

1 A Whig History of Generative Grammar

1.0 Introduction

Fish swim, birds fly, people speak. For the first two, the standard wisdom is that fish and birds do what they do *partly* in virtue of being biologically built to do what they do. Mentalist conceptions of linguistics apply similar reasoning to humans and their linguistic behavior. So situated, the goal of linguistics is to describe and explain the mental/brain properties that allow for human linguistic facility. More specifically, just as ornithologists take it for granted that many features of birds are biologically dedicated to efficiently supporting flight, and ichthyologists assume that fish come with many properties to optimize swimming, linguists (of the mentalist variety) propose that humans come with a faculty of language (FL) endowed with linguistically bespoke properties which *partly* ground the linguistic competence characteristic of humans.

We even know a little about the fine structure of FL due to sixty-five years of research by Generative grammarians. We know, for example, two very general things. First, that part of linguistic competence consists in having acquired a Grammar (G) able to recursively generate an unbounded number of distinct hierarchically organized structures that pair an articulation with a meaning (i.e. $\langle \pi, \lambda \rangle$ pairs). Second, we also know that any (non-pathological) child can acquire the G of any language and that the course of acquisition of that G is more or less the same across all acquirers and all Gs. This does *not* mean that there are no individual differences. Rather, the targets of acquisition and the time course of their acquisition is largely unaffected by anything other than placement in the appropriate speech community. Put any kid in any English/Swahili/Basque/... speaking environment and the child will acquire facility in English/Swahili/Basque/... in more or less the same way in more or less the same time. And acquiring facility *means* (at least) being able to pair an articulation π with a meaning λ over an *unbounded* domain of linguistic objects.

The “unbounded” part above directly implicates the existence of an acquired G; for the *only* way for a finite entity (the brain) to display an unbounded capacity like the one we find manifest in linguistic behavior (which we know involves dealing with unboundedly many different discrete hierarchically

organized objects) is as the expression of a finitely specifiable generative procedure that takes its prior computed outputs as subsequent inputs for further computation. In other words, the unbounded nature of human linguistic facility implicates the existence of Gs (i.e. recursive rule systems) that generate an infinity (i.e. unbounded number) of distinct hierarchically organized objects from a finite specifiable set of atoms, by combining these atoms together into larger structures that can themselves be further combined into yet larger structures. All of this we know, and we have known it for quite a while, and it should be neither controversial nor tendentious.

What is (rightly) debated and still under active investigation is the exact specification of the recursive procedures found in human Gs. To say that human Gs are recursive leaves open the question of what the relevant generative procedures look like. And this is a very, very, very BIG question. There are an infinite number of possible recursive functions, only a very, very, ... very small number of which (maybe just one really!) are attested in natural language grammars. Therefore, not surprisingly, generative research over the last sixty-five years has explored many options and has changed its collective mind repeatedly about the nature of the procedures that FL makes available to generate linguistic structures and establish linguistic dependencies. In what follows, I outline how the mentalist Generativist project has investigated the fine structure of FL and Universal Grammar (UG).¹ The goal is to appreciate the logic of this roughly seventy-year project and identify how the Minimalist Program (MP) conceptually fits into that project. Here goes.

1.1 Some Salient Facts, Some Obvious Consequences, and the Questions They Raise

The Generative Program began with a focus on two salient facts. The first is that native speakers of a (human) language are *linguistically creative* in the sense that they are capable of producing and understanding an unbounded number of qualitatively different discrete kinds of linguistic expressions (e.g. phrases and sentences).² The second salient fact, let's call it *linguistic*

¹ From now on, when I use the term 'Generative Program' I mean the mentalist (Chomskyan) version.

² In the immortal words of Noam Chomsky: "a mature native speaker can produce a new sentence on the appropriate occasion, and other speakers can understand it immediately, though it is equally new to them" (Chomsky 1964: 7). Importantly, these linguistic forms are discretely different, with very different contents over an effectively unbounded domain.

One further important point. 'Creativity' has been used in two very different ways in the Generative literature. The first, focused on here, deals with finitely specifying the unbounded number of well-formed structures that a native speaker can use and understand. The second, largely ignored in what follows, adverts to an observation that Chomsky highlights in his more philosophical work; namely, that language use is creative in the sense of not being

flexibility, is that *any* human child can acquire *any* human language if placed in the appropriate speech community. Further, so far as we can tell, the capacity to acquire competence in a specific language L is more or less uniform in the species in that the end state attained (linguistic competence) is (more or less) the same and its course of development is (roughly) uniform regardless of the child and regardless of the language.³

These two facts are not subtle. Nobody will win a fancy prize for doing clever and laborious experiments to discover their existence. However, until Chomsky noted them over sixty years ago, they were little noticed, and few bothered to ask how either was possible. A central ambition of the Generative Program has been to address how these two facts could be true. What allows humans to be linguistically creative and linguistically flexible? More specifically, what kinds of minds could support these two related yet different kinds of capacities?

The Generative answer to this pair of questions is now relatively well known. Here is a snapshot version.

Linguistic creativity (LC) is explained by assuming that native speakers of a human language L have a particular kind of knowledge of L . What kind? Native speakers have acquired a grammar (G) (aka, a generative procedure) that recursively specifies an open-ended number of meaning–sound pairs for that language. In other words, linguistic creativity in L (LC_L) rests (at least in part) on having a G of L (G_L). The fact that G_L is recursive explains LC 's open-ended nature, in that creatures endowed with G_L s will have knowledge of an unbounded number of linguistic objects. That's what recursive systems do. They *finitely* specify a capacity that extends over an *unbounded* (i.e. infinite) domain. LC reflects the fact that native speakers have internalized such a recursive system, and this recursively specified G is what (at least in part) endows native speakers with the power to produce endlessly many novel sentences/

stimulus bound. Here 'creativity' points to the fact that humans (generally) use language in ways appropriate to a context even though there is no sense in which the context "causes" their linguistic behavior (though, to use Cartesian terminology, it may "incline" them in certain ways – what the difference is between "causing" and "inclining" is a topic I will studiously avoid). This use of the term 'creative' points to very deep Cartesian issues concerning the distinctive nature of mental vs physical substance that lay at the heart of the Cartesian dualist project (and that twenty-first-century intellectuals have very little sympathy with today). At any rate, what is critical here is that the Generative Program has absolutely nothing of interest to say about this second sense of 'creativity.' Indeed, so far as I can tell, *nobody* has anything interesting to say about it. For a review of this issue in a psychological setting, the reader can do no better than reread Chomsky's wonderful review of Skinner's *Verbal Behavior* (1967). For what follows, however, the reader should simply assume that nothing we say here bears on the deep Cartesian issues alluded to.

³ Note the qualifiers 'more or less' and 'roughly.' Of course children differ, but the milestones along the acquisition path and the end state attained are very similar and seem to be largely independent of the target language being acquired. See Brown (1973) and Slobin (1986) for discussion.

phrases and allows them to understand such novelties upon hearing them. So, to the question: how is it that competent native speakers can be linguistically creative?, we have the answer: in virtue of having internalized a G_L , a recursive generative procedure, that specifies the (unboundedly many) objects of L for their particular native language L. So LC is (in part) explained in terms of internalized G_L s.⁴

And what of linguistic flexibility (LFL), the capacity humans have to acquire any G_L under the right input conditions? The possibility of LFL follows if humans come equipped with a faculty of language (FL) with the power to yield grammars when fed the (linguistically) relevant data. In the simple case, such data will be bits of language L produced/uttered by proficient native speakers of L based on the G_L s that they as proficient native speakers have internalized.⁵ So, LFL follows if humans are endemically endowed with FLs that can map the bits of a language L a child is exposed to (and takes in) (aka, “primary linguistic data of L” (PLD_L)) onto a grammar of L (G_L). Or pictorially:

$$(1) \quad \text{PLD}_L \rightarrow \text{FL} \rightarrow G_L$$

In other words, FL is a function that takes PLD_L and maps it onto a G_L (i.e. $\text{FL}(\text{PLD}_L) = G_L$). From the perspective of the Generative Program so construed, G and FL are empirical hypotheses about how two facts (i.e. LC and LFL) are possible. From this perspective, the program identifies the focus of inquiry to be G_L s (specific generative procedures internalized by native speakers of particular Ls) and FL (the recipe that allows humans to acquire G_L s when appropriately linguistically placed).

That G_L s exist in native speakers and *that* FL exists as a human biological endowment are **NOT** exciting claims. They are close to the conceptual minimum required to accommodate our two very salient facts, LC and LFL.

⁴ Let me emphasize, recursively specifying the well-formed objects of L via a G_L is a necessary but not sufficient step in explaining the fullness of linguistic creativity. Gs are not parsers and Gs are not producers (which have in addition, for example, memory structures (i.e. a “tape” or “stack”), write-to and read-from procedures, among other things, as well as a specification of the relevant G rules specified in the machine table), though they are a necessary part of both (see Berwick and Weinberg (1984) for elaboration). This noted, the Generative Program has focused on (i) specifying the generative procedures that Gs contain and (ii) explaining why particular Gs contain these and not others.

⁵ But this really is the simple case. It appears that humans can create Gs from much thinner linguistic input if the emergence of creoles from pidgins is any indication. And from rather confusing input: as the linguistic products of many Gs (say in multilingual settings) allow for the simultaneous emergence of multiple relatively neatly segregated Gs rather than a single G farrago of the different inputs. The generative program has largely restricted attention to the simple case, assuming that explicating how it might arise is challenging enough, and assuming that the complex cases will build on mechanisms identified in the simple one. For discussion and references, see Bickerton (1984) and accompanying critical notes.

That something like these two cognitive (and ultimately biological) objects exist is really a no-brainer. After all, to say that a native speaker has internalized a G_L is to acknowledge that s/he has an unbounded capacity to use and understand L . And to say that someone has an FL is just to say that s/he has a second-order capacity to acquire the first-order capacity specified by a G_L . But given that native speakers *are* quite obviously linguistically creative and given that humans *are* quite obviously capable of acquiring any G_L if exposed to PLD_L , the supposition that G_L s and FLs “exist” and are legitimate objects of inquiry must be correct. The inferential leaps from LC to G_L s and from LFL to FL are very, very short.

The hard empirical question, then, is not *whether* these objects exist, but *what they look like* in detail. In other words, the hard part of the Generative Program is specifying what G_L s look like (i.e. what *kinds* of recursive generative procedures they embody) and what the fine structure of FL is (i.e. what principles it must embody to allow for the acquisition of G_L s for arbitrary L s).

Given this framing, an important subsidiary question of interest is the degree to which the structures of G_L s and the fine structure of FL are linguistically bespoke or cognitively and/or computationally generic. In other words, a central sub-project of the program will be to determine to what extent (if any) our first- and second-order linguistic facilities require a mental apparatus specifically tuned to the properties of language and to what extent the capacities manifested in linguistic behavior reflect our combined cognitive and computational powers more generally.

In case you haven't noticed, this last question is quite definitely an empirical one. To date, the Generative answer has been that linguistic proficiency *does* require specifically linguistic cognition. The minimalist codicil to this general conclusion has been that it only requires a dollop of such, rather than a large heaping shovelful. We will return to this issue anon, but for now, let's take a quick trip through the history of Generative Grammar so that we can appreciate how Minimalism, the latest step in the Generative Program, fits into the entire Generative Grammar project.⁶

⁶ This chapter is titled “A Whig History of Generative Grammar.” According to *Wikipedia*, a Whig history is “a form of historiography that presents the past as an inevitable progression towards ever greater ... enlightenment culminating in modern forms.” In other words, it is not “real” history. Rather it is what my good and great friend Elan Dresher has described as “history we can use.” Philosophers label such accounts “rational reconstructions.” What makes Whig histories useful is that they adopt the charming conceit that actual intellectual development tracks the logic of the ideas involved. History is then a movement from good ideas to even better ones, with historical progression tracking intellectual improvement. This, we know, is not true of real intellectual history. Real history is much bumpier and haphazard. Whig history, then, is not so much a description of what actually happened but of what *should have* happened. And that is precisely what makes it useful, for it exposes the intellectual linkages among the evolving successful ideas.

1.2 The First Two Stages of the Generative Program

Again, let's start with the two big facts (i.e. LC and LFL) and ask how to rationally investigate them. Recall that addressing LC requires saying something about the G_L s that a native speaker of L has acquired, in particular a specification of the generative procedures that it embodies (i.e. the particular rules of grammar that characterize a native speaker's (unbounded) knowledge of/sense of the language L). And addressing LFL requires specifying the fine structure of FL that allows humans to become native speakers of a particular L, which means specifying how a person uses PLD_L to acquire their G_L .⁷

This description of the research problem immediately suggests a rational order of inquiry. To address LFL questions requires having some G_L specimens. After all, the LFL question is how humans acquire *grammars*, and unless we have some idea of what kinds of grammars humans actually acquire, it will be well-nigh impossible to investigate how humans do what they/we do.⁸ So, as a practical matter, the first step in the Generative Program will be to find some plausible candidate rules of grammar embodied in particular G_L s. Not surprisingly, this kind of investigation indeed characterizes a good deal of the first stages of Generative inquiry.

So, the first question on the research agenda should have been (and was): What properties (rules, generative procedures, principles) characterize individual Gs? More particularly, what kinds of recursive rules do G_L s incorporate?

We know part of the answer to this last question because of another obvious fact about natural languages: The kinds of linguistic objects that Gs relate are meaning–sound pairings. For example, among the things a native speaker of English knows is that *Dogs chase cats* does not mean the same thing as *Cats chase dogs* while *Cats are chased by dogs* does. There are an unbounded number of such systematic facts that a competent native speaker of a given natural language knows.

⁷ Note that this is an *idealization* of the problem. Nobody actually believes that native speakers have unique Gs. For example, it is widely recognized that native speakers have different “registers” (i.e. Gs) that they switch between. However, idealizing to the situation where a person's competence is restricted to a single G and where the relevant PLD used is uniform and unsullied still leaves a *very* hard acquisition problem. Moreover, there is little reason to think that solving this simplified problem will involve different mechanisms than solving the more realistic one, in the sense that we will have to adopt entirely different generative procedures or FL principles when we weaken this idealization. The more realistic case, in other words, should involve the same kinds of rules and principles *and then some*, rather than completely different kinds of rules and principles than are useful in the idealized case. For those that like analogies to physics (a “real” science), think *frictionless* inclined planes.

⁸ The ‘they’ is a nod to Chomsky's rational Martian scientist (the equivalent in Chomsky's world to Fodor's grandmother in his).

Thus we know two important things about any G_L : (i) it involves recursive rules and (ii) it produces meaning–sound pairings.

The first fact suggests that linguistic competence consists (in part) in mastery of a system of rules that specifies the natural language mastered. Why a rule system? Because that is the only way to finitely specify an effectively infinite capacity. We cannot just list the objects in the domain of a native speaker's competence and treat the capacity as akin to looking things up on a giant list because, given LC, the list would have to go on forever. The capacity can only be specified in terms of a finite procedure that describes (i.e. generates) it. Thus, we conclude that linguistic mastery of a language L consists (in part) in acquiring a set of rules (i.e. a G_L) that generate the kinds of linguistic objects that a native speaker of L is competent with.

The second fact tells us something more about these G_L s. They must specify pairings of meanings with sounds. Thus the rule systems that native speakers have mastered are rules that generate objects with two distinctive properties. G_L s consist of generative procedures that tie a specific meaning profile together with a specific sound profile,⁹ and they do this over an effectively infinite domain. So G_L s are functions whose range is meaning–sound pairs, viz. an infinite number of objects like this: $\langle m, s \rangle$. What's the domain? Some finite set of “atoms” that can combine again and again to yield more and more complex $\langle m, s \rangle$ pairs. Let's call these atoms “morphemes.”

Putting this all together, we know from the basic facts and some very elementary reasoning that native speakers master G_L s (recursive procedures) that map morphemes into an unbounded range of $\langle m, s \rangle$ s. **THIS. WE. KNOW.** What we don't know is what the specific rules that G_L s contain look like (or, for that matter, what the ‘m's and ‘s's look like). And that brings us to our first research question: *describe specific rules characteristic of natural language G_L s and specify their variety and interactions.* The earliest Generative research aimed to provide some candidate rules of specific grammars and show how their interactions would mirror some of the complexities that native speakers' competence displays. In other words, the first order of business in Generative research involved producing detailed model grammars of the kinds of rules that particular G_L s have and how these rules interact. Many different rules were investigated: movement rules, deletion rules, phrase structure rules and binding rules, to name four. And their complex modes of interaction were limned. Consider some details.

⁹ Typically, a “sound” profile involves vocal tracts and auditory perception. However, this is not required, as the existence of signed languages indicates. Here the articulators are manual and facial and the perceptual system is visual.

1.3 Step 1: Some Possible Rules of G_L s

Recall that one of the central facts about natural languages is that they contain a practically infinite number of objects that pair a meaning with a sound.¹⁰ They also contain dependencies defined over the structures of these objects. In early theories of Generative Grammar, phrase structure (PS) rules recursively specified the infinite class of well-formed “base” structures in a given G . Lexical insertion (LI) rules specified the class of admissible local dependencies in a given G , and transformational (T) rules specified the class of non-local dependencies in a given G .¹¹ Let’s consider each in turn.

PS-rules are recursive and their successive application creates bigger and bigger hierarchically organized structures on which LI- and T-rules operate to generate other structures and dependencies.¹² (2) provides some candidate PS-rules (the ‘(...)’ indicates optional expansion):

- (2)
- a. $S \rightarrow NP \text{ aux } VP$
 - b. $VP \rightarrow V (NP) (PP)$
 - c. $NP \rightarrow (\text{det}) N (PP) (S)$
 - d. $PP \rightarrow P NP$

These four rules suffice to generate an unbounded number of hierarchically structured objects. Thus, a sentence like *John kissed Mary* has the structure in (3) generated using rules (2a,b,c).

- (3) $[_S [_{NP} N] \text{ aux } [_{VP} V [_{NP} N]]]$

LI-rules like those in (4) insert terminals into these structures, yielding the structured phrase marker (PM) in (5):

¹⁰ Linguists have known for quite a while that these objects have a hierarchical as well as a linear structure. I put this aside for now.

¹¹ The earliest Generative grammars’ PS-rules characterized the class of “kernel” sentences. These were very simple and involved no embedding. The action in these theories was with two kinds of transformations, one that combined two kernels and one that changed the structure of a given input. I am abstracting from most of these complexities here. Remember what you are getting here is a *Whig* history, not the real thing.

¹² Strictly speaking, until the Minimalist Program period, Generativists took the string to be the basic ontological unit. The rules in (2) were defined in terms of strings and sets of sets (of sets ...) of strings. Recursive rules did not output hierarchical objects (i.e. derived phrase markers) but specified recursive generation paths (i.e. derivation trees), though this is how linguists thought of things. I abstract from these issues here, important though they are, especially in light of the fact that one of the interesting features of Minimalism has been the rejection of strings as the fundamental ontological grammatical unit. What I mean by this is that phrases and transformations in early Generative theories are objects defined in terms of strings, not themselves basic units. Minimalism fundamentally reverses this metaphysics. Merge treats sets (or, more accurately, constituents) as the fundamental units of analysis, with phrases and movement (transformational) dependencies as ontologically fundamental. In the standard minimalist versions, strings are secondary objects with little grammatical potency. This is a very big change in perspective, and I will ignore it in the discussion of the Whig history that follows.

- (4) a. $N \rightarrow \text{John, Mary} \dots$
 b. $V \rightarrow \text{kiss,} \dots$
 c. $\text{aux} \rightarrow \text{past}$
- (5) $[_S [_{NP} [_N \text{John}]] [_{\text{aux}} \text{past}] [_{VP} [_V \text{kiss}]] [_{NP} [_N \text{Mary}]]]]^{13}$

PMs like (5) also reflect local inter-lexical dependencies. Note that replacing *kiss* with *arrive* yields an unacceptable sentence: **John arrived Mary*. The PS-rules can generate the relevant structure (i.e. (3)), but the LI-rules cannot insert *arrive* in the *V* position of (3) because *arrive* is not lexically marked as transitive. In other words, $NP^{\wedge}kiss^{\wedge}NP$ is a fine local dependency, but $NP^{\wedge}arrive^{\wedge}NP$ is not.

Given structures like (5), T-rules can apply to rearrange them, thereby coding for a variety of non-local dependencies.¹⁴ What kind of dependencies? The unit of transformational analysis in early Generativism is the construction. Some examples include: Passive, *Wh*-questions, Polar questions, Raising, Equi-NP Deletion (aka control), Super Equi, Topicalization, Clefting, Dative Shift (aka Double Object Constructions), Particle Shift, *There* constructions (aka Existential Constructions), Reflexivization, Pronominalization, Extraposition, among others. Though these rules fall into some natural formal classes (see below), they also contain a great deal of construction-specific information, reflecting construction-specific morphological peccadillos and restrictions. Here's an illustration.

Consider the Passive rule in (6). 'X'/'Y' in (6) are variables. The rule says that if you can factor a string into the parts on the left of the arrow (viz. the structural description) you can change the structure to the one on the right of the arrow (the structural change). Applied to (5/7a), this yields the derived phrase marker (7b).

- (6) $X - NP_1 - AUX - V - NP_2 - Y \rightarrow X - NP_2 - \text{be+en} - V - \text{by } NP_1 - Y$
- (7) a. $X - [_{NP} \text{John}] [_{\text{aux}} \text{past}] [_V \text{kiss}] [_{NP} \text{Mary}] - Y$
 b. $X - [_{NP} \text{Mary}] [_{\text{aux}} \text{past}] \text{be+en} [_V \text{kiss}] \text{by } [_{NP} \text{John}] - Y$

Note that the rule codes the fact that what was once the object of *kiss* is now a derived subject. Despite this change in position, *Mary* is still understood as

¹³ What does (5) code? It makes explicit the grammatical 'is-a' relation. For example, it "says" that *John-past-kiss-Mary* is an S. It also "says" that that *Kiss-Mary* is a VP and that *Mary, John* are each an NP. The possible factoring of strings according to the is-a relation allows for the specification of rules that change a string that has one factorization (aka, structural description) into another via a rule. This is illustrated in (6) and (7).

¹⁴ In the earliest Generative theories, recursion was also the province of the transformational component, with PS-rules playing the far more modest role of specifying the kernel sentences. However, from *Aspects* (Chomsky 1965) onward, the PS-rules become the recursive engine of the grammar. Transformations do not generally create "bigger" objects. Rather, they specify licit grammatical dependencies within PS-created structures.

the kissee. Similarly, *John*, the former subject of (5) and the kisser, is now the object of the preposition *by*, and still the kisser. Thus, the passive rule in (6) codes the fact that *Mary was kissed by John* and *John kissed Mary* have a common thematic structure as both have a derivation which starts from the same underlying PM in (5). In effect, this proposal tracks the non-local dependency between *Mary* and *kiss* in *Mary was kissed by John* by proposing that the input to this sentence involves a PM where *kiss* and *Mary* are locally proximate (as in (5)).

The research focus in this first phase of grammatical investigation was on carefully describing the detailed features of a variety of different constructions, rather than on factoring out their common features.¹⁵ Observe that (7b) introduces new morphemes into the PM (e.g. *be+en*, *by*), in addition to rearranging the nominal expressions. T-rules did quite a bit of this, as we shall see below. What's important to note for current purposes is the division of labor between PS-, LI- and T-rules. The first generates unboundedly many hierarchical structures, the second "chooses" the right ones for the lexical elements involved (and locally codes their "thematic" properties) and the last rearranges them to produce novel surface forms that retain the "thematic" relations specified in the inputs to the T-rules, even when the relata are no longer in their original proximate positions.¹⁶ So, for example, in (7b) *Mary* is still understood as the kissee despite no longer being adjacent to the verb *kiss*.

T-rules, despite their individual idiosyncrasies, fall into a few identifiable formal families. For example, control constructions are generated by a T-rule (Equi-NP Deletion) that deletes part of the input structure. Sluicing constructions also delete material but, in contrast to Equi-NP Deletion, they do not require a PM-internal grammatical trigger (aka antecedent) to do so. Movement rules (like Passive in (6) or Raising) rearrange elements in a PM. And T-rules that generate Reflexive and Bound Pronoun constructions neither move nor delete elements but replace the lower of two identical lexical NPs with morphologically appropriate formatives (as we will illustrate presently).

In sum, the first epoch of Generative inquiry provided a budget of *actual examples* of the kinds of rules that Gs contain (i.e. PS, LI and T) and the kinds of properties these rules had to have to be capable of specifying the kinds of recursion and the kinds of dependencies characteristically found within natural languages. In other words, early Generative work developed a compendium of examples of *actual G rules in a variety of languages*.

¹⁵ This is not quite right, of course. One of the glories of early Generative research is Ross's discovery of islands, and many different constructions obeyed them.

¹⁶ Thematic is in scare quotes because the term is anachronistic. It was not used in the earliest papers, though the later term correctly describes what Deep Structures were intended to represent.

Nor was this all. Early Generative Grammar also provided models for how these different rules interact. Recall that one of the key features of natural languages is that they include effectively unbounded hierarchically organized objects. This means that the rules talk to one another and apply to one another's outputs to produce an endless series of complex structures and dependencies. Early Generative research started exploring how G rules could interact and it was quickly discovered how complex and subtle G interactions could be. For example, in the Standard Theory, rules apply cyclically (from smaller domains to larger domains that contain these smaller domains) and in a certain fixed order (e.g. PS-rules applying before T-rules). Sometimes the order of rule application is intrinsic (follows from the nature of the rules involved) and sometimes not. Sometimes the application of a rule creates the structural conditions for the application of another (feeding), sometimes it destroys the structures (bleeding) thereby preventing a possible operation from applying. These rule systems could be very complex, and these initial investigations gave linguists a first serious taste of what a sophisticated capacity natural language competence was.

It is worth going through an example to get a feel for this complexity. For illustration, consider some binding data and the rules of Reflexivization and Pronominalization, and their interactions with PS-rules and T-rules like Raising.

Lees and Klima (L&K) (1963) offered the following two rules to account for an interesting array of binding data in English (see data in (10)–(13)).¹⁷ These rules must apply when they can and are (extrinsically) ordered so that (8) applies before (9).

- (8) Reflexivization:
 $X - NP_1 - Y - NP_2 - Z \rightarrow X - NP_1 - Y - \text{pronoun+self} - Z$
 (where $NP_1 = NP_2$, pronoun has the phi-features of NP_2 , and NP_1/NP_2 are in the same simplex sentence)
- (9) Pronominalization:
 $X - NP_1 - Y - NP_2 - Z \rightarrow X - NP_1 - Y - \text{pronoun} - Z$
 (where $NP_1 = NP_2$ and pronoun has the phi-features of NP_2)

As is evident, the two rules are formally very similar. Both apply to identical NPs in a phrase marker and morphologically convert one to a reflexive or to a pronoun. (8), however, only applies to nominals in the same simplex clause (i.e. to "clause-mates"), while (9) is not similarly restricted. As (8) obligatorily applies

¹⁷ Chomsky's (1975 (1956)) *Logical Structure of Linguistic Theory* (LSLT) was a very elaborate investigation of an English G. The Lees and Klima (1963) rules for pronouns and reflexives discussed here have direct antecedents in LSLT. However, the specific rules discussed above as illustrations here are those offered in Lees and Klima (1963).

before (9), Reflexivization will bleed the environment for the application of Pronominalization by changing NP₂ to a reflexive (thereby rendering the two NPs no longer “identical”). A consequence of this ordering is that Reflexivization and Pronominalization rules apply in distinct domains. In English, this means that Reflexives and (bound) pronouns must be in complementary distribution.¹⁸

An illustration should make things clear. Consider the derivation of (10a) (where *himself/him* are understood as anaphorically dependent on *John*₁). It has the underlying form (10b). We can factor (10b) as in (10c) as per the Reflexivization rule (8). This results in converting (10c) to (10d) with the surface output (10e) carrying a reflexive interpretation. Note that the Reflexivization derivation codes the fact that *John* is both washer and washee, as well as that *John* non-locally relates to *himself*.

- (10) a. John₁ washed himself/*him
 b. John washed John
 c. X-John-Y-John-Z
 d. X-John-Y-him+self-Z
 e. John washed himself

What blocks *John likes him* with a similar anaphoric reading (i.e. where *John* is co-valued with *him*)? To derive this structure Pronominalization must apply to (10c). However, it cannot, as (8) is ordered before (9) and both rules are obligatory (i.e. they must apply when they can apply). But once (8) applies, we get (10d), which no longer has a structural description amenable to (9). Thus, the application of (8) bleeds the grammatical context for the application of (9) and *John likes him* with a bound reading of the pronoun cannot be derived (i.e. there is no licit grammatical relation between *John* and *him*).

This changes in (11). Reflexivization cannot apply to (11c) as the two *Johns* are in different clauses. As (8) cannot apply, (9) can (indeed, must) as it is not similarly restricted to apply to clause-mates. In sum, the inability to apply (8) allows (and demands) the application of (9). Thus does the L&K theory derive the complementary distribution of reflexives and bound pronouns.

- (11) a. John believes that Mary washed *himself/him
 b. John believes that Mary washed John
 c. X-John-Y-John
 d. X-John-Y-him
 e. John believes that Mary washed him

There is one other feature of note: The binding rules in (8) and (9) also effectively derive a class of (what are now commonly called) Principle C effects,

¹⁸ Cross-linguistic work on binding has shown the complementary distribution of reflexives and bound pronouns to be robust across natural languages, and so deriving the complementarity has become a boundary condition on the empirical adequacy of binding theories.

given the background assumption that reflexives and pronouns morphologically obscure an underlying expression which is identical to the antecedent. Thus, the two rules prevent the derivation of structures like (12) in which the bound reflexive/pronoun c-commands its antecedent.¹⁹

- (12) a. *Himself_i kissed Bill_i
 b. *He_i thinks that John_i is tall

It should be noted that deriving Principle C effects in this way is not particularly deep. The rules derive the effect by stipulating that it should be the higher (actually, leftmost) of two identical NPs that is retained in the structural change of the relevant transformation while the lower (rightmost) one is replaced by a reflexive/pronoun.²⁰

The L&K theory can also explain the data in (13) and (16) in the context of a G with a rule like Raising to Object in (14), which, let's assume, obligatorily applies before (8)/(9).

- (13) a. *John_i believes him/he-self_i is intelligent
 b. John_i believes he_i is intelligent

- (14) Raising to Object:
 $X - V - C - NP - Y \rightarrow X - V - NP - C - Y$
 (where C, the complementizer, is phonetically null and non-finite)²¹

(14) cannot apply to raise the finite embedded subject in (15) to the matrix clause, as the null complementizer C of the embedded clause is finite. This prevents (8) from applying to derive (13a), as (8) is restricted to NPs that are clause-mates. But, as failure to apply (8) requires the application of (9), the mini-grammar depicted here leads to the derivation of (13b) from (15).

- (15) John_i believes C John_i is intelligent

Analogously, (8), (9) and (14) also explain the facts in (15), if (14) is obligatory and must apply when it can.²²

¹⁹ α c-commands β iff every branching category that dominates α dominates β .

²⁰ Depending on how "identical" is understood, the L&K theory also prevents the derivation of sentences like *John kissed John* where the two *Johns* are understood as referring to the same individual. How exactly to understand the identity requirement was a vexed issue that was partly responsible for the replacement of the L&K theory. One particularly acute problem was how to derive sentences like *Everyone kissed himself*. It clearly does not mean anything like 'everyone kissed everyone.' What then is its underlying form so that (8) could apply to it? This was never satisfactorily cleared up and led to revised approaches to binding, as we shall see.

²¹ This is not how the original raising to object rule was stated, but it's close enough. Note too, that saying that C is finite means that it selects for a finite T. In English, for example, *that* is finite and *for* is non-finite. We are here assuming that there is a phonetically null version of each of these Cs.

²² I leave the relevant derivations as an exercise.

- (16) a. John₁ believes himself₁ to be intelligent
 b. *John₁ believes him₁ to be intelligent

The L&K analysis can be expanded further to handle yet more data when combined with other rules of G. And this is exactly the point: to investigate the kinds of rules Gs contain by seeing how their interactions derive non-trivial linguistic data sets. This allows us to explore what kinds of rules exist (by proposing some and seeing how they work) and what kinds of interactions rules can have (they can feed and bleed one another, they are ordered, obligatory, etc.).

The L&K analysis above illustrates two important features of these early proposals. First, it (in combination with other rules) compactly summarizes a (practically infinite) set of binding “effects,” patterns of data concerning the relation of anaphoric expressions to their antecedents in a range of phrasal configurations. It doesn’t outline *all* the data that we now take to be relevant to binding theory (e.g. it does not address the contrast in *John₁’s mother likes him/*himself₁*), but many of the data points discussed by L&K have become part of the canonical data set that any theory of binding is responsible for. Thus, the complementary distribution of reflexives and (bound) pronouns in these sentential configurations is now a canonical fact that every subsequent theory of binding has aimed to explain. So too the locality (*viz.* the clause-mate condition) required between antecedent and anaphor for successful reflexivization, the anti-locality requirement on licit bound pronouns (i.e. bound pronouns and their antecedents *cannot* be clause-mates) and the prohibition against anaphors c-commanding the antecedents of which they are anaphoric dependents.

The *variety* of data L&K identifies is also noteworthy. From very early on, the Generative Program understood that both positive and negative data are relevant for understanding how FL and Gs are structured. Positive data is another name for the “good” cases (examples like (10e) and (11e)), where an anaphoric dependency is licensed. Negative data are the * cases (examples like (12a) and (16b)) where the relevant dependency is illicit. Grammars, in short, not only specify what *can* be done, they also specify what *cannot* be done. Generativists have discovered that negative data often reveals more about the structure of FL and a particular G than positive data does.²³

Second, L&K provides a *theory* of these effects in the two rules (8) and (9). As we shall see, this theory was *not* retained in later versions of Generative Grammar.²⁴ The L&K account relies on machinery (obligatory rule application,

²³ The focus on negative data has also been part of the logic of the Poverty of Stimulus (POS) argument (aka Plato’s problem). Data that is absent is hard to track without some specification of what absences to look for (i.e. some specifications *not provided by the data itself* of where to look). More important still to the logic of the POS is the impoverished nature of the PLD available to the child. We return to this at the end of the next section.

²⁴ Though I am personally rather fond of the L&K theory and have argued that we should return to a modern version of it. For some relevant discussion which I don’t currently entirely endorse, see Hornstein (2008).

bleeding and feeding relations among rules, rule ordering, Raising to Object, etc.) that was replaced in later theory by different kinds of rules with different kinds of properties. The L&K rules themselves are also very complex (e.g. they are extrinsically ordered). Later approaches to binding attempt to isolate the relevant factors and generalize them to other kinds of rules. We return to this anon.

One more terminological point: In what follows, it is useful to distinguish between “effects” and “theory.” As Generative theories changed over the years, discovered effects (e.g. that Reflexivization and Prenominalization are in complementary distribution, that *Wh*-movement out of islands is illicit, that PRO appears in non-finite subject positions, etc.) have been largely retained, though the theories developed to explain these effects have often changed significantly.²⁵ For example, as we will see below, the L&K theory was replaced by Principles A, B and C of the binding theory, yet a central binding effect (viz. the complementarity between Reflexivization and Prenominalization) was retained. This is similar to what we observe in the mature sciences (think ideal gas laws with respect to thermodynamics and later statistical mechanics). What is *clearly* cumulative in the history of Generative Grammar is the conservation of discovered effects. Theory changes, and deepens. Some theoretical approaches are discarded, some refashioned and some resuscitated after having been abandoned. Effects, however, are largely conserved and a standard boundary condition of theoretical admissibility in later theory is that the new theory with its novel assumptions explain the effects that the older replaced theory explained.

I should also add that for large stretches of theoretical time, basic theory has also been conserved (e.g. some version of the cycle has been with us since almost the inception of the Generative Program). However, the cumulative nature of Generative research is most evident in the preservation of the various discovered effects. In Section 1.6, I list a number of these. It is an impressive group. But first, let’s take a look at how establishing a set of plausible G rules sets the stage for addressing the second Generative question concerning linguistic flexibility.

1.4 Step 2: Categorizing, Simplifying and Unifying the Rules

As noted, the first stage of Generative research yields a bunch of rules describing a bunch of linguistic constructions in addition to providing early models of how the different kinds of rules might interact to generate an unbounded number of $\langle m, s \rangle$ s within a given natural language *L*. Here we look at how this prepared the way for research focusing on the second question concerning the

²⁵ Though, like all theory, earlier ideas are recycled with some reinterpretation. See below for illustration.

nature of linguistic flexibility (LFL): *what must FL look like given that it can produce G_L s with these kinds of rules and these kinds of interactions?* At the risk of stating the obvious (not a risk I worry much about), observe that asking this question only makes *practical* sense once we have serious candidate G_L s, language-specific generative procedures. For the LFL question to be fecund *presupposes* that we have identified *some* G_L rules with the right properties, for it is G_L rules *like these* that we want FL to target. Given this, it is not surprising that LFL issues awaited (partial) answers to the conceptually prior LC question.

Investigations into FL moved along two tracks: (i) cross-linguistic investigations of G_L s different from English to see to what degree the G_L s proposed for English carry over to those of other natural languages and (ii) simplification and unification of G_L s so as to make them more natural “fits” for FL. The second Generative epoch stretches from roughly the mid-1970s to the early 1990s. Within the Chomsky mentalist version of the Generative Grammar Program, the classical example of this kind of work is *Lectures on Government and Binding* (LGB; Chomsky (1981)). Our question in this section is: What did LGB accomplish and how did it do this?

LGB was a mature statement of the work that began with Chomsky’s (1973) *Conditions on transformations*. This work aimed to simplify the rules that G_L s contain by distilling out those features of particular G rules that could be attributed to FL more generally. The distilled features were attributed to FL as design features and were dubbed “Universal Grammar” (UG). The basic GB research strategy was to simplify particular G_L s by articulating the innate UG principles of FL. Part of this consisted in categorizing the possible rules a G_L could contain. Part involved postulating novel theoretical entities (e.g. traces) which served two functions: (i) they allowed the T-rules to be greatly simplified and (ii) they allowed for a partial unification of two, heretofore distinct, parts of the grammar, namely binding and movement.

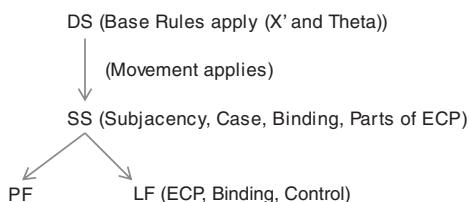
Articulating FL in this UGish way also had a natural acquisition interpretation relevant to addressing the fact of linguistic flexibility: in learning a particular G_L , the language acquisition device (LAD, aka the child) need “abstract”/“induce” only simple rules from the data, with the more recondite forms of knowledge attained by the child (that earlier theory had coded as part of a rule’s structural description) now being traced to built-in (i.e. innate) structural features of FL (aka the principles of UG). As UG principles are innate, they need not be acquired and so are not hostage to the details of the PLD. That’s the logic, and the program was to simplify language-specific rules by offloading many of their most intricate features to endemic properties of FL as embodied in principles of UG.

As noted, rule simplification has an appealing consequence for acquisition. As language-specific rules can (and do) vary, they must be learned. Thus,

simplifying them makes them easier to acquire, while enriching UG allows this simplification to occur without (it is hoped) undue empirical loss. That was the logic. Here are some illustrations.

LGB postulated a modular structure for natural language Gs with the following components and derivational flow.

- (17) The LGB Grammar
- A Rule Types/Modules
1. Base rules:
 - a. X'-Theory
 - b. Theta theory
 2. Movement rules (A and A')
 3. Subjacency theory
 4. Empty Category Principle (ECP)
 5. Case rules
 6. Binding rules:
 - a. Principle A, anaphors
 - b. Principle B, pronominals
 - c. Principle C, R-expressions
 7. Control rules
- B. The Derivational Y-Model:



The general organization of the grammar, the ‘Y’-model (17/B), specifies how/where in the derivation these various rules/conditions apply. The Base Rules (17/1) generate X'-structured objects (17/1a) that syntactically reflect “pure GF- θ ” (17/1b) (viz. that all and only thematic positions are filled; so logical subjects and logical objects are, at DS, grammatical subjects and grammatical objects), creating phrase markers analogous to (but not exactly the same as) Deep Structures in the Standard (i.e. *Aspects*) theory. Targets of movement operations are positions generated by the X'-rules in the base which lexical insertion (LI) rules have not lexically filled.²⁶ The output of the base component (the combination of X'-rules and LI-rules) is input to the T-component,

²⁶ X'-theory eliminates PS-rules and substitutes the idea that syntactic structure is projected (according to the X' schema) from heads coded with the relevant subcategory and selection information. Prior theories of the base were redundant in that subcategorization restrictions were coded twice, once in the PS-rules and then again in the lexical insertion filters. X'-theory eliminates this redundancy.

the part of the grammar that includes movement operations (and that extends the derivation from DS to SS). At SS, various relations are licensed (case, binding, some ECP trace licensing conditions). Derivations then split, with the grammatical structure relevant to sound interpretation (the phonological form (PF)) separated from that required for meaning interpretation. The latter is then mapped via (possible additional) abstract movement rules (rules that have no overt phonetic realization) to logical form (LF), which is the phrase marker that codes the grammatical information relevant to meaning interpretation.²⁷

Let's consider some of the key theoretical and conceptual innovations in the GB model.

Movement rules are entirely reconceptualized in LGB in two important ways. First, they are radically simplified. The simplification involves stripping movement of its constructional specificities (abstracting away from what was moved (e.g. a Topic, or a *Wh*-morpheme or a Focused element)) and distilling out the fundamental movement operation (dubbed "Move α "). The rule Move α can move any expression anywhere, subject to one restriction: all G rules, including Move α , are structure preserving in the sense that all the constituency present in the input to the rule is preserved/conserved in the output of the rule.²⁸

In concrete terms, this preservation/conservation assumption motivates the second key innovation in LGB: Trace theory. Trace theory has two important *theoretical* consequences: (i) it is a necessary ingredient in the simplification of movement rules to Move α and (ii) it serves to unify movement and binding theory.

So, simplification, unification and conservation are all pressed into service in developing the GB theory of FL. Let's consider how unifying and simplifying earlier Standard Theory accounts of G operations gets us to a GB-like theory.

First the process of simplification: LGB replaces complex construction-based rules like Passive, which in the Standard Theory look something like (18), with the simple rule of "Move NP," this being an instance of Move α with $\alpha = \text{NP}$.

$$(18) \quad X - \text{NP}_1 - Y - V - \text{NP}_2 - Z \rightarrow X - \text{NP}_2 - \text{be+en} - V - \text{by NP}_1$$

(where NP_1 and NP_2 are clause-mates)

Move NP is simpler in three ways. First, (18) involves the movement of two NPs (note: the structural change on the right of the arrow differs from the

²⁷ A friend tells me that this paragraph is heavy lifting for the uninitiated. There are excellent texts that go into the details of how the model works. My preferred text is Haegeman (1991) and I recommend it for anyone wishing to get into the glorious details. Luckily, knowing these is unnecessary for understanding the main points of what follows.

²⁸ Up to adjunction, which can create some new structure. The exact nature of adjunction was and remains a thorny theoretical problem. I will ignore it here.

structural description on the left in that NP₂ has moved to near the front of the string from post-verbal position and NP₁ has moved from the left edge to the right and now forms part of a *by*-phrase). Passivization, when analyzed in Move α terms (aka Move NP when $\alpha = \text{NP}$), involves two applications of the simpler rule rather than one application of the complex one. Second, (18) not only moves NPs around, but it also inserts passive morphology (*be + en*) as well as a *by*-phrase. Third, in contrast to (18), an application of Move α (where $\alpha = \text{NP}$) allows any NP to move anywhere. Thus, the Move α analysis of Passive *factors out* the NP movements from the other features of the passive rule. This effectively eliminates the construction-based conception of rules characteristic of the earlier Standard Theory and replaces it with a far more abstract conception of a G rule; it effectively treats earlier construction-based rules as interactions and combinations of simpler ones.

These simplifications, though theoretically desirable, create empirical problems. How? Rules like Move NP left to themselves wildly overgenerate, deriving all sorts of ungrammatical structures (as we illustrate below).²⁹ GB addresses this problem in a theoretically novel way. It eliminates the empirically undesirable consequences by *enriching* UG. In particular, GB theory targets two related dimensions: it *simplifies* the rules of G_L while *enriching* the structure of FL/UG. Let's consider this in more detail.

Move α is the simplest possible kind of movement rule. It says something like "move anything anywhere." Languages differ in what values they allow α to assume, thus allowing for a natural locus of cross-linguistic variation. So, for example, English moves *Wh* words to the front of the clause to form interrogatives. Chinese doesn't. In English α can be *Wh*, in Chinese it cannot be. Or Romance languages move verbs to tense, while English doesn't. Thus in Romance α can be V, while in English it can't. And so on. Again, while so simplifying the rules has some appeal, the trick is to simplify without incurring the empirical costs of overgeneration. GB achieves this (in part) via Trace Theory, which is itself a consequence of the Projection Principle, a more general conservation principle that bars derivations from *losing* syntactic information. Here's the story.

²⁹ One mark of an ungrammatical structure is that the sentences that coincide with these structures are judged unacceptable. However, '(un)grammatical' is a predicate of syntactic structures (and it therefore carries theoretical content) while '(un)acceptable' is a descriptive predicate applied to data. These two terms are often used interchangeably, which can result in quite a bit of unnecessary confusion. Speakers don't have judgments concerning grammaticality, though they are expert concerning acceptability. What Chomsky discovered is that acceptability judgments by native speakers can be used to investigate the grammaticality of linguistic structures. But – and this is important – there are various sources for unacceptability besides ungrammaticality and the two notions need not swing together, though they often do (which is why querying for acceptability is an excellent source of data concerning grammaticality).

In the GB framework, Trace Theory implements the general computational principle that derivations be monotonic. For example, if a verb has a transitive syntax in the base, then it must retain this transitive syntax throughout the derivation. Or, put another way, if some NP is an object of a V at some level of representation, the information that it was must be preserved at every subsequent level of representation. In a word, information can be created but not destroyed; that is, G rules are structurally monotonic in the sense that the structure that is input to a rule is preserved in the structure that is output from that rule. Within GB, the name of this general computational principle is the *Projection Principle*, and the way it is formally implemented is via Trace Theory.

This monotonicity condition is a novelty. Operations within the prior Standard model are not monotonic. To illustrate, take the simple case of Raising to Subject, which can be schematized along the lines of (19):³⁰

- (19) $X - T(\text{ense})1 - Y - NP - T(\text{ense})2 - Z \rightarrow X - NP - T1 - Y - T2 - Z$
(T2='to')

This rule can apply in a configuration like (20a) to derive a structure like (20b):³¹

- (20) a. $[_{TP} [_{T} \text{present}] [_{VP} \text{seem} [_{TP} \text{John} [_{T} \text{to}] [_{VP} \text{like Mary}]]]]$
b. $[_{TP} \text{John} [_{T} \text{present}] [_{VP} \text{seem} [_{TP} [_{T} \text{to}] [_{VP} \text{like Mary}]]]]$

Note that the information that *John* had been the subject of the embedded clause prior to the application of (19) is lost, as the embedded TP in (20b) no longer has a subject like it does in (20a).

As noted, Trace Theory is a way of implementing the Projection Principle. How exactly? Movement rules in GB are *defined* as operations that leave traces in positions from which movement occurs. Given Trace Theory, the representation of (20a) after Raising has applied is (21):

- (21) $[_{TP} \text{John}_i [_{T} \text{present}] [_{VP} \text{seem} [_{TP} t_i [_{T} \text{to}] [_{VP} \text{like Mary}]]]]$

Here t_i is a trace of the moved *John*, the co-indexing coding the fact that *John* was once in the position occupied by its trace. As should be clear, via traces, movement now preserves prior syntactic structure (the subject position in (20a) is retained in (21)). As noted, this kind of information-preserving

³⁰ Ditto with Passive. The Passive rule (18) syntactically detransitivizes a transitive verb, viz. the V in the structural output of the rule no longer has a direct object, though it did have one in deep structure. We leave the details as an exercise for those inclined to build up their syntactic muscles.

³¹ More exactly, (20a) informs us that we can factor the string *present-seem-John-to-like-Mary* as $X - T - Y - NP - T - Z$ (note X, Y, Z are null in this case) and so change it to a string that will have the structure in (20b) and so the factorization displayed on the right side of the arrow in (19).

principle (i.e. that grammatical operations cannot destroy structure) becomes a staple of all later theory.³²

Trace Theory is GB's first step towards simplifying G rules. The second bolder step is to propose that traces require licensing, and the third boldest step is to execute this by using traces to unify binding and movement. Specifically, binding theory expands to include the relation between a moved α and its trace. Executing this unification, given other standard assumptions (particularly that D-structure represents pure GF- θ) requires rethinking binding and replacing construction-specific rules like Reflexivization in favor of a more abstract way of coding the anaphoric dependency. Again, let's illustrate.

Say we treat Raising as just an instance of Move α , then we need a way of preventing the derivation of unacceptable sentences like (22a) from sentences with the underlying structure in (22b).

- (22) a. *John seems likes Mary
 b. [_{TP} [_T present] [_{VP} seem [_{TP} John [_T present] [_{VP} like Mary]]]]

Now, given a rule like (19), this derivation is impossible. Note that the embedded T is not *to* but *present*. Thus, (19) cannot apply to (22b) as its structural description is not met (i.e. the structural description of (19) codes its inapplicability to (22b) thus preventing the derivation of (22a)).³³ But, if we radically simplify movement rules to "move anything anywhere" (i.e. Move α), the restriction coded in (19) is not available and overgeneration problems (e.g. examples like (22a)) emerge.

To recap, given a rule that simply says "Move NP," there is nothing preventing the rule from applying to (22b) and moving *John* to the higher subject position. The unification of movement and binding via Trace Theory serves to prevent such overgeneration. How? By treating the relation between a trace and its antecedent as syntactically identical to that between an antecedent and a reflexive anaphor. Specifically, if the trace in (23a) is a kind of "reflexive" then the derived structure is illicit as the trace left by movement is not bound. In effect, (23a) is blocked in basically the same way that (23b) is.

- (23) a. [_{TP} John₁ [_T present] [_{VP} seem [_{TP} t₁ [_T Present] [_{VP} like Mary]]]]
 b. [_{TP} John₁ [_T present] [_{VP} believe [_{TP} he-self/him-self₁ [_T Present] [_{VP} like Mary]]]]

Let's pause and revel (maybe even wallow!) in the logic on display here: If derivations are monotonic (i.e. obey the Projection Principle) then when move NP (i.e. Move α , with $\alpha=NP$) applies it leaves a trace in the moved-from

³² Indeed, the principle gets strengthened in one important way. Traces make way for copies, so that an occurrence of a moved expression replaces earlier traces. See Chapter 2 for discussion.

³³ Actually, the structure underlying (22a), but you get the point.

position thereby preserving the syntactic structure. Further, if the relation between a moved α and its trace is the same as an anaphor to its antecedent, then the licensing principles that regulate the latter must regulate the former.³⁴ So, *simplifying* derivations by making them monotonic and *unifying* movement and binding allows for the radical *simplification* of movement rules (i.e. to a Move α format) without *any empirical costs*. In other words, simplifying derivations, unifying the modules of the grammar (always a theoretical virtue if possible) serves to advance the simplification of its rules.³⁵ The GB virtues of simplification and unification are retained as regulative ideals in contemporary Minimalist thinking.

That's the basic idea. However, we need to consider a few more details, as reducing (23a) to a binding violation requires reframing the theory of binding. More specifically, it requires that we abstract away from the specifics of the binding constructions and concentrate on the nature of the relations they specify. Here's what I mean.

The Lees–Klima rule of Reflexivization contrasts with rules like Raising in that the former turns the lower “dependent” into a reflexive while the latter deletes it. Moreover, whereas Reflexivization is a rule that applies exclusively to clause-mates, Raising only applies between clauses. Lastly, whereas Reflexivization is an operation that applies between two identical lexical items (viz. two items introduced by lexical insertion in Deep Structure), Raising does not (in contrast to Equi, for example).³⁶ From the perspective of the Standard Theory, then, Raising and Reflexivization could not look more different and unifying them would appear unreasonable. The GB theory, in contrast, by applying quite generally to *all* nominal expressions, highlights the relevant *dependencies* that they can enter into (and that differentiate them) and does not get distracted by other (irrelevant) features of the constructions (like their differing morphology or even their formal etiology).

³⁴ Well, more specifically, the unification is possible if we reanalyze reflexivization as Principle A, as we do below. Pardon the imprecision.

³⁵ There was another interesting theoretical consequence of this rule simplification. Consider a sentence like (i):

(i) John was believed (by Fred) to be intelligent.

Within the Standard Theory such a sentence raised a curious question: is (i) derived by Raising *John* or by Passivizing the sequence (*Fred*) *believed John* (which is part of the structure underlying (i))? Note that *John* is an embedded subject and so it has raised. But the predicate over which it has moved is passivized so it has also been passivized. So which is it? Given the replacement of construction-specific rules by Move α , the question disappears. Both Passivization and Raising are instances of Move α and so the problem of deciding disappears. This is a purely theoretical virtue of recasting the issues in GB terms. It answers a why-question by showing that correctly framed it is ill-posed. A confession: I *love* these kinds of explanations!

³⁶ Equi-NP Deletion (aka Equi) is the rule that underlies the derivation of sentences like (i), where the matrix subject is understood as determining that value of the embedded subject:

(i) John tried to win (=John tried so that John wins)

Let me state this another way. The construction specificity of the rules in the Standard Theory has the consequence that most rules look formally different from one another. Thus, unifying Reflexivization and Equi or Equi and Movement does not seem particularly plausible when one considers the formal features of the rules. Only Trace Theory and the abstractions it introduces makes the potential similarities between these various constructions readily visible.

In particular, GB unifies movement and binding via Trace Theory by recasting the rule of Reflexivization. Recasting Reflexivization constructions as Principle A effects allows FL to treat the *relation* between the nominal that has moved and the trace left by this movement and the *relation* between the reflexive and the nominal that serves as its antecedent as the same relation. The GB accomplishes this by treating A-traces and reflexives as morphemes of the same kind, subject to the same licensing condition. Thus, critically, this unification requires moving from binding rules like Reflexivization to licensing conditions like Principle A. Let's consider how.

GB binding theory (BT) divides all nominal (overt) expressions into three categories and associates each with a licensing condition. The three are (i) anaphors (e.g. reflexives, reciprocals, PRO), (ii) pronominals (e.g. pronouns, PRO, pro) and (iii) R-expressions (everything else). BT regulates the interpretation and distribution of these expressions. It includes three conditions, Principles A, B and C, and a specification of the relevant domains and licit dependencies:

- (24) GB Binding Principles:
 A. An anaphor must be bound in its minimal domain
 B. A pronoun must be free in its minimal domain
 C. An R-expression must be free
- (25) α is the minimal domain for β if α is the smallest clause (TP) with a subject distinct from β .³⁷
- (26) An expression α is bound by β iff β c-commands α , and β and α are co-indexed.

These three principles together capture all the data we noted in (4)–(10). Let's see how. The relevant examples are recapitulated in (27). (27a,b,e,f) illustrate that bound reflexives and pronouns are in complementary distribution. (27c,d) illustrate that R-expressions cannot be bound at all.

- (27) a. John_i likes himself/*him_i
 b. John_i believes Mary likes *himself/him_i

³⁷ Things can be (and are) more complex than this. What counts as a clause might differ (TP or CP) and one can extend BT to nominal domains as well with an extension of the notion "subject." We will put these complications aside here and assume that clause means TP.

- c. *I expect himself_i to like John_i
- d. *He_i expects me to like John_i
- e. John_i believes *himself/he_i is intelligent
- f. John_i believes himself/*he to be intelligent

How does BT account for these data? Reflexives are categorized as anaphors and so subject to Principle A. Thus, reflexives must be bound in their minimal domains. Pronouns are pronominals subject to Principle B. Thus, a pronoun *cannot* be bound in its minimal domain. Thus given BT, pronouns and reflexives must be in complementary distribution.³⁸ This accounts for the data in the mono-clausal (27a) and the bi-clausal data in (27b). It also accounts for the data in (27f). The structure is provided in (28):

(28) [TP₁ John Present [v_{VP} believe [TP₂ himself/he to be intelligent]]]

The minimal domain for *himself/he* is the matrix TP1. Why? Because of (25), which requires that the minimal domain for α must have a subject distinct from α . But *himself/he* is the subject of TP2. The first TP with a distinct subject is the matrix TP1 and this becomes its binding domain. In TP1 the anaphor must be bound and the pronoun must be free. This accounts for the data in (27f).

(27e) requires some complications. Note that we once again witness the complementary distribution of the bound reflexives and pronouns. The minimal domain should then be the embedded clause if BT is to explain these data. Unfortunately, (25) does not yield this. This problem received various analyses within GB, none of which proved entirely satisfactory. The first proposal was to complicate the notion ‘subject’ by extending it to include the finite marker (which has nominal *phi/φ* (i.e. person, number, gender) features).³⁹ This allows the finite T to be a subject for *himself/he* and their complementary distribution follows given the contrary requirements that A and B impose on anaphors and pronominals.⁴⁰

Principle C excludes (27c,d), as in both cases *he/himself* binds *John*. (27c) also violates Principle A.

³⁸ Recall: This fact was also accounted for by the Lees–Klima theory surveyed in Section 1.1, albeit in a very different way. Effects constant, theories different. Just what we want.

³⁹ This effectively analogizes agreement markers to pronominals. Pronouns are also just bundles of person, number and gender features. Later approaches to binding were able to eliminate this assumption. See note 41.

⁴⁰ A later proposal (see Chomsky 1986a) suggested a more radical approach. Chomsky, following a proposal by Lebeaux, assumed that to be bound, reflexives must (covertly) adjoin to a position proximate to the antecedent (in this case matrix T, akin to what Romance reflexives do overtly). Such a movement is (plausibly) an ECP violation. Observe, that accommodating the data in this way effectively points to reducing binding to conditions on movement rather than movement to conditions on binding (see Hornstein (2008) for an elaboration of this idea, which, to be fair, has not received the traction Hornstein had hoped it would).

For what it is worth, neither the fix in the text nor the one in this note is without additional problems, but both serve for current purposes, which is to derive the contrast in (27e).

In sum, BT accounts for the same binding effects the earlier L&K theory does, though in a very different way. It divides the class of nominal expressions into three groups, abstracts out the notion of a binding domain, and provides universal licensing conditions relevant to each.⁴¹ As with the GB movement theory, most of the BT is plausibly part of the native (“universal”) structure of FL and hence need not be acquired on the basis of PLD. What the learner needs to determine (i.e. acquire) is what group a particular nominal expression falls into. Is *each other* an anaphor, pronominal or R-expression? Once this is determined, where it can appear and what its antecedents can be follow from the innate architecture of FL. Thus, BT radically *simplifies* FL by distinguishing what binding applies to from what binding is, and this has a natural interpretation in terms of acquisition: knowledge of what belongs in which category must be acquired, while knowledge of what the relevant categories are and how something in a given category behaves is part of FL and hence innate.⁴²

With this as background, let’s return to how GB allows for the unification of binding and movement via Trace Theory. Recall that BT divides *all nominal expressions* into three groups. Traces are nominal expressions (viz. $[_{NP} e]_i$) and so it is reasonable to suppose that they too are subject to BT. Moreover, as traces determine the θ -roles of their antecedents, they *must* be related to them for semantic reasons. This would be guaranteed were traces treated like anaphors falling under Principle A. This suffices to assimilate (23a) to (23b) and so it closes the explanatory circle.⁴³

So, by generalizing binding considerations to *all* nominal expressions, and by recasting the binding theory so as to showcase binding domains and binding dependencies, GB makes it natural to unify movement and binding by categorizing traces as anaphoric nominal expressions (a categorization that would be part of FL, hence innate and so in no need of learning). So, simplifying derivations with the Projection Principle leads to Trace Theory, which in turn allows

⁴¹ It is worth noting that the GB version of the binding theory requires no extrinsic ordering (i.e. stipulative ordering) of the operations that license reflexives and those that license pronouns. One can “apply” A and B in any order with the same results, in contrast to (8) and (9). As stipulation is never theoretically welcome, its elimination in the repackaging of the BT is a theoretical step forward.

⁴² Chomsky has often noted that GB eliminates constructions as fundamental units of analysis. He is, once again, entirely correct. Note that both the simplification of movement and binding and their unification builds on distinguishing the relation itself (e.g. A-binding, *Wh*-movement) from the relata involved in the relation (a reflexive anaphor and its DP antecedent, a WH DP and the trace left behind). This is what frees GB operations from the constructional specificity of earlier Standard Theory rules and which allows for the unification of movement and binding central to the LGB theory. Thus, the elimination of constructions as fundamental units of analysis is key to GB’s theoretical success.

⁴³ It was reasonably assumed that FL/UG categorized the various flavors of empty categories. Hence which category A-traces (under BT-A) or *Wh*-traces (under BT-C) fell into did not have to be learned as their categorization is given innately.

for the unification of movement and binding, which in turn leads to a radical simplification of movement transformations, all without any apparent diminution of empirical coverage.

Let me add one more point concerning linguistic flexibility: recall that one of the big questions concerning language concerns its acquisition by kids on the basis of relatively simple input. The GB story laid the foundations for an answer: The rules were easy to learn because where languages vary the differences are easy to acquire on the basis of simple PLD (e.g. is $\alpha = \text{NP}$ or V or *Wh* or ...). GB factors out the *intricacies* of the Standard Theory rules (e.g. ordering statements and clause-mate conditions) and makes them intrinsic features of FL; hence a child can know them without having to acquire them via PLD. Thus, not only does GB radically simplify and unify the operations in the Standard Theory, a major theoretical accomplishment in itself, it also provides a model for successfully addressing what has been called “Plato’s Problem”: How does knowledge of Gs arise in native speakers despite the relative paucity of data available in the PLD to fix their properties? In other words, how can kids acquire Gs despite the impoverished nature of the PLD?

Let’s end this section. We have illustrated how GB, building on earlier research (and conserving its discovered effects), constructed a more principled theory of FL. Though we looked carefully at binding and movement, the logic outlined above was applied much more broadly. Thus, phrase structure was simplified in terms of X’-theory (pointing towards the elimination of PS-rules altogether in contemporary theory) and island effects were unified under the theory of Subjacency. The latter echoed the discussion above in that it consolidated the view that T-rules are very simple and not construction centered. Rather constructions are complexes of interacting simple basic operations. The upshot is a rich and articulated theory that describes the fixed structure of FL in terms of innate principles of UG. In addition, the very success of GB theory opens a *further* important question for investigation. Just as research in the Standard Theory paves the way for a fruitful consideration of linguistic universals and “Plato’s Problem,” the success of GB allows for a consideration of “Darwin’s Problem”: How could something like FL have arisen in the species so rapidly and remained so unchanged since its inception? We turn to this in the next chapters, but first, as promised, a (by no means exhaustive) list of effects that sixty years of Generative Grammar research has unearthed.

1.5 Some Effects Generative Grammar Has Discovered over the Last Sixty Years

Here is a partial list of some of the effects that are still being widely investigated (both theoretically and empirically) within Generative research. Some of these effects can be considered analogous to “laws of grammatical structure”

which serve as probes into the inner workings of FL. As in the case of L&K's binding proposal, the effects comprise both negative and positive data and they have served as explanatory targets (and benchmarks) for theories of FL.

These effects also illustrate another distinguishing mark of an emerging science. In the successful sciences, most of the data is carefully constructed, not casually observed. In this sense, it is not "natural" at all, but factitious. The effects enumerated here are similar. They are not thick on the conversational ground. Many of these effects concentrate on what *cannot* exist (i.e. negative data). Many are only visible in comparatively complex linguistic structures and so are only rarely attested in natural speech or PLD (if at all). Violations of the binding conditions such as *John believes himself is intelligent* are never attested outside of technical papers in Generative syntax. Thus, in Generative research (as in much of physics, chemistry, biology, etc.) much of the core data used to probe FL is constructed, rather than natural.⁴⁴ To repeat, this is a hallmark of modes of investigation that have made the leap from naturalistic observation to scientific explanation. The kinds of data that drive Generative work are of this constructed kind.⁴⁵

Here, then, is a *partial* list of some of the more important effects that Generative Grammar has discovered.⁴⁶

- (29) A partial list of empirically discovered laws of grammar
1. Island effects
 - a. Weak island effects
 - b. Strong island effects
 2. Crossover effects
 3. Control vs Raising effects
 4. Minimal distance effects in control configurations
 5. Binding effects (A-effects-B-effects)
 6. Cyclicity effects
 7. Principle C-effects: an anaphoric element cannot c-command its antecedent
 8. CED (condition on extraction domain) effects
 - a. Subject condition effects
 - b. Adjunct condition effects
 9. Fixed subject effects
 10. Unaccusativity effects
 11. Connectedness effects
 12. Obligatory control vs non-obligatory control effects
 13. The subject orientation of long-distance anaphors

⁴⁴ See Cartwright (1999) for discussion of this in the context of the "real" sciences.

⁴⁵ Constructed data are generally more robust than naturalistic data, as Cartwright (1999) observes. Furthermore, it allows for investigations to be more systematic by allowing researchers to put their own questions to nature and make it answer these rather than simply waiting until nature voluntarily gives up its secrets.

⁴⁶ For a similar annotated list, see D'Alessandro (2019).

14. Case effects
15. Theta Criterion effects (Principle of Full Interpretation)
16. NPI (negative polarity item) licensing effects
17. Phrasal headedness effects
18. Clause-mate effects
19. Expletive-associate locality effects
20. Parasitic gap effects
21. Pro-drop effects
22. ECP (Empty Category Principle) effects
23. Weakest crossover effects
24. Coordinate structure constraint
 - a. ATB (across-the-board) effects
25. Ellipsis effects
26. A-movement/scrambling obviating WCO (weak crossover) effects
27. Intervention/minimality effects
28. Constituency effects
29. Scope reconstruction effects
30. Lexical integrity effects
31. Psych verb effects
32. Double object construction effects
33. Predicate-internal subject effects

As in the case of the L&K binding proposal outlined above, just describing these effects involves postulating abstract rules that derive natural language expressions and abstract structures that describe them. Thus, each effect comes together with sets of positive and negative examples and rules/restrictions that describe these data. As in any scientific domain, simply describing the effects already requires quite a bit of theoretical apparatus (e.g. what's an island, what's a deletion rule, what's the difference between A- and A'-movement, what's case, what's a clause, etc.). And, as is true elsewhere, the discovery of such effects sets the stage for the next stage of inquiry: explaining why we find these particular effects and seeing what these explanations can tell us about the structure of FL.

1.6 The Minimalist Program and a Novel Research Question

Where are we? Here is a quick recap.

First on the agenda was the problem of linguistic creativity, the fact that native speakers of a given language L have the capacity to understand an unbounded number of different expressions of L. This fact raised the obvious question: How is this possible? The answer: This capacity supervenes on having an internalized finitely specified G that recursively characterizes the linguistic objects of L. So, (part of) the explanation for the fact that native speakers are linguistically creative in their language L is to give a recursive characterization of what constitutes a possible object of L and treat this recursive specification as part of a native speaker's mental make-up.

More specifically, we reviewed how the first period of syntactic research investigated how grammars might be structured so that they could generate an unbounded number of distinct hierarchically organized objects. The research strategy was to propose specific Gs for given Ls whose interacting rules yielded interesting empirical coverage, generating both a fair number of acceptable sentences and *not* generating an interesting number of unacceptable sentences. In the process, Generativists discovered an impressive number of effects that served as higher-level targets of explanation for subsequent theory. To say the same thing a little more pompously, early Generative Grammar discovered a bunch of “effects” which catalogued deep-seated generalizations characteristic of the products of human Gs. These effects sometimes fell together as “laws of grammar” taken to reflect the built-in design features of FL. More simply, these effects are plausible reflections of the properties of FL and so can be used to explore the structure of FL.

Or to say this slightly differently, the success in adumbrating properties of particular Gs led to a second further stage of research that built on this success, and which targeted a related, yet different, question: How must humans be built so that they can acquire Gs with the properties we discovered Gs to have? The project, in effect, comes down to specifying the class of possible human Gs.

Importantly, the project of adumbrating the range of possible Gs becomes fruitful once we have a budget of empirically plausible properties of actual Gs! Without some decent examples of actual Gs with their identified properties, it makes little sense to ask how to delimit such Gs. To say this another way, we need to *empirically* bound the domain of inquiry to make it tractable (and worth investigating), and this is why investigating the properties of FL (properties that will serve to limit the range of possible Gs) only becomes a fertile pursuit once Generativists have identified some features of actual Gs to serve as targets of explanation.

So, we have some empirically plausible features of Gs and we want a theory of FL (or UG) to explain why we find Gs with these properties and not others. Plato’s Problem served as an additional boundary condition on this line of inquiry into FL/UG. Plato’s Problem is the observation that what native speakers know about their languages far exceeds what they could have learned about it by examining the PLD available to them in the course of their G acquisition. Conceptually, addressing Plato’s Problem in the context of a budget of identified G effects suggested a two-pronged attack: first, radical simplification of the rules that Gs contain, and second, enrichment of what FL/UG brings to the task of G acquisition. Eliminating the complexity built into *Aspects*-style rules and factoring out some simple very general operations like Move α made the language-particular rules that children acquired *easier* to acquire. This simplification, however, threatened generative chaos by allowing for massive over-generation of ungrammatical structures. The theoretical task was to prevent

this. This was accomplished by enriching the innate structure of FL/UG in principled ways. The key theoretical innovation was Trace Theory motivated by the idea that derivations are information preserving (monotonic). Traces simplified derivations by making them structure preserving, and this further allowed for the unification of movement and binding. These theoretical moves together addressed the overgeneration problem.⁴⁷

This line of inquiry coalesced around a “standard” model of FL/UG (i.e. GB). In particular, GB provided a substantive model of FL/UG and thereby set the stage for contemporary minimalist investigations. More specifically, just as the success of early Generative inquiry into language-particular Gs allowed us to fruitfully address the question why we have *these* Gs and not others (answer: because we have a GBish FL/UG), adumbrating an empirically substantive conception of FL/UG now allows Generativists to ask the next obvious question: Why does FL/UG have *these* GBish properties and not others? Moreover, just as asking the question concerning the limits on human Gs would have been premature (and so idle) without first discovering some of the empirical features of actual Gs, so too investigating why we have the FL/UG we actually have (rather than another with other possible organizing principles) would have been premature without first having some reasonable empirically grounded theory of FL/UG like the one GB delivered.⁴⁸

⁴⁷ A remark for the cognoscenti: The prettiest possible theory, one that Chomsky advanced in early GB, failed to hold empirically. The first idea was, effectively, to treat *all* traces as anaphoric. Though this worked very well for A-traces, it proved inadequate for A'-traces, which seemed to function more like R-expressions than anaphors (or at least the “argument” A'-traces did). A virtue of assimilating A'-traces to R-expressions is that it led to an explanation of strong crossover effects in terms of Principle C. Unfortunately, it failed to explain a range of subject-object and argument-adjunct asymmetries that crystalized as the ECP. These ECP effects led to a whole new set of binding-like conditions (so-called “antecedent government”) that did not fit particularly comfortably with other parts of the theory. Indeed, the bulk of GB theory in the last part of the second epoch consisted in investigations of the ECP and various ways of trying to explain the subject-object and argument-adjunct effects. Three important ideas came from this work. First, that the domains relevant for ECP effects are largely identical to those relevant for subjacency effects. Second, that ECP effects really do come in two flavors, with the subject-object cases being quite different from the argument-adjunct cases. Third, relativized minimality. This was an important idea due to Rizzi (1990b), and one that fit very well with later minimalist conceptions. This said, ECP effects, especially the argument-adjunct asymmetries, have proven theoretically refractory and still remain puzzling, especially in the context of minimalist theory. Importantly, recent empirical work (see Lu et al. 2020) suggests that perhaps the argument-adjunct asymmetry does not exist, which would explain its refractory theoretical obstinacy. I discuss this further in Chapter 8.

⁴⁸ I am focusing on GB here because that is the theory whose basic properties the Minimalist Program (as I understand it) has tried to explain. However, there were many (putative) alternative conceptions of FL/UG developed in the period from the mid-1970s to the mid-1990s. In my opinion, these were far less different from GB than generally advertised (there is, after all, a premium gained by claiming novelty). However, the kind of argument outlined in the chapters that follow can be pursued using any of these conceptions of FL/UG, though, no doubt, the details will differ.

I want to emphasize, re-emphasize and re-re-emphasize this point before getting into some details in the following chapters. I have often heard the claim that Minimalism offers nothing new to the Generative enterprise, methodologically speaking. I agree with part of this. Generative Grammar has always prized explanation, and so the hallmarks of explanation (i.e. deriving the properties one wants explained from simple, elegant, theories that derive them) have also always been valued. To wit: We *explain* the basic features of a given native speaker's linguistic productivity in L by showing how they result from a G_L that generates the unboundedly many different hierarchical $\langle m, s \rangle$ s characteristic of that L (or more exactly, that coincides with a native speaker's "sense" of L) and does not generate any pairs inconsistent with a native speaker's competence in L. Similarly, we *explain* why we find the $G_{L,s}$ that are empirically attested by showing that our theory of FL/UG (e.g. GB) derives $G_{L,s}$ with these properties and does not derive any without them. In both cases we prize simple elegant theories over more complex inelegant ones for the reasons that scientific inquiry has always prized the former over the latter. In the first two periods of Generative Grammar, the methodology remained constant, *even as the questions addressed changed*.

So too as regards the current minimalist stage of Generative inquiry. We still want simple, elegant theories, but now their derivational target is (roughly) GB and the laws of grammar (and concomitant effects) it adumbrates. These are what minimalist theories aim to explain. Or, to be more precise, we now want theories that derive the theoretical principles of GB and/or the associated effects that these GB principles aimed to explain. Note, we have the same methodological standards as ever (simplicity, elegance, naturalness), *but we are now entertaining a different explanandum*. Moreover, as we have noted above, targeting the principles of GB and its associated laws of grammar for explanation only really makes sense if we take GB to be reasonably well grounded, both empirically and theoretically.

Let me put this point more broadly: There is a reason Minimalism was a brainchild of the mid-1990s. ***It took that long to make it a substantive project.*** Minimalism awaited a plausible theory of FL/UG, which in turn awaited plausible G_s of particular L_s . And all of that took about forty years to develop. By the mid-1990s Generative Grammar had an empirically viable (though not perfect) theory of FL/UG (i.e. GB) and so it made sense to investigate its properties and ask why they are the way they are.

So, is Minimalism just the same old, same old or something new? And by now I hope you know the right answer: YES! It is both nothing new and something very different. That is what makes the Minimalist Program (MP) interesting.

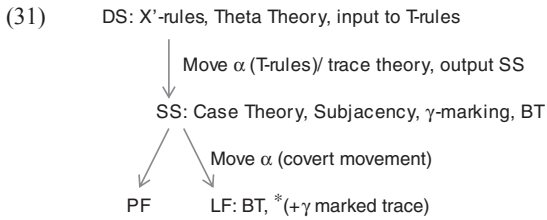
1.7 The Minimalist Program: Explaining the Properties of GB

How should one go about explaining why FL/UG has the properties it does? By deriving them from simpler, more natural, more economical assumptions. And this entails assuming that whatever GB's merits, it is not the *fundamental* theory of FL. Standard methodological considerations lend credence to this last assumption. GB is simply too complex to be fundamental. And also too linguistically sui generis. Here's what I mean.

For Generativists, FL is a *human-specific* cognitive capacity. This entails that the human-specific linguistic capacity evolved from pre-human ancestors that were not linguistically proficient (at least not the way we are). In other words, FL, the capacity to acquire and deploy Gs, is cognitively novel in humans. In this evolutionary context, GB is a problematic account of FL's basic properties precisely because it is too "complex" and too linguistically specific. In particular, the more FL's properties are linguistically bespoke (rather than cognitively and computationally generic), and the more complex the internal organization of FL, the harder it is to explain how it arose from non-linguistic minds (i.e. minds bereft of FLs). Put more positively, the simpler the structure of FL, and the less linguistically specific its operations and principles, the easier it should be to explain how they could have arisen from a-linguistic minds. And this line of thought has an immediate consequence and suggests a concrete research program. The consequence is that though GB might be a good description of FL, it cannot be the *fundamental* theory of FL. The fundamental theory *must* be simpler and less linguistically specific. The program is to develop such a simpler theory that has (roughly) GB and its properties as limit consequences. Let's flesh these general points out bit.

Within GB, FL is both very complex and the proposed innate principles and operations are very linguistically specific. The complexity is manifest both in the overall modular architecture of the basic GB theory and in the specific principles and operations characteristic of each module. (30) and (31) reiterate the basic structure of the theory.

- (30) a. X'-theory of phrase structure
- b. Case
- c. Theta
- d. Movement
 - i. Subjacency
 - ii. ECP
- e. Construal
 - i. Binding
 - ii. Control



Though some critical relations crosscut (many of) the various modules (e.g. government), the modules each have their own special features. For example, X'-theory traffics in notions like specifier, complement, head, maximal projection, adjunct and bar-level. Case theory also singles out heads but distinguishes between those that are case assigning and those that require case. There is also a case filter, case features and case assigning configurations (government). Theta theory also uses government but for the assignment of θ -roles, which are assigned in D-structure by heads and are regulated by the Theta Criterion, a condition that requires every argument to get one and at most one θ -role. Movement exploits another set of concepts and primitives: bounding node/barrier, escape hatch, subjacency principle, antecedent government, head government, γ -marking, γ -checking and more. Last, the construal rules come in four different types: one for PRO, one for local anaphors like reflexives and reciprocals, one for pronouns and one for all the other kinds of DPs, dubbed R-expressions. There is also a specific licensing domain for anaphors and pronouns, indexing procedures for the specification of syntactic antecedence relations and hierarchical requirements (c-command) between an antecedent and its anaphoric dependent. Furthermore, all of these conditions are extrinsically ordered to apply at various derivational levels specified in the Y-model.⁴⁹

If the information outlined in (30) and (31) is on the right track, then FL is richly structured with very domain-specific (viz. linguistically tuned) information. And though such linguistic specificity is a positive with regard to addressing Plato's Problem, it raises difficulties when trying to address Darwin's Problem. Indeed, the logic of the two problems has them largely pulling in opposite directions. A rich linguistically specific FL plausibly eases the child's task by restricting what the child needs to use the PLD to acquire. However, the more cognitively *sui generis* FL is, the more complicated the evolutionary path to FL. Thus, from the perspective of Darwin's Problem, we want the operations

⁴⁹ By 'extrinsically' we mean that the exact point in the derivation at which the conditions apply is empirically motivated but theoretically stipulated.

and principles of FL to be cognitively (or computationally) general and very simple. It is this tension that the Minimalist Program aims to resolve.

The tension is exacerbated when the evolutionary timeline is considered. The consensus opinion is that humans became linguistically capable about 100,000 years ago and that the capacity that evolved has remained effectively unchanged ever since.⁵⁰ Thus, whatever the addition to a-linguistic minds that made them “language ready,” it must have been relatively minor (the addition of at most one or two linguistically bespoke operations/principles). Or, putting this another way, our FL is what you get when you wed (at most) one (or two) linguistically specific features with a cognitively a-linguistic generic brain.

Navigating the narrows between Plato’s Problem and Darwin’s Problem suggests a twofold strategy: (i) Simplify GB by unifying the various FL-internal modules and (ii) Show that this simplified FL can be distilled into largely general cognitive/computational parts plus (at most) one linguistically specific one.⁵¹

Before proceeding, please note yet again that *GB is the target of explanation*. In other words, the Minimalist Program takes GB to be a good approximate model of FL’s fine structure. It is not fundamental, but it is still very good, in that Minimalism assumes that GB has largely correctly identified (and described) phenomena (laws) that directly reflect the innate structure of FL. If MP is realizable, then FL is less linguistically parochial than GB supposes, even though it has operations and principles of the kind that GB adumbrates. If MP is realizable, then FL exploits many generic operations and principles (i.e. operations and principles not domain restricted to language) in its linguistic computations and uses these for linguistic ends. On this view, Minimalism takes GB’s answer to Plato’s Problem to be largely correct though it disagrees with GB about how domain-specific the innate architecture of FL is. Borrowing terminology common in physics (perhaps grandiosely), Minimalism takes GB to be a good *effective* theory of FL but denies that it is the *fundamental* theory of FL. A useful practical consequence of this is to take the principles of GB to be targets for derivation by the more fundamental principles that minimalist theories will discover.

That’s the program and that’s how the Minimalist Program fits into the overall Generative research program. Has the program been successful? I believe it has been triumphantly so. I try to make this case in the chapters that follow.

⁵⁰ See Chomsky (2018) and references therein.

⁵¹ We distinguish cognitively general from computationally general for there are two possible sources of relief from GB specificity. Either the operations/principles are borrowed from other pre-linguistic cognitive domains or they arise as a general feature of complex computational systems as such. Chomsky has urged the possibility of the second in various places, suggesting that these general computational principles might be traced to as yet unknown physical principles. However, for research purposes, the important issue lies with the non-linguistic specificity of the relevant operations and principles, not whether they arise as a result of general cognition/biology or natural physical law.