN).

While FLB consists of all the cognitive systems necessary for human language, including those which are based on mechanisms shared with non-human animals (e.g. the sensory-motor and the conceptual-intentional systems), HCF postulate that the FLN is the only uniquely human component of the faculty of language. HCF proceed to put forward the hypothesis that "FLN comprises only the core computational mechanisms of recursion as they appear in narrow syntax and the mappings to the interfaces." (HCF 2002: 1573). This hypothesis, which Pinker and Jackendoff (2004, hereafter PJ) call the "recursion-only hypothesis", falls at the centre of the debate regarding recursion in language.

In 2005, Everett published his research on the Pirahã language. Pirahã is a language spoken by a community of 300 to 400 people, living by the Maici River in Amazonas, Brazil. Based on his study, Everett (2005, 2007, 2009) claims that "there is no evidence in Pirahã for recursion". Everett's argument is that, since the "recursion-only hypothesis" is intended to apply to the human faculty of language (and so all human languages) finding evidence that there is no recursion in even just one language (Pirahã) nullifies the recursion-only hypothesis. Specifically, Everett claims that recursive concepts are conveyed through discourse in Pirahã, and not through the grammar.

In the chapters that follow, I will review the arguments put forward by Everett (2005, 2007, 2009), PJ and Parker (2006), as well as responses to these critics by Fitch, Hauser and Chomsky (2005) and Chomskyan linguists such as Nevins, Pesetsky and Rodrigues (hereafter NPR 2007, 2009). In doing so, I will show that recursion is indeed important to the structuring of discourse, and thus a fundamental part of our faculty of language.

In Chapter 1, I will introduce and discuss the concept of recursion, as well as examine several definitions and examples of recursion. In Chapter 2, I will discuss recursion and the recursion-only debate. In Chapter 3, I will review Everett's “no-recursion” hypothesis as well as NPR's counter-arguments. In Chapter 4, I will present the mounting evidence against Everett, and conclude that Pirahã does indeed employ recursion at a fundamental level, which in turn serves to confirm the role of recursion in the form of Merge, in discourse.

Chapter 1 – What is Recursion?

Nothing is particularly hard if you divide it into small jobs – Henry Ford

Russian Matroyska Dolls are a common example of a recursion. The dolls are stacked inside one another and so serve as a simple visual example of recursion; opening the outer doll reveals another doll inside it, opening that doll reveals another doll and so on.

Diagram 1. Russian nesting dolls
http://en.wikipedia.org/wiki/File:Russian_Dolls.jpg

A recursive *structure* is an object that contains smaller versions of itself. The smaller, embedded, version of the object can contain an infinite number of further smaller embedded objects.

A recursive *process* is one in which the output from the first process is used as the input for the next iteration of the process. Recursive processes (or functions) operate on themselves – and the output from the entire recursive process can be a recursive object, but does not need to be.

The Tower of Hanoi is a puzzle, which is solved through the application of a recursive process (Roberts 2008:205).

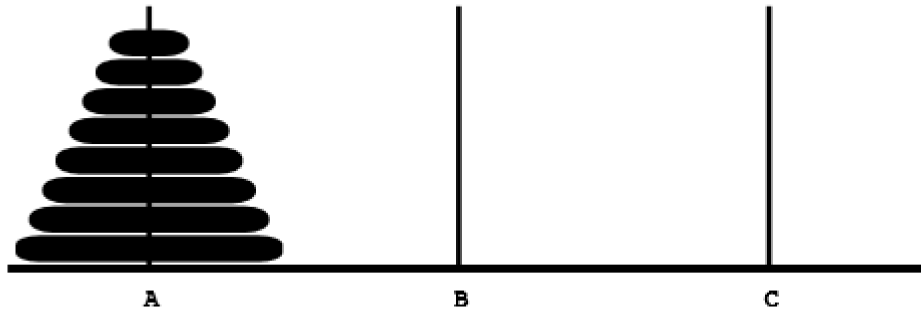


Diagram 2. Initial State of Towers of Hanoi puzzle - Roberts (2008:205)

The puzzle consists of three spires (A, B and C) and eight disks. In the initial state of the puzzle, the eight disks are all placed onto spire A. The aim of the puzzle is to move all eight disks from spire A to spire B, while observing the following two rules:

1. Only one disk can be moved at a time.
2. A larger disk cannot be placed on top of a smaller disk.

In order to solve the puzzle, the problem has to be broken down into smaller parts (or sub-problems). Looking at the puzzle more generally, the problem is to move the 8 disks from spire A to spire B, using spire C as temporary storage.

More specifically, each step of the recursive process will need the following inputs (Roberts 2008:205):

1. The number of disks to move
2. The name of the spire where the disks start
3. The name of the spire where the disks should finish
4. The name of the spire used for temporary storage

The temporary storage is used to house the disks during the move. In order to move 8 disks to spire B, we must first move 7 disks to spire C, then the 8th disk to B. Finally, we'll move the 7 remaining disks from C on top of the 8th disk on spire B, to complete the puzzle.

Having identified these four inputs, one can write the prototype for the operation that moves a stack of disks (or tower) as follows:

Code Listing 1.

```
void MoveTower(int n, char start, char finish, char temp);
```

The first call to this operation (which in computing terms is known as a *function*, or a *method*) looks as follows, and means in English "Move a stack of 8 disks from spire A to spire B, using spire C as temporary storage":

Code Listing 2.

```
MoveTower(8, 'A', 'B', 'C');
```

The solution to the Tower of Hanoi puzzle looks as follows in Java code (Roberts 2008: 209):

Code Listing 3.

```
void MoveTower(int n, char start, char finish, char temp) {
        if (n == 1) {
                MoveSingleDisk(start, finish);
        } else {
                MoveTower(n - 1, start, temp, finish);
                MoveSingleDisk(start, finish);
                MoveTower(n - 1, temp, finish, start);
        }
}
```

Note how concise the solution is. Only a few lines of code are required to solve the puzzle – this is the power of recursion. The MoveTower() method calls itself (recursively) when moving stacks of disks, and calls the moveSingleDisk() method when moving a single disk. Note that only one disk is moved per call to the MoveTower() method.

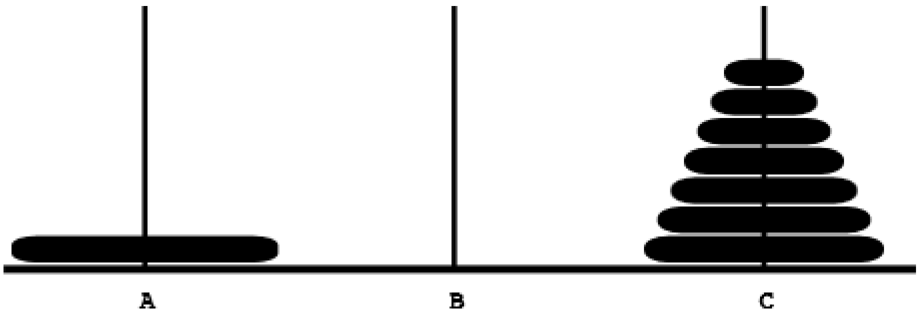


Diagram 3. After the first MoveTower() call. Roberts (2008:207)

Diagram 3. depicts the state of the puzzle, after the first MoveTower() call. In this step, the top n-1 (in this case, 7) disks are moved from spire A to spire C, leaving behind the bottom disk. This step is executed by the line of code MoveTower(n - 1, start, temp, finish);

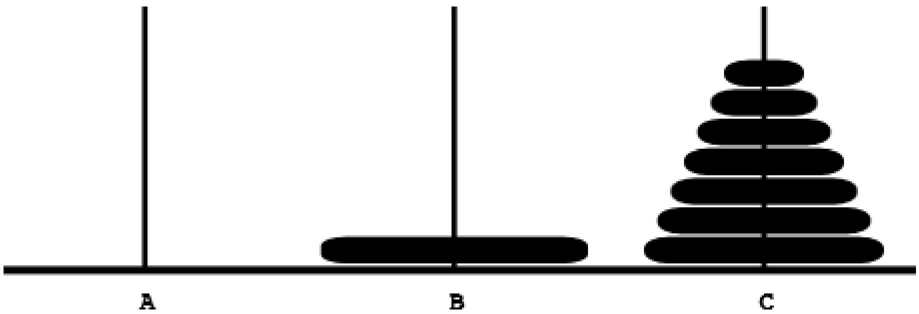


Diagram 4. The largest disk is moved. Roberts (2008:207)

Diagram 4. shows the next step (executed by the line of code MoveSingleDisk(start, finish);), where the largest disk is moved from spire A to spire B.

Diagram 5. Shows the completion of the puzzle, as the remaining N-1 disks are moved from spire C to spire B by execution of the line of code MoveTower(n - 1, temp, finish, start); Note that the MoveTower(n-1...) methods are recursive – so the first method that moves 8 disks calls the same method again to move 7, then 6, then 5 (etc.) disks.

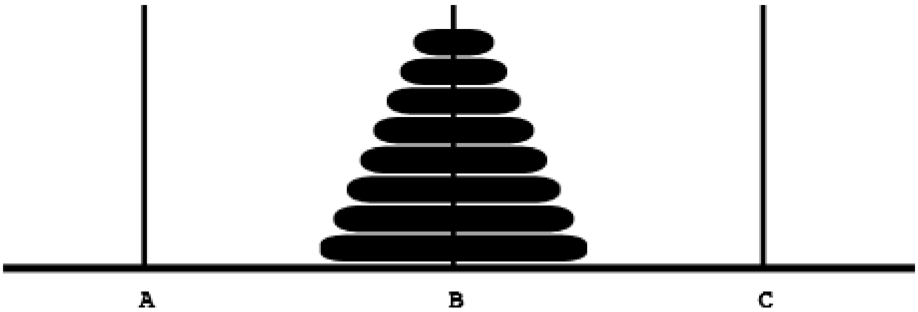


Diagram 5. The puzzle is solved. Roberts (2008:207)

The Tower of Hanoi puzzle provides an illustration of how a recursive process (MoveTower()) can simplify a problem so that it can be solved in a step-by-step manner. In order to move all 8 disks, the MoveTower() method moves one disk, then *calls itself again* with the instructions to move 7 disks, then 6 disks, then 5 disks and so on.

The same function, MoveTower(), that is used to move the n-1 (or 7) disks to spire C (in Diagram 3.) is used to move the single remaining disk from spire A to spire B (as per Diagram 2.) and also to move the 7 disks from spire C to spire B.

The recursive nature of this solution is that, in the same way as moving 8 disks from spires A to B relies on being able to move 7 disks to spire C, moving 7 disks relies on being able to move 6 disks, and moving 6 disks relies on moving 5 disks, and so on.

In mathematics, a Fibonacci sequence is created by means of a recursive process by which numbers in the sequence are generated by adding together the previous two numbers in the sequence. The first two numbers are 0 and 1, and the next numbers are 1, 2, 3, 5, 8, 13, 21 and so on. The Fibonnacci sequence is typical of a recursive process, because the output from one iteration of the process is used as the input for the next.

A Fibonacci spiral is created by the recursive process of drawing arcs to the opposite corner of squared tiles whose sizes are determined by the Fibonnacci sequence. The first tile is 1x1, the next square is 1x1, and the following tiles are 2x2, 3x3, 5x5, 8x8, 13x13, 21x21 and so on. The spiral starts at the bottom left of the first square and arcs towards the opposite corner of each new square, as it is added.

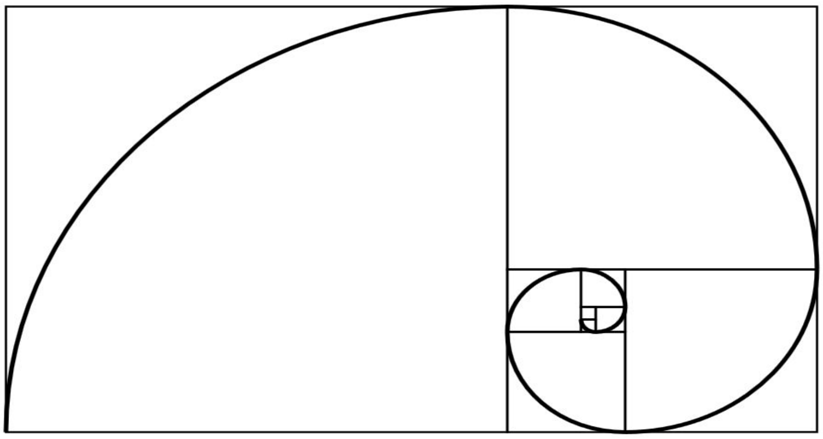


Diagram 6. Fibonnacci Spiral
http://commons.wikimedia.org/wiki/File:Fibonacci_spiral_34.svg

Another object, which is created by a recursive process, is a Sierpinski Carpet; a two-dimensional fractal, which was first described in 1916 by the Polish Mathematician, Waclaw Sierpinski. Barlow *et al* (2009:2) describe the recursive process by which a Sierpinski Carpet is formed: a square has its central section removed, and the process is recursively applied to the remaining 8 sub-squares.

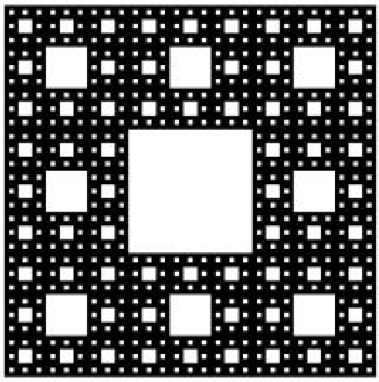


Diagram 7. A Standard Sierpinski Carpet. Barlow et al (2009:2)

A Menger Sponge is an object created by the application of the recursive Sierpinski Carpet process, in three-dimensions.

Diagram 8. Sculpture in the shape of a Menger sponge
http://en.wikipedia.org/wiki/File:Mengerova_houba.jpg

1.1 Recursion in Computer Science

It is perhaps easiest to explain recursive processes (called functions or methods) using examples from Computer Science, because of the sequential, stepwise manner in which computers operate. Instructions are grouped together to form functions and each step is executed individually, in sequence.

First, consider the following two JavaScript functions, each of which take a string (called current) as an input, add a letter to the end of the string and return the result as an output.

Example 1.1 will take a string that is passed into it, and add an "A" letter to the string.

1.1. addA() function

```
function addA(current)
{
```

```
        //Adds the letter 'A' to the string
        return current + 'A';
}
```

1.2. addB() function

```
function addB(current)
{
        //Adds the letter 'B' to the string
        return current + 'B';
}
```

Now consider the recursive function in 2.1, which calls the functions above, and itself, recursively:

2.1. Centre Embedding

```
function recurse(current)
{
        current = addA(current);
        current = recurse(current);
        current = addB(current);
}
```

The function in 2.1 executes in three discrete steps. First, there is a call to the addA() function, then the recursive call to the recurse() function, then execution passes to the addB() function.

At first, let us allow the function to execute only once. We can do so by using a count variable, which terminates execution when a particular level of recursion is reached. We can instruct the method to execute only once by setting maxLevel to 1 (or to 2 to make the method recur twice, 3 for three times and so on).

For the purposes of simplicity of display, I will leave out count and maxLevel from the other listings, but do be aware of this mechanism, which terminates execution once a certain level of recursion has been reached.

2.1a. Centre Embedding (full listing)

```
int count = 0;
int maxLevel = 1;

function recurse(current)
{
        if(count < maxLevel)
        {
        current = addA(current);
        current = recurse(current);
        current = addB(current);
        }
        else
{
        break;
}

count++;
}
```

The first line (addA(current);) will be executed, which contains a call to addA(). At this point, the variable **current** will be blank. AddA() will add an A to the blank string, and it will contain 'A'.

On the next line (recurse(current);), there is the recursive call to the recurse() method. Since we are only allowing one level of recursion, this line will do nothing, and execution passes to the third line. The third line calls addB(current), which adds a 'B' character to the contents of current (which is currently 'A') to form the string 'AB'. Once the third line ends, the program will terminate, and the end result of the current string is 'AB'.

Imagine now that we let the recursive function run to a second level of recursion. This time, the first line produces the letter 'A'. The second line contains the recursive call (recurse(current);), which will pass current (which contains the letter 'A') into the recurse() function.

The recurse() function's first line adds an 'A', so the string current now contains 'AA'. The third line adds a 'B' to the string to form 'AAB'. After the third line, processing is passed back to the original recurse() method, which completes its third line (addB(current);) by adding 'B' to the string to give the complete output of 'AABB'.

I will detail the steps, and their output, in the following table:

Recursive Function	**Line**	**Added by this Line**	**Current Output**
1	addA(current);	A	A
1	recurse(current);		
2	addA(current);	A	AA
2	recurse(current);		
2	addB(current);	B	AAB
1	addB(current);	B	AABB

Table 1 – Two levels of recursion

If we called the recurse() function and allowed for three levels of recursion, the table would look as follows:

Recursive Function	**Line**	**Added by this Line**	**Current Output**
1	addA(current);	A	A
1	recurse(current);		
2	addA(current);	A	AA
2	recurse(current);		
3	addA(current);	A	AAA
3	recurse(current);		
3	addB(current);	B	AAAB
2	addB(current);	B	AAABB
1	addB(current);	B	AAABBB

Table 2 – Three levels of recursion

If we let the function (2.1) above recursively execute once, the output is 'AB'. Thereafter, each time the recursive method is called, it embeds 'AB' at the centre of the output string:

....A [AB] B....

If we let 2.1 recursively execute 3 times, the output looks like:

AAABBB

A distinction that is made in the literature is of the type of embedding – head, centre and tail recursion (see e.g. Coolidge, Overmann, Wynn (2010), PJ, Lobina (2011:159) also see Parker (2006: 3))

Function 2.1 above is an example of centre embedding. The function in 2.2., below is an example of head recursion, and the function in 2.3. is an example of tail recursion.
2.2. Head Recursion

```
function recurse(current)
{
        current = recurse(current);
        current = addA(current);
        current = addB(current);
}
```

The output from 2.2. is as follows, with the recursive material occuring at the beginning of the string:

Recursive Function	**Line**	**Added by this Line**	**Current Output**
1	recurse(current);		
2	recurse(current);		
3	recurse(current);		
3	addA(current);	A	A
3	addB(current);	B	AB
2	addA(current);	A	ABA
2	addB(current);	B	ABAB
1	addA(current);	A	ABABA
1	addB(current);	B	ABABAB

Table 3 – Three levels of head recursion (2.2)

2.3. Tail Recursion

```
function recurse(current)
{
        current = addA(current);
        current = addB(current);
        current = recurse(current);
}
```

In 2.3, the recursion occurs at the end of the operation, and so 2.3 is an example of tail recursion:

Recursive Function	Line	Added by this Line	Current Output
1	addA(current);	A	A
1	addB(current);	B	AB
1	recurse(current);		
2	addA(current);	A	ABA
2	addB(current);	B	ABAB
2	recurse(current);		
3	addA(current);	A	ABABA
3	addB(current);	B	ABABAB
3	recurse(current);		

Table 4 – Three levels of tail recursion (2.3)

Note that the output from 2.2 and 2.3 is the same (ABABAB), although head recursion (as per 2.2) appends the [AB] at the beginning of the string, while tail recursion places the [AB] at the end of the string. Tables 3 and 4 list the steps by which the output is produced.

On closer examination of Table 4, one sees that in the case of tail recursion, the lines are executed in such a way that first all of the lines are executed on the first level of recursion, then all the lines on the second level, then all the third.

When we look at the values in the first column in the table, we see first 1, 1, 1, then 2, 2, 2, then 3, 3, 3. Tail recursion can be quite difficult to tell apart from *iteration*, which is simply the same method called over and over again.

The difference between tail recursion and iteration is that in the case of tail recursion, the recursive method call is nested inside the method, while in the case of iteration, the method is called from the outside. The iterative code and method corresponding with the examples 2.1 to 2.3 above looks something like:

2.4 Iteration

```
current = iterativeMethod(current) + iterativeMethod(current) + iterativeMethod(current);

function iterativeMethod(current)
{
	current = addA(current);
	current = addB(current);
}
```

1.2 Some Definitions of Recursion in Language

One type of recursion in language is mentioned in Radford (2004: 69), which I will term *structural recursion*. Structural recursion is a view of recursion that deals with structures embedded in other structures (much like Russian Dolls). Radford's definition of recursion is found in the following statement:

An interesting property of syntactic structures illustrated in [Diagram 12] is that of recursion – that is, the property of allowing a given structure to contain more than one instance of a given category (in this case, more than one verb phrase/VP – one headed by the verb *help* and the other by the verb *trying*). (Radford 2004: 69)

Diagram 12. that Radford refers to is provided below.

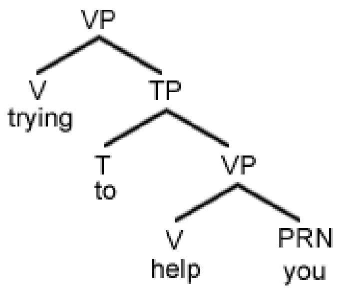


Diagram 12. An embedded VP

According to Radford (2004: 69), the structure in Diagram 12 is recursive because the complex VP includes another VP-category. The embedding of a VP inside another VP forms a recursive structure in the same way that Russian dolls are structurally recursive.

Another example of structural recursion is sentence embedding.The categorial rule in (1)a. states that a sentence (S) can consist of a noun phrase (NP) and a verb phrase (VP) and the rule in (1)b. states that a VP can consist of a verb (V) and a sentence.

(1) a. S -> NP VP

b. VP -> V S

The first S in 1(a). contains an NP and a VP, and that VP in turn contains a V and another S (which can go on to contain another NP, VP and ultimately, S again, recursively)

Diagram 13. illustrates another example of structural recursion. The syntactic structure is derived by the application of both rules in (1), and shows that the sentence "I know" is embedded inside another sentence.

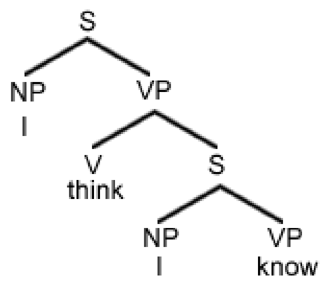


Diagram 13. An embedded S

Further examples of structural recursion include possessive constructions, which feature recursive NPs (for example: "John’s mother’s cousin’s house", or "the car’s engine’s piston") and recursive prepositional phrases (for example: "on the lake near the cottage in the forest").

Everett (2009: 5) supplies the following definition of recursion: “Recursion consists in rule (or operation) sets which can apply to their own output an unbounded number of times.” Notice that Everett’s definition provides a view of recursion, that is subtly different from that of Radford (2004). In Everett's definition, the focus shifts from examining *structures* (as per Radford’s definition) to *rules* or *operations*.

On one hand, we have *structural* recursion, in which structures can recursively contain embedded instances of categories (as per Radford's definition). On the other hand, we have *functional* recursion, which involves a process, (or function, operation or action), which refers back to itself as part of the execution of the process, creating a new instance of the process.

The definition provided by PJ (2004: 203), makes the distinction between the two definitions of recursion explicit: "Recursion refers to a procedure that calls itself, or to a constituent that contains a constituent of the same kind". The former type of recursion is *functional* recursion, and the latter is *structural* recursion.

Parker (2006: 3) provides the following definition of recursion: "the *embedding* at the edge or in the centre of an *action* or *object* one of the *same type*" (my italics). On careful examination of this definition, one finds that it contains definitions of both types of recursion. The first reads: The embedding at the edge or in the centre of an *object.* The second reads: The embedding at the edge or in the centre of an *action*.

The two definitions provided by Parker (2006:3) neatly detail both functional and structural recursion. Structural recursion occurs when an object is embedded within another object. Whether it is embedded at the centre or edge is a secondary concern – what matters for the definition is that the object is contained by another object of the same type.

Functional recursion follows the same principle in that a process is embedded within another process. Just like with objects, processes can be embedded within other processes.

A process embedded "in the centre" of another process indicates that the first process starts, and is interrupted midway to allow the embedded child process to start. Only when all the embedded child processes have completed, is the first process allowed to continue. See example 2.1. above (Centre Embedding).

Processes that are embedded "at the edge of another process" execute either before or after the rest of the containing process completes. In the case of head recursion, the embedded child processes appear at the beginning of the function, and in the case of tail recursion, the embedded child processes appear at the end of the containing function.

In this chapter, I have examined some definitions and examples of recursion in mathematics, computer science and in language. In the next chapter, I will present a claim central to Chomsky's Minimalist Program – the recursion-only hypothesis.

Chapter 2: Recursion and the Recursion-only Debate

HCF (2002:1572) introduce three hypotheses about the evolution of the faculty of language. According to the first hypothesis, the faculty of language in the broader sense (FLB) is strictly homologous to animal communication: homologues of FLB, including the faculty of language in the narrow sense (FLN) are found in other species.

According to the second hypothesis, the entirety of FLB is a derived, uniquely human adaptation. Even if homologues exist in other animals, the human versions are so far superior that they can be seen as truly unique and novel human traits.

The third hypothesis presented by HCF (2002: 1573) is that only FLN is uniquely human. According to this hypothesis, most of FLB is shared with non-human animals. The FLN, however, is recently evolved and unique to our species.

What is the role of recursion in language? HCF (2002: 1573) argue for the third hypothesis and state that "FLN comprises only the core computational mechanisms of recursion as they appear in narrow syntax and the mappings to the interfaces."

Referring to Diagram 1, lexical entries are found in the (FLB), and are combined by the recursive mechanisms found in the FLN to form phrases and sentences.

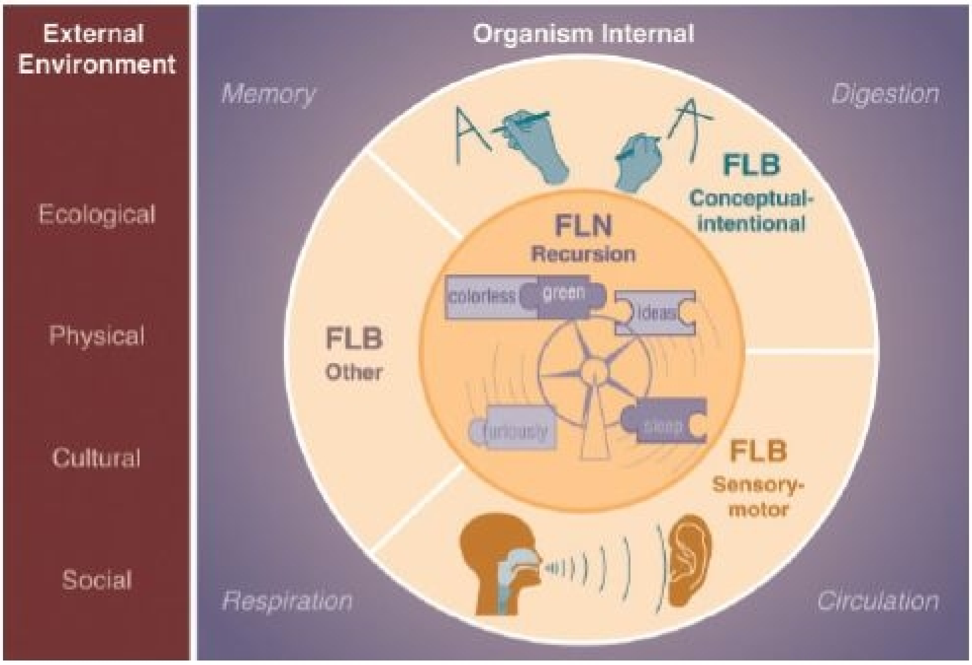


Diagram 1. Schematic of factors relating to the faculty of language. (HCF 2002: 1570)

The recursion-only hypothesis entails that the faculty of language in the narrow sense (FLN) comprises *only* of the core computational mechanisms of recursion. Note that this allows not only for structural recursion, in the form discussed by Radford (2004: 69), but also for functional recursion.

Functional recursion in linguistic theory and in the Minimalist Program (Chomsky 1995, 2000, 2007) takes the form of the operation Merge, which combines two syntactic objects to form a new one (Chomsky 2007: 3). The operation works recursively, combining all syntactic objects into increasingly larger structures, until the entire sentence is built.

The operation (or function), Merge, which creates syntactic objects by combining linguistic elements (words and phrases) is recursive, since every object created by Merge can again serve as the input to Merge (functional recursion). This type of recursion will be contrasted with the recursive properties of linguistic structures *created by Merge*, namely those in which nodes of a certain type are embedded within the same type of node (structural recursion*).*

Chomsky (2007: 8-26) defines Merge as follows: Merge(a,b) = {K, {a,b}}. The application of the Merge function to **a** and **b** is to create a syntactic object, {a, b} with K as the head (or label) of the newly created syntactic object.

1. Merge (a, b) = {K, {a, b}}

This set relationship can be depicted by means of a tree diagram, like in Diagram 1.

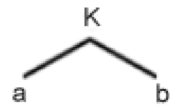


Diagram 1: Merge (a, b) - (Gallejo (2010: 10))

Merge is an operation, which can take outputs from previous Merge operations. In this manner, the output from Diagram 1 (the Merged terms **a** and **b**) can be used as the input to another Merge operation, depicted in Diagram 2.

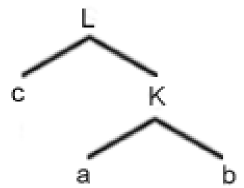


Diagram 2: 2nd application of Merge
(Adapted from Gallejo (2010: 13))

This process of Merging lexical items to form ever-larger syntactic structures can continue indefinitely to form sentences, which have no set upper bounds.

2.1 Recursion in other cognitive domains

Pinker and Jackendoff (PJ) (2004: 230) state that, “the only reason language *needs* to be recursive is because its function is to express recursive *thoughts*”.

Furthermore, Jackendoff and Pinker (2005: 217, hereafter JP) contest what they call the "recursion-only hypothesis", taking issue with idea that the FLN is unique to language, quoting the following from Fitch, Hauser, Chomsky (2005: 179-210):

"There are no unambiguous demonstrations of recursion in other human cognitive domains, with the only clear exceptions (mathematical formulas, computer programming) being clearly dependent on language."

JP (2005) and Jackendoff (2011: 9) argue that recursion is not uniquely part of FLN under HCF's definition and provide examples of recursion in other human cognitive domains, such as visual processing and music.

```
XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX
OOOOO OOOOO   OOOOO OOOOO   OOOOO OOOOO
XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX

XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX
OOOOO OOOOO   OOOOO OOOOO   OOOOO OOOOO
XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX

XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX
OOOOO OOOOO   OOOOO OOOOO   OOOOO OOOOO
XXXXX XXXXX   XXXXX XXXXX   XXXXX XXXXX
```

Diagram 9. A 2-dimensional array. (Jackendoff 2011:9)

In Diagram 9, the lines of x's and o's are combined first horizontally, then vertically to form larger and larger blocks. By providing this example, JP (2005) and Jackendoff (2011: 9) aim to present recursion (or at a minimum "grouping" or "embedding") in the visual domain and thus argue that recursion is not unique to language.

Jackendoff (2011: 10) writes that "structural recursion is not unique to language: the visual faculty contains a form of recursion considerably richer than what binary Merge can produce" and "if recursion occurs in the visual system as well, it *does* tell us something new about language, *pace* Chomsky, namely that recursion per se is not what makes language distinctive."

HCF propose that recursion is what makes human language different from animal language. Certainly, as JP (2005) and Jackendoff (2011) point out, recursion does occur elsewhere in our cognition, but this does not neccesarily entail that "recursion per se is not what makes language distinctive". Recursion could indeed be what makes human language distinctive, while appearing in other human cognitive domains.

JP (2005) and Jackendoff's (2011) examples of recursion (or embedding) in other cognitive domains are valuable in providing a fuller understanding of recursion, and so I will discuss two more of their examples. Jackendoff cites David Marr's (1982) theory of the 3D model in visual cognition, in which the body is understood in terms of its constituent structure: the torso is the head of the body, the palm is the head of the hand, and so on.

The structure of the human body might not be *recursive*, but it can indeed be understood in the structurally *embedded* manner in which Jackendoff describes it. Fingers are connected to a palm, and it we move the palm, the fingers move too. Similarly, the hand is a part of the lower arm, which is part of the arm, and so on.

We can see a clear embedded constituent structure in the same way as Russian dolls are nested within each other. The difference with analysing the body in this way is that Russian dolls are smaller versions of themselves (recursion), while the body consists of dissimilar parts, which combine to form a whole (embedding).

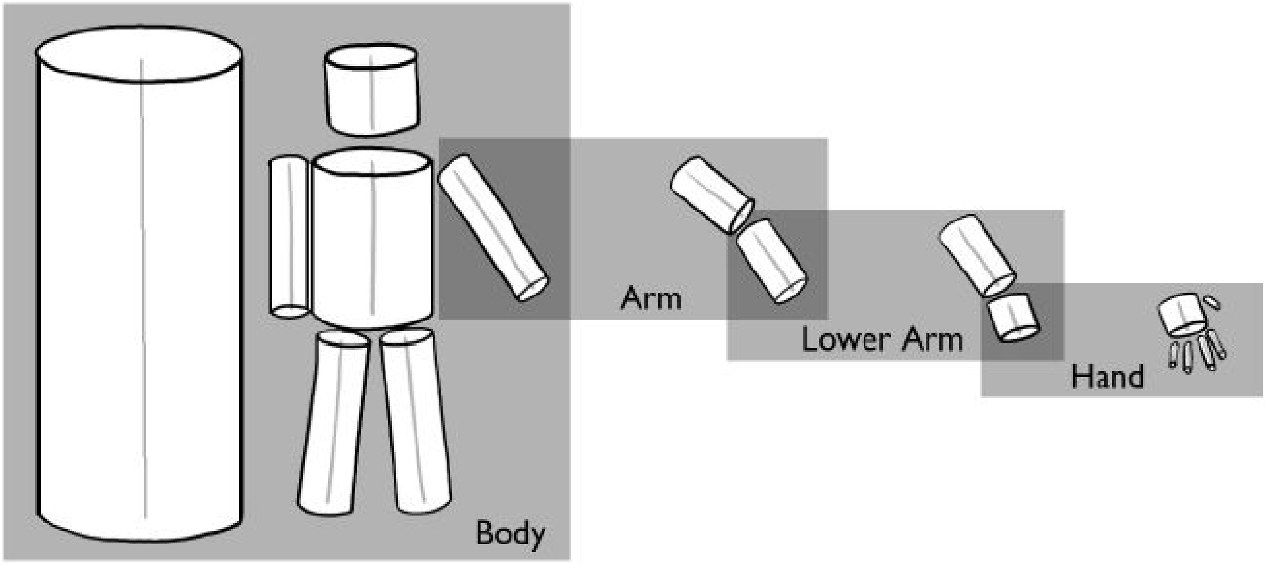


Diagram 10. Marr's (1982) 3D model structure for the human body. (from Jackendoff 2011:11)

Finally, Jackendoff (2011: 12) discusses recursion in music, using the song *Norwegian Wood* as an example:

The *grouping structure* below the notes encodes the phrasing of the music: the smallest groups consist of the music associated with the texts *I*; *once had a girl*; *or should I say*; and *she once had me*. These are grouped by pairs: *I once had a girl* and *or should I say she once had me*. These two groups form a larger group consisting of the whole fragment, which is paired with the second phrase of the song (*She showed me her room…*), and so on until the entire song forms a group. Groups can be embedded indefinitely deeply, all the way to 15-minute movements of symphonies; hence this domain is structurally *recursive.*

Jackendoff (2011: 12)

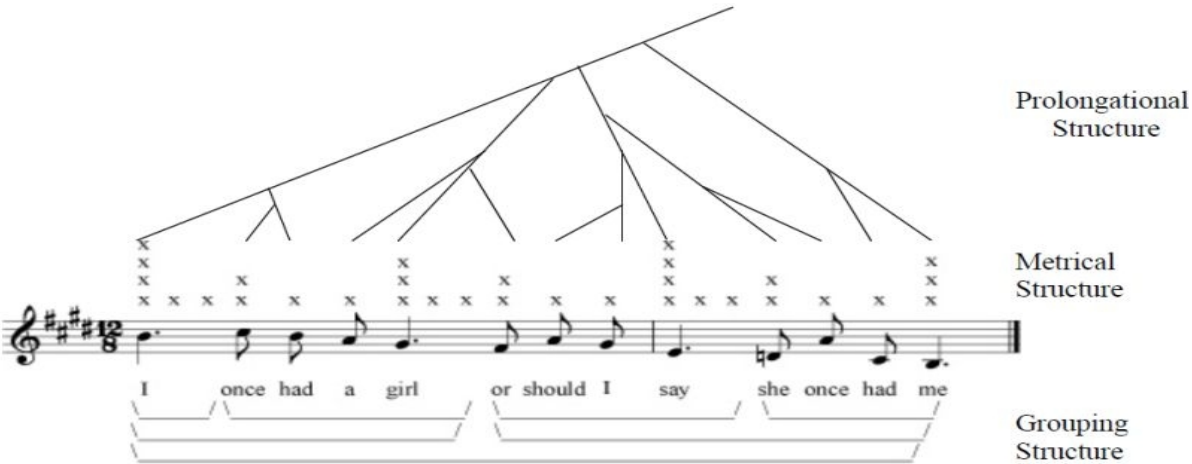


Diagram 11. Recursive structures in music. (Jackendoff 2011:12)

Notes are combined into phrases, and those phrases can be embedded to infinite depth to create larger structures of music, such as songs or symphonies.

PJ (2004: 230), JP (2005: 217) and Jackendoff (2011: 9-12) aim to undermine a part of HCF's recursion-only hypothesis, attempting to show that recursion is not a unique feature of the FLN, but occurs elsewhere in human cognition. They present examples of recursion in other human cognitive domains, saying that language only needs to be recursive, in order to express recursive thoughts.

Pinker and Jackendoff's criticisms of HCF's recursion-only hypothesis, however, do not seem to put the hypothesis at risk. Even if recursion is found elsewhere in human cognition, FLN could still have as its only component a recursive function in the form of Merge.

Returning to language, there are still further forms of recursion, different to sentential recursion, which are not found inside sentences, but are instead found in discourse. By way of defining "discourse", Radford (2004: 448) notes the following:

"...discourse factors are factors relating to the extrasentential setting in which an expression occurs (where extrasentential means 'outside the immediate sentence containing the relevant expression'). For example, to say that the reference of PRO is discourse-determined in a sentence such as 'It would be wise PRO to prepare yourself for the worst' means that PRO has no antecedent within the sentence immediately containing it, but rather refers to some individual(s) outside the sentence (in this case, the person being spoken to). (Radford 2004: 448)

In the light of Radford's explanation above, *discourse* is extrasentential, meaning that the anaphor points outside the uttered sentence, to something in the world (or to something elsewhere in the text or discussion). In this more traditional analysis, the domain of syntax is *within* the sentence, and the domain of discourse is *outside* the sentence.

Thus far, we have examined structural recursion (in the form of embedding) and functional recursion (in the form of the recursive function, Merge) *within* sentences. Koschmann (2010: 2691-2694) provides examples of an additional type of recursion, which is unique to discourse, involving two speakers.

1. A: Did you put the garbage out?
   B: Did I put out the garbage?
   A: Yes.
   B: Yes.

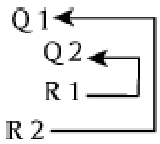


Diagram 1. Embedded Questions and Answers

In (1), there is an exchange between two speakers where the response to the second question is nested inside another question-response pair. Two questions are asked, one after the other, and then two answers are given. This type of embedding is different to the types of recursive embedding within phrases or sentences that we have discussed thus far, because it involves two speakers.

2. A: Hey did you talk Marcia into coming down here?
   B: Was she here?
   A: Yeah.
   B: When did she leave?
   A: About half an hour ago.
   B: Yeah, I talked her into living here with me.

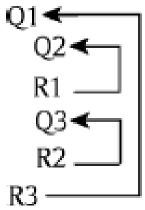


Diagram 2. Embedded Questions and Answers

Example (2) shows two question and answer pairs nested inside a question and answer pair. This type of recursion occurs extrasententially, due to the nature of the question answer pairs and the interactions between the two speakers.

Examples (1) and (2) from Koschmann (2010: 2691-2694) show recursively structured dialogues, showing that recursion can play a role in structuring discourse.

Koschmann (2010: 4) cites Roeper (2007: 110-111) to provide the example (3) below, in which we can see evidence of yet another type of recursion — structural discourse recursion that involves meanings from sentences embedded inside meanings from other sentences to create a larger whole.

3. The earth is flat.
The textbook said that.
Mary told Bill that.
The teacher didn't believe that.

In (3), each occurence of *that* refers to the previous text: the first *that* refers to the first sentence, the second *that* refers to the previous two sentences, and the final *that* refers to all three of the preceding sentences. In example (3), then, we have an instance where references are spread across four sentences, and with each sentence, the reference of each 'that' grows to include the previous references.

(1) and (2) are examples of recursive interactions between speakers, where questions and answer pairs are embedded inside other question and answer pairs. (3) is an example of a complex, recursive thought, which is expressed in discourse.

In this chapter, we have examined HCF's recursion-only hypothesis, by which HCF state that only FLN is uniquely human, and furthermore, that FLN consists only of recursion. We also considered some examples of recursion in visual cognition (JP 2005, Jackendoff 2011: 9), music (Jackendoff 2011: 12), and in discourse (Koschmann 2010: 2691-2694).

In the following chapter (Chapter 3), I will examine the case made by Everett (2005, 2007, 2009, 2010) against HCF's recursion-only hypothesis.

Chapter 3: No Recursion in Pirahã?

Everett (2005, 2007, 2009, 2010) provides a strong argument against the recursion-only hypothesis, and attempts to do so also for the recursive operation, Merge. Based on his study of Pirahã (a language spoken by a community of 300 to 400 people, living by the Maici River in Amazonas, Brazil), Everett states that "no evidence for recursion in the syntax, semantics or phonology of Pirahã" (Everett 2010: 8).

Everett's argument is that, since the "recursion-only hypothesis" is intended to apply to the human faculty of language (and so all human languages) finding evidence that there is no recursion in even just one language (Pirahã) nullifies the recursion-only hypothesis.

Everett (2005, 2007, 2009, 2010) challenges HCF's recursion-only hypothesis directly, stating that the language Pirahã poses a counter-example to the hypothesis. Everett (2010: 10) states the evidence from Pirahã as follows:

a. There are no long-distance hierarchical relationships in Pirahã sentences (but Everett notes that these relationships are found in Pirahã discourse).
b. Pirahã sentences do have upper bounds (and so are not infinite as per HCF)
c. Pirahã has no evidence of embedded phrases within phrases.

Everett (2010: 12) expands point c. by listing several examples from the Pirahã data, which according to Everett provide evidence that Pirahã indeed lacks recursive structure, i.e. structures with embedded phrases within phrases. I illustrate these examples, and the ensuing debate between Everett and Nevins, Pesetsky and Rodrigues (2007, 2009), proponents of HCF's recursion-only hypothesis, in the following sections.

3.1 Restricted Reference

Everett (2010: 12, and 2009: 11) states that Pirahã has no "restricted reference", or relative clauses, and that 1a), b), c) and d) are four separate sentences. The sentences translate to something like "I want a (Brazilian) hammock like the one that Chico sold*."*

1a) *Ti baósaápisí xoog -abagaí.*
1 hammock want almost
'I want a hammock.'

b) *Xigi -ábií xaoói.*
Associated remain foreigner
'I am like a Brazilian.'

c) *Chico hi goó baósaápisí bagá -boí.*
name 3 (third person) focus hammock sell -away
'Chico sold a hammock'

d) *Baósaápisí xais -igí -ai.*
hammock same -associated -be
'It is the same hammock.'

According to Everett (2005: 634), the Pirahã do not identify themselves as "Brazilians", but rather refer to themselves as "Pirahã", and to Brazilians as "foreigners". In the short discourse in 1a) - 1d), the speaker refers to a hammock, then qualifies his/her desire for the hammock as being like that of a foreigner.

In 1c) and 1d), the type of hammock is further described to the listener as being like the hammock that Chico sold.

The Pirahã construction in (1) requires four sentences to contain all the referents, which can be contained in a single English sentence ("I want a hammock like the one Chico sold."). Everett's (2010: 12) statement is that Pirahã has no grammatical equivalent to (a), since Pirahã lacks embedding.

Nevins, Pesetsky and Rodrigues (NPR 2007: 35), make the claim that Everett, in an earlier publication (Everett 1986: 281, hereafter HAL), supplies an example that provides different evidence. Based on the sentence below, it would appear that Pirahã speakers are indeed capable of producing sentences that contain relative clauses:

2) ti baósaápisi og- abagaí gíxai
1 hammock want -Frustrated INIT 2

[go- ó baósaápisi big -áo -b -í -i xai sigíái]
WH-OBL hammock show -TELIC-PERF-PROX-COMPLETE CERT be same

"I want the same hammock which you just showed me."
[more lit. What/That very hammock you just showed me, I want the hammock.] [HAL (281)]

NPR (2007: 35) cite another of Everett's (2005: 630 (33)) Pirahã examples below:

3) ti baósa -ápisí ʼogabagaí. Chico hi goó bag -áoba
I cloth -arm want. name he what sell -completive

(Everett 2005: 630 (33))

NPR (2007: 35) quote Everett's (HAL) analysis of the example:

"The two sentences are connected contextually, but this is not embedding. Each is an independent, well-formed sentence. The second sentence, on its own, would be a question, 'What did Chico sell?' In this context, however, it is the co-relative."

NPR take issue with the last sentence from Everett (2005: 630 (33)). It would appear that, first, Everett presents two separate sentences in his example, but in his analysis, he refers to the second sentence as a "co-relative", which would imply that the clause is embedded.

Everett (2007: 17) replies on this point, that the word *goo* has the meaning of "that there" and has a deictic function, to refer to old information. Thus, in the first sentence, the speaker refers to the "baósa -ápisí" by name, and in the second sentence, *goo* is employed to refer (in a deictic manner, with the speaker in physical proximity to the item) to the hammock.

NPR (2009: 675) point to Everett's (2007: 17) new claim that the element *goo*, is not a WH-form (information question) at all, and is instead a focus marker. NPR conclude that this new statement by Everett undermines his previous statements (in Everett 2005) about the lack of embedding in Pirahã.

3.2 Recursive Possession

The next example discussed by Everett (2010: 12) in support of his claim that Pirahã lacks recursion is illustrated below, in examples 4a-c).

4a) John's brother's house.
b) * Xaikáibaí xahaigí kaiíi xaisigíai.
c) Xahaigí kaiíi xáagahá. Xaikáibaí xahaigí xaoxaagá. Xahaigi xaisigíai.
'Brother's house. John has a brother. It is the same one.'

According to Everett (2010: 12), in Pirahã, the construction in (b) is impossible, where all the referents (John, brother, house) are contained in one sentence. Instead, in Pirahã, the referents are spread across three sentences.

NPR (2007: 11) provide examples from English and German, which show, first, that both English and German have possessive forms, but that (Saxon) German lacks a recursive possessive form:

5a) John's car	(English)
b) Hans-ens Auto	(German)

6a) John's car's motor	(English)
b) *[Hans-ens Auto]-s Motor	(German)

The argument that NPR attempt to build (here, and in the discussion of other examples in NPR 2007), is that Pirahã displays behaviour similar to other languages, and thus, that Pirahã is not as exceptional as Everett (2005) claims.

Everett (2007: 9-10) replies that while the particular form (-s) mentioned by NPR is not ideal in German (Everett 2007: 9 provides some marginal examples like "Peters Vaters Auto"), that there is another possessive form (von), which is fully recursive, as per the examples below.

7a) von dem Vater von meiner Frau
by/from the father of my wife

b) der Liedertext von der Mutter von James Bond von Elsterglanz
the lyrics by the mother of James Bond of Elsterglanz

c) zur Lage *von der Sprache von Peter Kohler
on the state of the language by Peter Kohler

I have altered the case marking from Everett's examples to be more correct. Also, Everett's (2007: 9-10) example 7c) is controversial, because the first *von* is not really grammatical.

Everett (2007: 10) states that "there is no conventionalized route for Pirahã speakers to express recursive possession", and then attempts to show that German does have such a conventionalized route, and thus in the case of recursive possession, that German is much more like English, than it is like Pirahã.

3.3 Recursive Imperatives

Everett (2009: 7) continues, stating that Pirahã contains no "recursive imperatives". NPR (2007: 14), however, point to examples in Everett (1986: 263), which interpret the *-sai* element as a nominalizer in embedded complement clauses, noting the following:

“The morpheme typically found on the verb in Pirahã embedded complement clauses is *-sai*, glossed as 'NOMLZR' (nominalizer) by Everett. It looks more nominal than other verbs in lacking the ability to show tense and aspect distinctions.”

Furthermore, according to NPR (2007: 14-15), it is "common across languages for embedded clauses to look more 'nominal'". NPR examine Everett's analysis of the *-sai* morpheme as a "Nominalizer" (or NOMLZR as Everett writes it in the glosses of his examples) and come to the conclusion that Pirahã is like other languages in this regard.

The existence of embedded complement clauses in Pirahã would be considered evidence of recursion. The following is one of the examples that NPR highlight:

8a). hi ob - áaxáí [kahaí kai - sai]
3 see/know-INTNS arrow make -NOMLZR

'He really knows how to make arrows.' (Everett 1986: 263)

Everett (2009: 7) states that while in many languages, speech acts like "order" take an embedded complement, it is has to determined whether Pirahã is a language that has a verb like "to speak", which takes no complement, or only an NP complement, or has a verb like "to say", which takes a sentential complement.

Everett uses the scope of negation in the following example to show that -*sai* is not a nominalizer:

9a) I am not ordering you to make an arrow.
b) Ti xibíibihiabiigá. Kahaí kaisai.
'I am not ordering you. (We are speaking of) (you) make an arrow.'

Everett (2009: 7) replies to NPR's analysis, saying that, "NP&R are mistaken. There is no imperative in the second clause and the scope of negation is mono-clausal." He goes on to claim that the translation should rather be something like "I am not ordering you. You make the/an arrow(s)."

Everett (2009: 7) thus presents an example that consists of two sentences, instead of one sentence with an embedded complement clause as per the examples that NPR were reviewing in Everett 1986. Everett (2009: 7) goes on to state that -*sai* is not a nominalizer (as he claimed in Everett 1986), but rather a marker of old information, saying that if "arrow-making" has already been discussed in the discourse, -*sai* might (but not necessarily) be attached.

According to Everett (2009: 7), depending on the context and the preceding information, -*sai* can appear in none, one or both clauses, and this would indicate that while -*sai*-marked sentences might look nominalized, they are not. Everett continues by stating that -*sai* can also appear on conditional sentences, nouns and declarative sentences, where its purpose is to mark old information in the discourse. What this entails is that Everett's earlier (1986) analysis of example (8a) as one sentence, with embedding, has to be re-analysed as (9b) – in two sentences.

With the Pirahã examples above, Everett (2010: 12-28) aims to show that embedded phrases are not possible in Pirahã. Instead, Pirahã has to employ several sentences, which are possible in English in a single embedded sentence.

Everett (2010: 21) provides another example, which he calls 'recursive modification', or multiple modification. He states that trying to spell out recursive (or multiple) modification in one sentence is ungrammatical in Pirahã, and that, in fact, three sentences are required to articulate what in English is the phrase 'two big red barrels':

10a) Kabogaohoi biisai. Kabogao xogii piaii. Kabogaohoi hoihio piaii.
'Red barrel. Barrel is also big. A relatively larger quantity of barrels too.'

Everett (2010) continues, stating that Pirahã lacks other mechanisms, which might require embedding, such as recursive quotatives ("He says there is no child here"), temporal clauses ("When I finish here, I want to talk to you") and WH-questions ('What does he know how to make?').

Finally, Everett (2010) points out that Pirahã lacks coordinating particles, such as *and* and *or*, which might point to the construction of embedded sentences if they were found in the language. Embedded sentences are a type of structural recursion, since sentences are contained within sentences, and co-ordinating particles might be associated with this kind of construction.

3.4 Sauerland's study of the *sai*/*saí* morphemes

Sauerland (2010: 1-10) uses speech analysis software to show that several Pirahã speakers' use of the *sai* morpheme can be seen as evidence of embedding in Pirahã. Sauerland's study provides contrary evidence to Everett's examples (8), (9) and (10) above, which all contain the *sai* morpheme.

Sauerland (2010) presents an analysis of Pirahã field data collected by Stapert, Everett, Frank, and Gibson in 2007. The data was originally presented by Stapert (2007). Sauerland (2010) specifically looks at the usage of *sai*/*saí* morphemes by the test subjects – an aspect of the data that was not considered by Stapert (2007).

In the study, Stapert (2007) conducted an experiment with nine Pirahã speakers using a sentence correction task. The nine Pirahã speakers took part in individual sessions, during which they were asked to repeat utterances that were presented to them by a Pirahã-English interpreter.

In some of the utterances, the word order had been shifted around, so that the utterances were ungrammatical. The Pirahã subject was instructed to repeat the interpreter's utterances, but to make any corrections in order to make the utterance grammatical.

Each session was audio-recorded in its entirety and Sauerland (2010: 6) performed pitch extraction with the PRAAT software and then computed the maximum pitch on the syllable *sai*. This study shows that the native Pirahã speakers pronounced conditional *saí* with lower pitch than nominalizer *sai*. Sauerland (2010) claims that Everett proposes three *different* analyses of *sai*/*saí* morphemes in his 1986, 2005 and 2009 publications.

According to Everett's 1986 analysis, both versions of *sai* are used to connect sentences. The tone of *sai* depends on the role the *sai*-marked clause has in the embedding sentence. Everett's 1986 analysis predicts an obligatory tone difference, but in the opposite direction to Sauerland's (2010) findings.

According to Sauerland (2010: 8-9) Everett's 2005 analysis predicts *no tonal difference* between the two *sai* morphemes, while Everett's 2009 analysis allows for an optional tone difference – though in the opposite direction from the one observed by Sauerland (2010).

Sauerland (2010: 6) focuses on the questions given during the tests (including those presented in examples (1) and (2)), and states that in (1), the sentence in which *sai* occurs could be a nominal argument or adjunct of the preceding clause (if the verb xobáaxái ('good at') can combine with an extra nominal).

1) Hiaitíihi	xobáaxái	kaíi	kai-sai
Pirahã	good at	house	make-SAI

'The Pirahã are good at making houses.'

‘The Pirahã are good. They make houses.’

2) Pi-boi-bai-sai ti kahápihiaba
rain-MOVE: DOWN- INTNS-SAI 1 go-NEG

‘If it rains, I’m not going.’
‘It is raining. I’m not going.’

Sauerland (2010: 6) points out that in (2), however, the *sai* clause precedes a second clause which has the verb *kahápihiaba* (‘going’). Because of the difference in the verb's position, an analysis as conditional could be pragmatically more plausible in (2).

The results of Sauerland's study suggest two possible hypotheses, firstly, a lexical explanation: there are two morphemes, conditional *saí* and nominalizer *sai*, each distinguished by lexical tone, and secondly, a prosodic explanation whereby sentence prosody could explain the observed tonal distinction, since in the data *sai* (the nominalizer) always occurred at the end of the utterance, while saí (the conditional, as noted by the accented í; a morpheme with a different tone to *sai*) occurred in the middle of the utterance.

In the prosodic hypothesis, it would be possible to postulate that there is only one morpheme *sai*/*saí* but it is pronounced with low tone if it occurs in the middle of a complex clause and with high tone if it occurs on the right edge of a complex clause.

Everett (2009: 7, 2010: 21) makes use of examples (9) and (10) attempting to show (contra NPR) that sai is a marker of old information (and not a "nominalizer"), to further his claim that there is no recursion in Pirahã.

However, in both possible conclusions reached in Sauerland (2010), the data appears to support the existence of complex syntax in Pirahã, which is consistent with the presence of embedding and thus recursion.

3.5 Structural and Functional Recursion

There is another aspect in which Everett's claims fall down, with regards structural and functional recursion. Everett (2005, 2007, 2009, 2010) makes the claim that there is no embedding *at all* in Pirahã and that Pirahã is a language, which lacks linguistic structures like recursive possession or modification, as well as coordinating particles, which are traditionally used to combine sentences into larger sentences, which to speakers of languages like English are basic building blocks of language.

Referring back to the definitions of recursion in Chapter 1, Everett (2005, 2007, 2009) refers to the type of recursion that Radford (2004: 70) describes in Diagram 3, structural recursion, where a sentence is structurally embedded within another sentence.

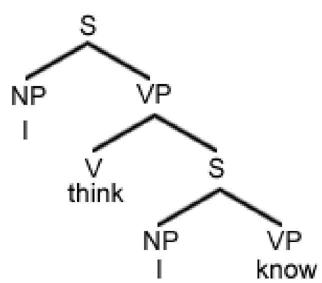


Diagram 3: An embedded S

Structural recursion (as per Diagram 3.) is a visible type of recursion, where an element (such as the S in Diagram 3.) is embedded inside a similar type of element.

However, as was discussed in Chapter 1, there is another type of recursion that is relevant for the generation of linguistic structures, which is different from *structural recursion*, namely *functional recursion.* It is this type of recursion that HCF refer to in their recursion-only hypothesis. HCF's recursion-only hypothesis leaves open the possibility for *functional recursion,* in the form of the recursive function Merge, whereby words are combined into phrases and then sentences.

NPR (2007) make the distinction between functional and structural recursion in their criticism of Everett (2005), pointing out that a language without embedding may still generate phrase structures through the recursive operation Merge. In his reply to NPR, Everett (2009: 36) states the following:

"NP&R try to remove the force of my criticisms of Hauser, Chomsky, and Fitch's (HC&F) (2002) proposal on recursion by arguing that what HC&F meant was not the very specific definition of recursion that I offer, but 'Merge' is [sic] a Minimalism-internal subtype of recursion. They argue that because I missed this, my criticisms do not go through. But whether HC&F meant Merge or not is immaterial to my criticism of their proposal. Merge fares no better nor worse in relation to the facts of Pirahã than recursion more generally." (Everett 2009: 36)

Everett makes the claim that “Pirahã lacks recursion” (Everett 2007: 7), and attempts to go on to make the further claim (Everett 2010: 11) that Pirahã lacks *all* types of recursion, both structural and functional. While Everett’s research appears to be able to show that there is no structural recursion in Pirahã, it fails to do so conclusively for functional recursion in the form of Merge.

3.6 Maximum Sentence Length

HCF (2002: 1571) state that “FLN takes a finite set of elements and yields a potentially infinite array of discrete expressions.” HCF go on to say that “there is no longest sentence (any candidate sentence can be trumped by, for example, embedding it in “Mary thinks that . . .”), and there is no non-arbitrary upper bound to sentence length.”

When NPR (2007: 9-10) introduce the recursive function Merge to the discussion, Everett (2009: 36) replies that “NP&R would have the reader believe that if there is no Merge in Pirahã then Pirahã sentences can have no more than two words.”

Everett (2009a: 233-240) provides the sentence “Xahoapióxio xigihí toioxaagá hi kabatií xogií xi mahaháihiigí xiboiíopí piohoaó, hoíhio” (Another day an old man slowly butchered big tapirs by the side of the river, two of them) saying that this sentence cannot be added to in any way to make it longer, without also making the sentence ungrammatical.

Everett uses this sentence to conclude that “Pirahã is finite and cannot be recursive”. According to Everett, and contra to HCF, it appears that the sentence cannot be added to by embedding the sentence in “Mary thinks that...”, which supports the conclusion that there are indeed limits to the lengths of sentences in Pirahã.

Everett (2009a: 239-240) makes the analogy between Pirahã and chess saying the following:

“Think of something like chess, which has also got a finite number of moves. This finitude of chess moves doesn’t have much of a practical effect, though. Chess is an enormously productive game...” (Everett 2009a: 239-240)

However, it does not follow from Everett's 'longest sentence' and chess analogy that Pirahã 'cannot be recursive'. On the contrary, NPR's argument is that Pirahã is not as exceptional as Everett claims. Furthermore, the application of recursive function, Merge to Pirahã means that all Pirahã sentences are made up through recursively embedded structures.

Pirahã is clearly able to express recursive thoughts and relations through its discourse, that is, several sentences that are joined together. According to Everett (2005, 2007, 2009, 2010), the nature of this "joining" is not embedded, and thus recursive. NPR's argument, which I find more convincing, is that Pirahã's sentences are joined together to form embedded, recursive structures.

In Chapter 4, I will examine in more detail how groups of Pirahã sentences "join together".

Chapter 4: Parataxis versus Recursion

In the last chapter, we examined Everett's (2005, 2007, 2009, 2010) case against HCF's recursion-only hypothesis. Instead, Everett (2007: 8) puts forward what he calls a "no-recursion hypothesis": that the Pirahã language makes use of *no recursion* (contrary to HCF's recursion-only hypothesis).

Everett (2007: 8) lists linguistic phenomena, which are absent from Pirahã and which might support recursion or embedding:

a. No numerals or counting (since these are built on recursion, as per Wiese (2003))
b. No auxiliary verbs
c. No stacked possessors.
d. No conjunction.
e. No disjunction.

Everett (2010: 9) concludes his argument against Chomsky's Minimalist Program by saying:

"If the infinity of language turned out to reside in discourse, rather than the sentence grammar, then Minimalism would be unable to express this. That is because its only form-creating operation is Merge and Merge only forms sentences and phrases from lexical items."

Additionally, consider the statement by Jackendoff (2011:16):

"...verbal narratives typically contain multiple sentences that are not connected syntactically. This means that the structure of verbal narratives cannot arise through Merge in syntax, since syntax is limited to the structure of single sentences."

Both authors stress that Merge is limited to applying *within* the sentence, and claim that Merge operates only on *syntactic objects.* They assume that Merge is incapable of operating on syntactic objects that occur outside sentences (what I term *discourse objects*).

On this matter, I strongly disagree with both Everett and Jackendoff. My hypothesis is that the recursive function Merge, in addition to being able to combine words and phrases, is able to combine clausal structures to form larger syntactic structures that make up a discourse.

Thus, in my view, the recursive function Merge remains intact, with its full explanatory power, and is simply extended to also allow for the Merging of sentences. Table 1 below summarises three subtly different (but not incompatible) views of the role of recursion in language:

	HCF	Everett (Pirahã)	This thesis
Sentential Recursion	Recursion-only Hypothesis	Pirahã has no sentential recursion	Recursion-only Hypothesis
Extra-Sentential Recursion	n/a	Pirahã expresses recursive thoughts through discourse	Merge also applies when joining clauses.

Table 1. Comparison of the roles of recursion in language.

Everett: (2005: 635) claims that Pirahã sentences are added together paratactically like 'beads on a string' to create meaning. What is possible in a single sentence in English, would need to be expressed by means of several linked sentences in Pirahã.

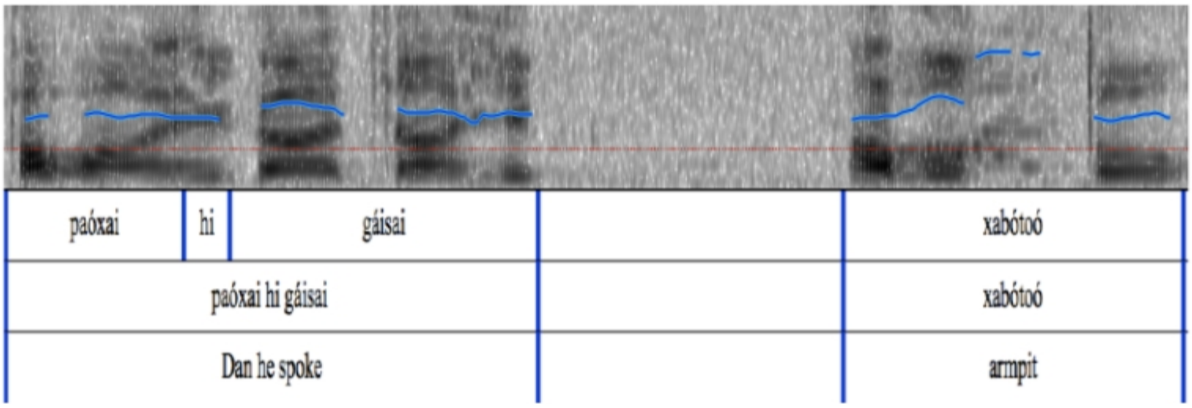


Diagram 1. (Everett 2010: 15)

1) Paóxai hi gáisai xabótoó
Dan he spoke armpit

Diagram 1. above, depicts the intonational chart for sentence in (1), which according to Everett (2010: 15) can be translated freely into English as "*Dan says (that it is called an) armpit"*, or "*Dan says, 'Armpit'".* Diagram 1. provides an example of a gap between two clauses.

Everett argues that this type of example cannot be expressed in an embedded form in Pirahã (as it is in English), due to the intonational gap and so Diagram 1. must be expressed as two sentences in Pirahã – in this case, like two beads on a string.

Mark Liberman (2006) speculates on the nature of these "beads" or mini-sentences that Everett reports from his analysis of Pirahã. Liberman presents two possible ways that the mini-sentences could be joined together: in the first scenario, they could be embedded recursively, in a hierarchical fashion, or secondly, they could be appended in the way that Everett (2007, 2009, 2010) describes Pirahã sentences, in a flat, non-hierarchical fashion like 'beads on a string'.

Liberman (2006) provides the example below from Elmore Leonard's *LaBrava.* The style of English spoken is like that of 'wiseguys' or Italian immigrants to the USA:

"What're you having, conch? You ever see it they take it out of the shell? You wouldn't eat it."

Liberman (2006) provides the following diagram of what the structure of the (recursively embedded) sentence would look like in standard English if it were supplied the missing conjunctions, *if* and *when*.

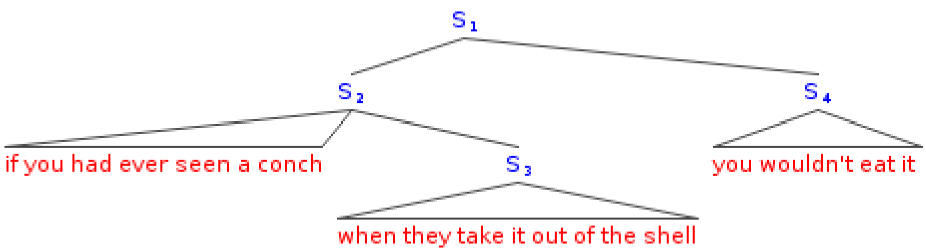


Diagram 2. (Liberman 2006)

A possibility that Liberman discusses is that the sentence group might have a flat, non-hierarchical structure as per the diagram below. This flat, non-hierarchical structure is consistent with Everett's analysis of Pirahã's 'beads on a string' sentence groups.

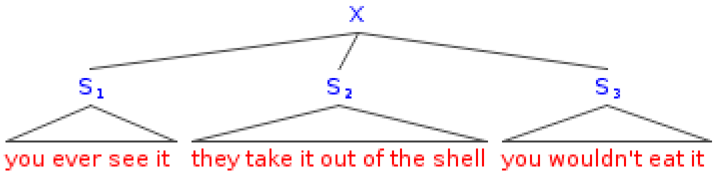


Diagram 3. (Liberman 2006)

Finally, Liberman raises the question of whether the sentence group can be structured as follows, in an embedded fashion. In this structure, the first two sentences are mutually dependent and the third sentence is an adjunct. This paper aims to show that this latter structure (Diagram 4) is the more accurate representation of the way in which sentences join in Pirahã.

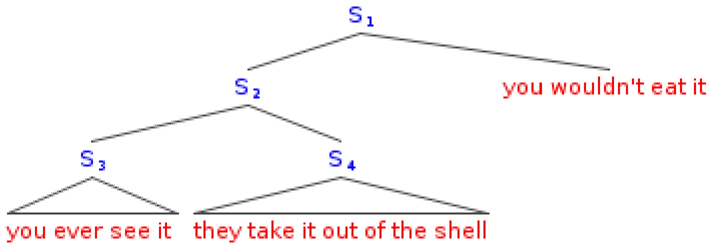


Diagram 4. (Liberman 2006)

Liberman cites Everett's (2005: 630) Pirahã example which might be analysed as a temporal clause embedded in a single sentence, but which Everett claims to be two separate (paratactically juxtaposed) sentences.

2) *kohoai*	*-kabáob*	*-áo*	*ti*	*gí*	*'ahoai*	*-soog*	*-abagaí*
Eat	finish	temporal	I	you	speak	desiderative	frustrated - initiation

When [I] finish eating, I want to speak to you.
(lit. "Eating finishes, I you speak-almost-want")

Everett (2005: 630) writes the following about the example:

"There is almost always a detectable pause between the temporal clause and the 'main clause'. Such clauses may look embedded from the English translation, but I see no evidence for such an analysis. Perhaps a better translation would be 'I finish eating, I speak to you.'" (Everett 2005:630)

Additionally, Everett (2009: 8) notes the following about parataxis:

"These are reasons to analyze examples of intersentential relations as parataxis in Pirahã. The basic form of evidence for parataxis and against embedding or recursion in the syntax comes from the semantic looseness of the connection between the two paratactic clauses." (Everett 2009:8)

In order to better understand the paratactic joining that Everett (2007, 2009, 2010) proposes, consider the definition of parataxis from The Oxford English Dictionary:

"The juxtaposition of clauses or phrases without the use of coordinating or subordinating conjunctions, as in *It was cold; the snows came*. (Retrieved from http://www.oxforddictionaries.com/definition/english/parataxis)

The key point from The Oxford English Dictionary definition is that clauses are joined without the use of conjunctions. Everett (2007, 2009, 2010) claims that Pirahã sentences are joined together without the use of conjunctions (and, in, fact as was pointed out above Pirahã has no conjunctions: Everett 2007: 8), which he then claims entails that Pirahã sentences are not embedded or recursive.

In order to better understand Everett's meaning, let us consider an example of parataxis from the English language. Lang (1987: 216) presents Julius Caesar's famous statement "I came, I saw, I conquered" as an example of parataxis – by which the three main clauses occur together in one sentence, without the use of any form of conjunction.

3a) I came. I saw. I conquered.
3b) I came, I saw, I conquered.

Julius Caesar's campaign in Gaul must have taken several months or years to complete – and indeed involved several distinct occasions of travel, surveying and conquering. In Julius Caesar's statement, parataxis combines the three actions in a single sentence, to convey a singular action.

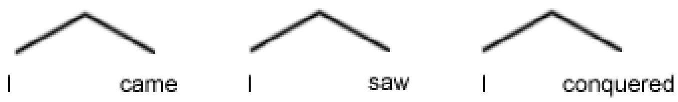


Diagram 5. “I came. I saw. I conquered.”

3a) uses three sentences to indicate three separate actions.

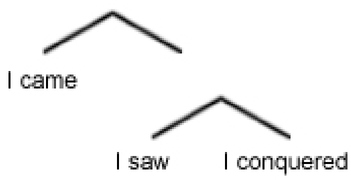


Diagram 6.

Example 3b. contains the three sentences, but this time paratactically joined (this is noted through the use of commas instead of periods), indicating the singular action of Julius Caesar's coming, seeing and conquering – at once (Lang 1987: 216).

The question posed by Liberman (2006) is whether the sentence in 3b) has a structure like the one in Diagram 5, or like the structure in Diagram 6.

Everett's (2007, 2009, 2010) approach is that paratactically joined sentences have a structure like that in Diagram 6, and are thus not embedded (and thus cannot be recursive). If, however, it turned out that the joined sentences have a structure like that in Diagram 7, then it follows that there is embedding and recursion in Pirahã.

4.1 Conjunctions and Coordination

One of the defining features of the parataxis phenomena in Pirahã is that sentences are joined without the use of conjunctions. Let us briefly examine the way in which conjunctions join sentences. Diagram 7. shows an example in English, in which two phrases are joined together by means of a coordinating conjunction *and* (Johannessen 1998; Kayne 1994; Radford 2004: 443).

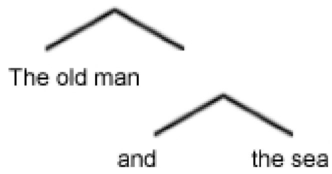


Diagram 7.

Conjunctions are used to combine complete sentences as well. In Diagram 8, two sentences are combined by the coordinating conjunction *and* again, the same coordinating conjunction that was used in Diagram 8.

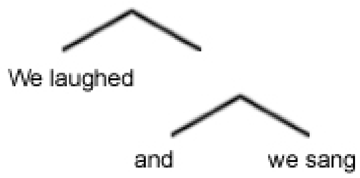
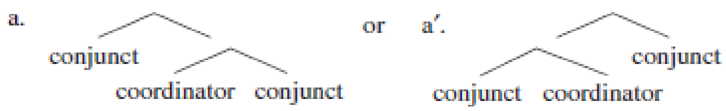


Diagram 8.

The structure of coordination is still a matter of debate in linguistics. Zhang (2010: 10) presents two different approaches to conjunction, one based on a binary syntax, in which the coordinator is the head of a phrase which takes one conjunct as a specifier and the other as a complement (see (9a) and (9a')), and one that assumes a flat structure, as in (9b):



Diagrams 9a and 9a'. Binary (hierachical) Conjunction

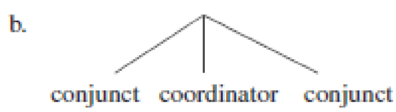


Diagram 9b. Ternary ("flat") Conjunction

Zhang shows, using examples like (4a) and (4b) that there is an asymmetry between conjuncts in binding. In (4a) the first conjunct "every man" can be the antecedent of a pronoun "his", but the reverse cannot be the case (as in (4b)). This demonstrates that the first conjunct asymmetrically c-commands the second conjunct, a structural relation that is adequately reflected by the syntax in (9a), but not by (9a') or (9b).

4a) Every man$_i$ and his$_i$ dog left.
4b) *His$_i$ dog and every man$_i$ left.

Similarly, Zhang (2010: 12) shows that there is an asymmetry between conjuncts in possessee pronominalization:

5a) Sally's mother and yours have turned vegetarian.
5b) *Yours and Sally's mother have turned vegetarian.

Zhang (2010: 13) concludes that the asymmetry in possessee pronominalization suggests that the first conjunct is structurally higher than the second conjunct. This hierarchical structure is an example of the binary conjunction as depicted in Diagram 9a.

Zhang's (2010) arguments support the hypothesis that the recursive function Merge is responsible for the creation of coordinate structures. This implies that Merge is also involved in the conjunction of clauses. The phenomenon of asymmetrical conjunction that Zhang (2010) describes is expressed structurally as one element embedded within the other. Because the one element is embedded within the other, and not vice versa, there is an asymmetry in the structure, hence the term "asymmetric conjunction".

4.2 Omitted Conjunctions

There are cases where two sentences are joined, but the conjunction is implies and not present. One such example is from Monthy Python's famous "Hungarian Phrase Book" skit. (Available at http://www.youtube.com/watch?v=2YYM209GJoE, accessed on 28 January 2014. The sentence (6a) occurs at 0:26)

6a) I will not buy this record; (...) it is scratched.
6b) I will not buy this record, *because* it is scratched.

Note that in (6a) there is a definite pause at the position indicated by the ellipsis (...), implying the presence of a conjunction. The resulting structure might look something like Diagram 10.

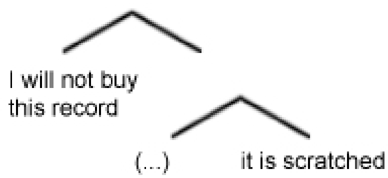


Diagram 10.

Example 6a) and Diagram 11 thus show that it is possible in English to join sentences without the use of a conjunction. It is quite possible that the same is true in Pirahã – that Pirahã can have recursively coordinated sentences, even without conjunctions.

4.3 Run-on Sentences

Another example of coordinate structures without conjunctions are so-called “run-on sentences”. A run-on sentence is a linguistic phenomenon which occurs in German, Swedish and to a lesser extent, English, where two complete sentences are joined together without the use of a conjunction. A famous example of a run-on sentence is found in the Beatles 1967 song, "Hello, Goodbye".

7a) You say goodbye, I say hello.
7b) You say goodbye. I say hello.
7c) You say goodbye and I say hello.

In English, the structure in 7a) is available sententially — all in one sentence, with the two clauses combined to form a larger sentence. From the example in Diagram 11 above, it seems likely that the structure of the sentences in 7a) and 7b) is as depicted in Diagram 12.

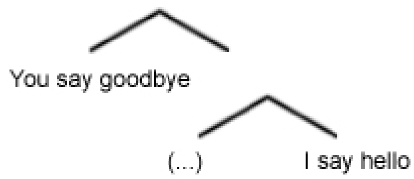


Diagram 11.

The position indicated by ellipsis in Diagram 11. can be filled by a conjunction, such as "and", or a null conjunction (as show in Diagram 11).

4.4 Reviewing Everett's case against HCF

The binary coordinate structures in Diagrams 7 to 11. take the form of "X and X”, where X can be a noun phrase or a sentence. In addition, it appears that (as per examples 6a and 7a) the coordinating element can be null.

This hierarchical, conjunctive structure can be extended indefinitely ([X and [X and [X and [X and X]]]] etc), with each "and X” further embedded within the larger structure. This is an example of the extendable, recursive nature of language, in line with HCF's recursion-only hypothesis.

Everett (2007, 2009, 2010) claims that Pirahã does not have the type of structure found in Diagrams 7 to 11, because of the fact that Pirahã has *no conjunction* (Everett: 2005: 631-632, 2007:8), and provides the following examples in support of his argument:

7a) *’ogi* *-a´ag* *-ao´* *toı´o*
big -be -thus old

-’aaga´ ’ igihı´ ’ *ipo´ihiı´* *pı´aii*
-be man woman also

“Everyone [lit “people bigness”] is old. Men and women too.”

7b) *ogia´* *agao´* *toı´o’aaga´ ’igihı´ toı´o’aaga´*
big old man old

’ipo´ *ihiı´* *toı´o’aaga´ pı´aii*
woman old also

“Everyone [lit “people bigness”] is old. Men and women too.”

If Pirahã has no conjunction, it could lend support to Everett's argument that there is *no recursion* in Pirahã. If there is no recursion in Pirahã, Everett's argument could provide a counter-example to HCF's hypothesis that recursion is the defining feature of all human faculties of language.

However, as I have shown in this chapter, it is indeed possible that Pirahã can have recursively coordinated sentences, even if it does not have conjunctions. Everett's argument for “flat, paratactic sentences joined like beads on a string” does not hold, and the far more compelling conclusion is that the clauses in Pirahã are joined via embedding, just like in other languages, simply without the use of conjunctions.

This conclusion, in turn, supports the view that Merge also applies to the conjunction of clausal structures, as the discourse in Piraha is formed via asymmetrical coordination without conjunctions, Merge plays a fundamental role in joining the clauses together, to form longer sentences and complex clausal structures.

Conclusion

First, in Chapter 1, I introduced and discussed the concept of recursion, providing several definitions and examples of recursion. Recursion can mean a structural nesting or embedding of objects one within the other, or it can refer to a process which can contain embedded processes, where the more embedded versions of the process need to complete first, in order to allow the larger process to complete. The Tower of Hanoi, the Fibonacci Spiral, and the Sierpinski Carpet are all examples of recursive processes in action.

Throughout the first and second chapters, we examined examples of recursion in different fields, and what recursion means in each of: mathematics, computer science, visual cognition, music and language. Furthermore, in Chapter 2, I introduced the topic of recursion in language as well as the debate surrounding the recursion-only hypothesis, which states that recursion is the “core computational mechanism”, which enables the generation of an infinite set of expressions in a language.

In Chapter 3, I reviewed Everett's “no-recursion” hypothesis as well as NPR and Sauerland's counter-arguments. Everett (1986) had analysed the *sai* morpheme in a way that made it look like a nominalizer. Everett (2005) re-examines his own analysis, and comes to a different conclusion — that there is no recursion in Pirahã. When NPR (2007: 14) question his analysis, Everett (2009: 7) reasserts that Pirahã has no recursion by providing a slightly modified argument. Finally, Sauerland (2010) provides a study using speech software, which makes use of Stapert's (2007) Pirahã field data. Sauerland (2010) focuses on the use of the *sai* morpheme and finds that there are two places that the *sai* morpheme is used, and both use cases point to evidence of complex complex syntax (and thus embedding and recursion) in Pirahã.

In Chapter 4, I examined Everett's (2005, 2007, 2009, 2010) claim that Pirahã sentences are joined in flat structures like “beads on a string”, via parataxis, since as Everett observes, Pirahã also has no conjunctions.

In section 4.1, I examined the way that conjunctions combine phrases in an embedded manner, which is known as asymmetric conjunction. The asymmetry comes about, because one element in the conjunction is embedded within the other, and not vice versa.

I put forward examples in section 4.2, of omitted conjunctions and in section 4.3. of run-on sentences, which show examples of asymmetric conjunction which are possible, without the presence of any conjunctions.

I examined the way in which sentences in other languages are able to join, without the use of conjunctions and came to the conclusion that Pirahã sentences could well be joined in a similarly embedded manner, via asymmetrical coordination.

What this entails is that discourse in the form of complex sentences (ie: sentences containing sentences) in Pirahã could be formed by the application of the recursive function, Merge, which in turn provides support for my own hypothesis: that Merge also applies to the conjunction of clausal structures in language.

Further Thoughts: Clause Chaining

I would like to open the possibility for further consideration that the sentence-conjoining phenomenon that Everett (2005, 2007, 2009, 2010) describes in Pirahã, could be a type of clause chaining. Clause chaining is a phenomenon that occurs in other languages, in which clauses are combined together to form a larger, complex sentence. Superficially at least, this appears similar to what Everett describes in Pirahã.

Dooley (2010) presents properties of the prototypical clause chaining construction:

> (A) Each clause is individually asserted and advances the timeline of the discourse.
>
> (B) Though only one peripheral clause is inflected, all other clauses are interpreted as if identically inflected, and may furthermore be marked for switch-reference (e.g. same or different subject as the preceding clause).

The properties (A) and (B) are testable, and their potential applicability to Pirahã can be verified by further study.

Dooley (2010: 3-18) continues by providing an overview of clause chaining and identifies three types of clauses that can occur in clause chains:

1. *Independent clauses* can stand on their own, optionally without the need for supporting clauses.

2. *Foreground clauses* consist of asserted sequential events, and assert the main stream of information.

Sequences of foreground clauses have certain predictable features, including those which comprise “quasi-coordination”.

3. *Background clauses* do not narrate, but instead support, amplify or comment on the narration.

According to Dooley and Levinsohn (2001:12), background clauses, are "linguistic means to signal coherence", and are employed as a type of cohesion. They are not required by chaining, but commonly occur as adjuncts in sentences of various types. Clause-chaining sentences thus typically contain an independent clause, optionally a sequence of foreground sentences, which may in turn optionally be modified by background clauses.

Dooley (2010) provides example (3) below (from Waltz 1976: 31, 125) in order to explain the distinction between background and foreground clauses:

> 3) Guanano (Tucano, SOV, Colombia and Brazil)
>
> a) *Buha,* b) *co wa,* c) *wuh p tjuata,* d) *sãa wahahi.*
> go.to.river water get to.house return enter I.went
>
> ‘a) Going to the river, b) getting water, c) returning home, d) I went and entered.’

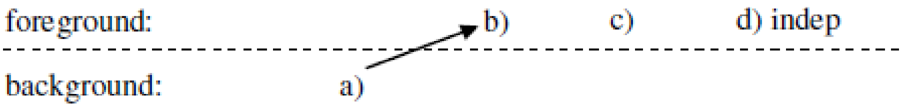


Diagram 1.

In (3), clauses a) - c) are all dependent on the independent clause d). The arrow shows that a) modifies b). A background clause usually modifies an adjacent clause, like in example (3), where a) modifies b, and not independent clause d). An initial background clause, however, may serve to introduce an entire discourse unit, not just the clause that it specifically modifies.

4) *Kanite* (Trans-New Guinea, SOV, Papua New Guinea; Longacre 2007:401-402):

a) *his-u'a-ke-'ka* b) *naki a'nemo-ka hoya ali-'ka*
do-1PL-DS-2PL so woman-2PL garden work-2PL

c) *naki ali ha'anoma hu-ne'atale-'ka* d) *popo hu-'ka* e) *inuna kae-'ka*
so work finish do-COMPL-2PL hoe do-2PL weeds burn-2PL

f) *naki ha'no hu-talete-ke-ta'a*
so finish do-COMPL-DS-1PL

g) *naki viemoka-ta'a keki'yamo'ma ha'noma ne-his-i-ana*
so man-1PL fence finish FUT-do-3-CONJ

a) If we do this, b) you women work the garden,
c) when it is finished, d) hoe e) and burn the weeds,
f) when that is finished, g) we men will finish making the fence.

Example 4) has three "mini-utterances", a) - b), c) - d) - e), f) - g), each containing a background orientation clause followed by foreground.

According to Dooley (2010: 5), clause a) is conditional ("if we do this"). c) and f) are temporal, expressing a sense of time ("when"). a), c) and f) are all background clauses of orientation. d) and e) are foreground clauses, and appear to be coordinate.

Diagram 2.

The sentence in (4) above demonstrates how long clausal chains can be constructed.

Asymmetric Clausal Coordination

Rafael Nanoto (2013) asserts that “clause chaining” (as discussed by Dooley (2010) and the authors cited by Dooley 2010) is a “sui generis construction”, meaning that authors tend to attribute properties to the clause chaining phenomenon that are “specific” and unique.

Nanoto (2013) makes the argument that clause chaining is not “special” and is rather, an instance of a broader phenomenon, asymmetric clausal coordination, which has been discussed by authors such as Postal (1998), Culicover and Jackendoff (1997), and Bjorkman (2011).

According to Nanoto (2013), clausal coordination is not very well understood, and this has had the result that in the literature, constructions have been erroneously identified as clause chaining, which are actually simple cases of asymmetric vP coordination.

Perhaps the “beads on a string” clausal conjoining phenomenon that Everett (2005, 2007, 2009, 2010) describes in Pirahã, will, with further study, turn out to not be unique to Pirahã and rather a more generic type of operation, such as asymmetric vP coordination (as per Nanoto 2013).

Asymmetric vP coordination requires the presence of embedding and recursion, and so an analysis of clausal conjunction in Pirahã, which involves asymmetric vP coordination, would also provide support for HCF’s recursion-only hypothesis.

References

Barlow, M. et al. (2009) Brownian motion on the Sierpinski carpet. [Online] Cornell University. Available from http://arxiv.org/pdf/0812.1802v2.pdf. [Accessed on 3 July 2014].

Bjorkman, B. (2011) A Syntactic Correlate of Semantic Asymmetries in Clausal Coordination. In LaCara, N., Park, Y. & Fainleib, L. (eds.). *Proceedings of NELS 41*. Philadelphia: University of Pennsylvania.

Chomsky, N. (1979) *Language and Responsibility.* New York: The New Press. p. 81.

Chomsky, N. (1981) *Lectures on Government and Binding: The Pisa Lectures.* Berlin: Walter de Gruyter.

Chomsky, N. (1986) *Knowledge of Language.* Westport: Praeger Publishers.

Chomsky, N. (1995) *The Minimalist Program.* Cambridge, Mass.: MIT Press.

Chomsky, N. (2000) Minimalist inquiries: The framework. In Martin, R. *et al*. (eds.), *Step by step: Essays on minimalist syntax in honor of Howard Lasnik*. Cambridge, MA: MIT Press. p. 240-242.

Chomsky, N. (2006) *Language and Mind (3rd Edition)*. Cambridge: Cambridge University Press.

Chomsky, N. (2007) Approaching UG from below. In Sauerland, U. & Gärtner, H.M. (eds.). *Interfaces + recursion = language?* New York: Mouton de Gruyter. p. 1-29.

Coolidge, F., Overmann, K. & Wynn, T (2010) Recusion: What is it, who has it, and how did it evolve? *Wiley Interdisciplinary Reviews: Cognitive Science*, *1*, 1-8. John Wiley & Sons Ltd. p. 547–554.

Culicover, P. & Jackendoff, R (1997) Semantic Subordination despite Syntactic Coordination. In *Linguistic Inquiry* 28.2. Cambridge, Massachusetts : The MIT Press. p. 195–217.

Dooley, R (2010) Exploring Clause Chaining. *SIL Electronic Working Papers.* Available from http://www.sil.org/resources/archives/7858. [Accessed on 3 December 2013].

Dooley, R. & Levinsohn, S. (2001) *Analyzing discourse: a manual of basic concepts.* Dallas: SIL Intl.

Everett, D. (1986) Pirahã. In Derbyshire, D.C. & Pullum, G.K. (eds.) *Handbook of Amazonian Languages.* Berlin: Mouton de Gruyter. p. 200-325.

Everett, D. (2005) Cultural constraints on grammar and cognition in Pirahã. *Current Anthropology* 46. Chicago: University of Chicago. p. 621 – 646.

Everett, D. (2007) Cultural constraints on grammar in Pirahã: a reply to Nevins, Pesetsky, and Rodrigues (2007). [Online] Available from http://ling.auf.net/lingBuzz/000427. [Accessed on 3 July 2014].

Everett, D (2009) Pirahã culture and grammar: A response to some criticisms. *Language* 85. Washington, DC: Linguistic Society of America. p. 405-442.

Everett, D. (2010) You drink. You drive. You go to jail. Where's recursion? [Online] Available from http://ling.auf.net/lingbuzz/001141. [Accessed on 3 July 2014].

Fitch, W., Hauser, M. & Chomsky, N. (2005) The evolution of the language faculty: clarifications and implications. *Cognition* 97. Elsevier. p. 179-210.

Gallejo, A. (2010) Phase Theory. *Linguistics Today* 152. Amsterdam & Philadelphia, PA: John Benjamins. p. 8-19.

Halliday, M.A.K. (2003) *On Language and Linguistics*. New York: Bloomsbury Academic. p. 96-99.

Hauser, M., Chomsky, N. & Fitch, T. (2002) The faculty of language: What is it, who has it, and how did it evolve? *Science* 298. Washington, DC: American Association for the Advancement of Science (AAAS) p. 1569–79.

Jackendoff, R. & Pinker, S. (2005). The nature of the language faculty and its implications for evolution of language (Reply to Fitch, Hauser and Chomsky). *Cognition* 97. Elsevier. p. 211-225.

Jackendoff, R. (2010). *Meaning and the Lexicon: The Parallel Architecture 1975–2010*. Oxford: Oxford University Press. p. 504.

Jackendoff, R. (2011). What is the human language faculty? Two views. *Language* 87. Washington, DC: Linguistic Society of America. p. 586-624.

Johannessen, J.B. (1998) *Coordination (Oxford Studies in Comparative Syntax).* New York: Oxford University Press.

Kayne, R.S. (1994) *The Antisymmetry of Syntax*. Cambridge, Massachusetts: The MIT Press.

Koschmann, T. (2010) On the Universality of Recursion. *Lingua* 120, No 12, Elsevier.

Lang, B. (1987) *The Concept of Style*. New York: Cornell University. p. 216-218.

Liberman, M. (2006) *Parataxis in Pirahã*. [Online] Available from http://itre.cis.upenn.edu/~myl/languagelog/archives/003162.html. [Accessed on 3 July 2014].

Lobina, D. (2011) A Running Back and Forth: A Review of Recursion and Human Language. *Biolinguistics* 5. 1-2. Berlin: De Gruyter Mouton. p. 151-169.

Longacre, R.E. (2007) Sentences as combinations of clauses. In Shopen, T. (ed.). *Language typology and syntactic description, vol. 2: complex constructions*, 2nd ed. Cambridge: Cambridge University Press. p. 372-420.

Nanoto, R. (2013) Clause chaining is asymmetric vP coordination. [Online] Available from http://ling.auf.net/lingbuzz/001912. [Accessed on 3 July 2014].

Nevins, A., Pesetsky, D. & Rodrigues, C. (2007) Pirahã exceptionality: a reassessment. [Online] Available from http://ling.auf.net/lingBuzz/000411. [Accessed on 3 July 2014].

Nevins, A., Pesetsky, D. & Rodrigues, C. (2009) Evidence and argumentation: A Reply to Everett (2009). *Language* 85. Washington, DC : Linguistic Society of America. p. 671-681.

Parker, A. (2006) Evolving the narrow language faculty: was recursion the pivotal step? In *Proceedings of the 6th International Conference on the Evolution of Language*. Available from http://www.tech.plym.ac.uk/socce/evolang6/parker.doc. [Accessed on 3 July 2014].

Pinker,S. & Jackendoff, R. (2004) The faculty of language: what's special about it? *Cognition* 95. Elsevier.

Postal, P.M. (1998) *Three Investigations of Extraction*. Cambridge, Massachusetts: The MIT Press.

Radford, A. (2004) *Minimalist Syntax*. Cambridge: University of Cambridge. p. 91.

Roberts, E. (2008) *Chapter 6 – Recursive Procedures*. [Online] Lecture notes, available from http://www-cs-faculty.stanford.edu/~eroberts/courses/cs106b/chapters/06-recursive-procedures.pdf. [Accessed on 3 July 2014].

Roeper, T. (2007) *The prism of grammar: how child language illuminates humanism*. Cambridge, Massachusetts: The MIT Press.

Sauerland, U. (2010) Experimental Evidence for Complex Syntax in Pirahã. [Online] Available from http://ling.auf.net/lingbuzz/001095. [Accessed on 3 July 2014].

Stapert, E. (2007) Linguistic theory and fieldwork in interaction: the case of Pirahã. In Austin, P.K., Bond, O. & Nathan, D. (eds.). *Proceedings of Conference on Language Documentation and Linguistic Theory.* London: SOAS. p. 235–242.

Waltz, N.E. (1976) Discourse functions of Guanano sentence and paragraph. In Longacre, R.E. & Woods, F. (eds.). *Discourse grammar: studies in indigenous languages of Colombia, Panama, and Ecuador, Part 1*. Publication number 52. Dallas, TX: Summer Institute of Linguistics & University of Texas at Arlington. p. 21-145.

Wiese, H. (2003) *Numbers, Language, and the Human Mind*, Cambridge: Cambridge University Press.

Zhang, N. (2010) *Coordination in Syntax*. Cambridge Studies in Linguistics Series 123, Cambridge: Cambridge University Press.

COLLEGE OF HUMANITIES

DECLARATION - PLAGIARISM

I, Donovan Cowie declare that

The research reported in this thesis, except where otherwise indicated, is my original research.

This thesis has not been submitted for any degree or examination at any other university.

This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
Their words have been re-written but the general information attributed to them has been referenced
Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.

This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

Signed

Donovan Cowie

College of Humanities
Schedule of Revisions Completed Post-Examination
Masters/PhD

Student Name: Donovan Cowie
Student Number: 941 300 724
Degree: Masters
Title of Dissertation: Recursion in Language: A Critical Discussion

#	Concern Expressed/Revision Required (verbatim, source, by whom, page reference)	My understanding of the concerns	Actions taken (detailed description, new page reference if applicable)
	Concerns: Examiner 1 (Internal)		
1	A number of places where the font changed.	Font change.	Applied a consistent font across the document (Arial 11)
2	(Page 3) The description of the biolinguistic framework is not linked to earlier manifestations thereof in a meaningful way. I would find it helpful to interrelate the nativist framework and the biolinguistic framework more clearly.	Make explicit the link between the nativist framework and the biolinguistic framework.	Added the paragraph below, as well as Chomsky (1979) to the reference list: Chomksy (1979: 81-86) argues against the empiricist view that the brain is a tabula rasa, empty and unstructured. In the empiricist view, all of language is learned and nothing is innate. Chomsky's view aligns itself with the nativist framework, whereby certain skills or abilities are available to the brain from birth. The biolinguistic framework arose from this view and is a cross-disciplinary approach whereby the faculty of language (FL) is understood to be a "cognitive organ", shared among humans and unique to the species, and central to Chomsky's linguistic framework.
3	(Page 3) The candidate needs to be explicit about the 'locus' of the parameters in the principles and parameters framework. He needs to make it explicit that parameters are binary values to	Make explicit the locus of parameters in the P&P framework. Make explicit that parameters are binary values to principles in UG.	Amended the following paragraph on page 3: The biolinguistic program gave rise to Chomsky’s Principles and Parameters (P&P) framework, which explains the diversity of human languages as a set of principles, such as that sentences will always have a subject, whether it is overtly pronounced or not, and a series of binary switches, or parameters, whereby for example a language could be head-initial like English, or head-final like Japanese

	principles in Universal Grammar.		(Chomsky 1981; 1986). Added the following paragraph on page 3: Principles and parameters are a part of an innate Universal Grammar and do not need to be learned by exposure to language. Rather, exposure to language merely triggers the parameters to adopt to the linguistic setting.
4	Use of ampersand “&” where “and” would have been appropriate.	Replace ampersands with “and”.	Replaced all ampersands in my discussion with “and”.
5	Colloquial expression “maths” (p. 15)	Replace “maths” with “mathematics”	Replaced “maths” with “mathematics” on page 15.
6	Candidate confuses a hyphen with a dash.	Replace incorrect occurrences of hyphens “-” with dashes “—”	Page 21: we can see evidence of yet another type of recursion — structural discourse recursion Page 36: In English, the structure in 7a) is available sententially — all in one sentence, with the two clauses combined to form a larger sentence. Page 38: Everett (2005) re-examines his own analysis, and comes to a different conclusion — that there is no recursion in Pirahã.
7	Be consistent when to use/not use an author's first name throughout.	Remove first names of authors, to be consistent.	Removed the first names of authors where they occurred, such as Noam (Chomsky) and Daniel (Everett)
8	The literature list needs to be completely reformatted/ reworked. The candidate can consult a style sheet and attend to this shortcoming.	Reference list to be reformatted according to a style sheet.	Completely reworked the reference list to be consistent with the following style sheet (Harvard guide): http://www.staffs.ac.uk/assets/harvard_quick_guide_tcm44-47797.pdf
9	Nine titles appear to be in the literature list but seem not to have been used in the text.	Remove titles in the reference list that don't appear in the text.	Removed the following titles. The other titles all appear in the text: Berwick, R, and Chomsky, N (2011) "The Biolinguistic Program. The current state of its development" in Di Sciullo and Boeckx (ed.) *The Biolinguistic Enterprise.* Oxford :

			Oxford University Press, pp. 19-41. Berwick, R (2011a) "All you need is Merge: Biology, computation, and language from the bottom up" in Di Sciullo and Boeckx (ed.) *The Biolinguistic Enterprise.* Oxford : Oxford University Press, pp. 461-491. Berwick, R (2011b) "Syntax Facit Saltum Redux: Biolinguistics and the leap to syntax" in Di Sciullo and Boeckx (ed.) *The Biolinguistic Enterprise.* Oxford : Oxford University Press, pp. 65-99. Fitch, W and Hauser, M (2004) "Computational constraints on syntactic processing in a nonhuman primate" *Science* 303. Washington, DC : American Association for the Advancement of Science (AAAS), pp 377-380.
10	Two publications appear with different dates in the text as compared with to the literature list. These are Zhang 2009 or 2010 and Pinker and Jackendoff 2004 or 2005.	Change dates in the text to be consistent with the reference list.	Changed the dates in the text (2009) to match the publication dates for Zhang in the reference list (2010). Changed the date in the text for PJ from 2005 to 2004. Pinker and Jackendoff (PJ) is a 2004 publication, as noted in the reference list.