Emergence is the process of complex pattern formation from more basic constituent parts or behaviors, and manifests itself as an emergent property of the relationships between those elements. For a phenomenon to be termed emergent it should generally be unpredictable from a lower level description. (from Wikipedia entry on Emergence)

Exaptation is a biological adaptation where the biological function currently performed by the adaptation was not the function performed while the adaptation evolved under earlier pressures of natural selection. Classic example: The flying function exapted thermoregulatory function of wings.

ECONOMY IN THE MINIMALIST PROGRAM

1. BIOLINGUISTICS

ECONOMY IN BIOLOGICAL SYSTEMS

"Behind every phenomenon in our universe, we can find a maximum or minimum rule"

(Leonhard Euler [1707-83]; from Jenkins 1997; Hildebrandt & Tromba, 1996:34)

ECONOMY in biological systems serves as an OPTIMUM PRINCIPLE using symmetry. For instance, in the design of the bee honeycomb, selecting from hexagonal, triangular or square shapes, bees choose the form that maximizes content (hexagonal).

NATURAL LANGUAGE VS. BIOLOGICAL SYSTEMS

- Language is rule-base system, comparable to vision in mammal, insect communication....
- Study of language comparable to study of limitations on phenotypic variability

- Syntax: unlike other biological systems, language is a system of discrete infinity. Language takes objects already constructed and assembles new ones (Merge; recursion).
- Semantics: unlike other biological communication systems, language can refer and manipulate non-existing or temporally or spacially dislocated objects (unicorns; the Battle of Hastings; an election in Ghana; Hockett’s ‘displacement property’).

TWO EARLIER VIEWS ON THE EVOLUTION OF LANGUAGE

- Alfred Russell Wallace (co-founder of evolutionary biology; contra Darwin): Language cannot be accounted for in terms of variation and natural selection
- Salvador Luria (Nobel Prize in Physiology or Medicine 1969): communicative needs did not provide any great selective pressure to produce a system such as language.

EXAPTATION OR NATURAL SELECTION?
The question whether the evolution of language is a by-product of the evolution of other cognitive capacities (an emergent\(^1\) or exaptive\(^2\) property) has far reaching consequences for how
to design of the theory of the language faculty. In short, if the language faculty is subject to selective pressure, it is expected to represent an **OPTIMAL DESIGN SOLUTION**, in the same sense as the brain or the eye or the honey comb are perfect solutions to a given design problem. Conversely, if the language faculty arose exaptively, it should not have been shaped by selective pressures, and should (at least *a priory*) not be optimized.

**THE LANGUAGE FACULTY**

- Chomsky: Subject matter of linguistics is study of mind internal *language faculty*, which can be thought of as a *mental organ* of the body. Biology meets linguistics: → **BIOLINGUISTICS**

- Language faculty represents an **OPTIMAL DESIGN SOLUTION** whose growth and development is determined by three factors (from Chomsky 2004)
  - External data (linguistic input, driving language acquisition)
  - Genetic endowment (properties can be inferred from properties of Universal Grammar; gene FOX2P might be linked to evolution of language)
  - Principles of structural architecture and developmental constraints that are not specific to the organ under investigation, and may be organism-independent.
    (Compare to theormoregulatory systems, which evolved in order to keep constant the body temperature, even though their development came at a high caloric cost.)

The Minimalist Program (MP; Chomsky 1991-2004) defines the contours of a family of theories of the language faculty.

---

**2. THE MINIMALIST PROGRAM**

**2.1. Lexicon/Numeration**

1. a. Lexicon
   b. $L_{\text{English}} = \{\text{aardvark}, \text{believes}, \text{fever}, \text{John}, \text{saw}, \text{Mary's}, \text{zygoma}\}$

In MP, the derivation does not access the lexicon directly, but works on a proper subset of the lexicon only, called the **NUMERATION**. First, this reduces computational complexity. Second, there is empirical evidence for N from questions surrounding the proper definition of the concept of **COMPETITION** in syntax (see section 3 below).

2. a. The **NUMERATION** (N) contains a subset of the lexicon (and a subscript indicating the number of occurrences of each item; for details see section 2.3. below)
   b. $N = \{\text{John}, \text{Mary}, \text{see}\}$

---

(from Wikipedia entry on *Exaptation*)

---

3 Various (socio)biological system (e.g. avian breeding habits) are known to include aspects of optionality, indicating that sometimes, there is not a single optimal solution (Krebs and Davies 1997).
2.2. The Interfaces and Convergence

- The language faculty consist of two systems:
  - Cognitive system (narrow language faculty, or \( C_{HL} \) [C for 'computational system', HL for 'human language'])
  - Performance system

- Two language external systems that serve as interfaces to the language faculty
  - Sensorimotor system (articulation, auditory perception)
  - Conceptional-intentional system (reference, intention)

- Particular languages (German, Kiswahili,...) represent specific states of the language faculty.
- Derivations yield pairs relating the two interfaces PF and LF.
  - Phonetic Form (PF): interface to the articulatory-perceptual system, contains information on how to map specific morpheme combinations to phonemes, and how to phonetically realize these phonemes (e.g. div[ai]ne - div[I]nity,...)
  - Logical Form (LF): interface to the conceptual-intentional system, contains what is sometimes called logico-semantic information (e.g. why can 'A barber from every city hates it' mean the same as 'In every city, there is a barber that hates this city', but 'A barber from no Western city hates it' cannot be paraphrased as 'In no Western city, there is a barber that hates this city.')

A slightly more formal way to express this (rather a pseudo-formalization) looks as follows:

(3) A syntactic **DERIVATION** maps the numeration onto pairs \(<\pi, \lambda>\)

\(\pi\): Phonetic Form [PF] representation; \(\lambda\): Logical Form [LF] representation

- Ungrammatical derivations are ruled out by general interface conditions at PF and LF, as defined by the property of convergence:

(4) a. A derivation \(D\) based on a numeration \(N\) is successful iff \(D\) **CONVERGES** at the two interfaces LF and PF. (NB: This does not mean that the derivation will yield a well-formed output! Economy principles still have to be factored in at this point)

b. \(D\) converges at an interface \(I\) iff \(D\) is interpretable at \(I\) (informally: is well-formed w.r.t. the requirements imposed by \(I\)).

c. If a derivation \(D\) does not converge, \(D\) is said to **CRASH**.

In essence, the need for a \(D\) to converge entails that \(D\) must only contain information which can be processed by \(I\). For instance, \(D\) should no longer include instructions about phoneme structure once \(D\) reaches the LF-interface. For LF, it does not matter whether the subject in a sentence such as 'John saw Mary' is pronounces as [d3] or as [f]. Conversely, derivations that converge at PF must be void of semantic information. For PF, it is immaterial which of the two possible interpretations the string 'Everybody didn’t come to the party' is assigned.
The requirement to pass on only the minimal amount of information relevant to an interface can be motivated by considerations of methodological economy.

2.3. Structure building

The derivation exclusively uses the operations **MERGE** and **MOVE**. Recently, the number of operations has been reduced even further, as Chomsky (2004), re-interprets Move as a specific instance of Merge (*viz.* internal Merge; classic cases of Merge are referred to as *external* or *root* Merge for details see discussion of Bare Phrase Structure below).

\[
\text{Merge}(\alpha, \beta) := [\lambda, \alpha \beta] \quad (\text{where } \lambda \text{ is the label of the resulting tree})
\]

- Technically, the interaction between the structure building component (Merge) and the numeration is implemented by the algorithm in (6).

(6) 
   a. The numeration N contains indexed lexical items.
   b. For each lexical item, the index marks the number of times this lexical item is used.
   c. Each time the derivation accesses the numeration, it selects a single lexical item.
   d. Each time the derivation selects a lexical item, its index is reduced by 1.
   e. The derivation stops if the numeration has been exhausted and all indices equal 0.

Sample derivation:

(7) 
   a. Initial numeration \( N_0 = \{\text{the}_2, \text{saw}_1, \text{dog}_1, \text{cat}_1, \text{park}_1, \text{in}_1\} \)
   b. Select *the* from \( N_0 \)
      \( N_1 = \{\text{the}_2, \text{saw}_1, \text{dog}_1, \text{cat}_1, \text{park}_1, \text{in}_1\} \)
   c. Select *dog* from \( N_1 \)
      \( N_2 = \{\text{the}_2, \text{saw}_1, \text{dog}_0, \text{cat}_1, \text{park}_1, \text{in}_1\} \)
      Merge [the dog]
   d. Select *saw* from \( N_2 \)
      \( N_3 = \{\text{the}_2, \text{saw}_0, \text{dog}_0, \text{cat}_1, \text{park}_1, \text{in}_1\} \)
      Merge [saw [the dog]]
   e. Select *the* from \( N_3 \)
      \( N_4 = \{\text{the}_1, \text{saw}_0, \text{dog}_0, \text{cat}_1, \text{park}_1, \text{in}_1\} \)
   f. Select *park* from \( N_4 \)
      \( N_5 = \{\text{the}_1, \text{saw}_0, \text{dog}_0, \text{cat}_1, \text{park}_0, \text{in}_1\} \)
      Merge [the park]
   g. Select *in* from \( N_5 \)
      \( N_6 = \{\text{the}_1, \text{saw}_0, \text{dog}_0, \text{cat}_1, \text{park}_0, \text{in}_0\} \)
      Merge [in [the park]]
      Merge [[saw [the dog]] [in [the park]]]
   h. Select *the* from \( N_6 \)
      \( N_7 = \{\text{the}_0, \text{saw}_0, \text{dog}_0, \text{cat}_1, \text{park}_0, \text{in}_0\} \)
   i. Select *cat* from \( N_7 \)
      \( N_8 = \{\text{the}_0, \text{saw}_0, \text{dog}_0, \text{cat}_0, \text{park}_0, \text{in}_0\} \)
      \( \rightarrow \) \( N \) is exhausted
      Merge [the cat]
      Merge [[the cat][[saw [the dog]] [in [the park]]]]
2.4. THE ARCHITECTURE OF THE GRAMMAR

The derivation proceeds as follows: the lexical items are combined by the computational system \( C_{IL} \). Once all requirements of the derivation have been fulfilled and the derivation converges, the resulting a syntactic representation is transferred to the PF component by an operation called **SPELL-OUT**. Subsequent to Spell-out, the derivation proceed covertly: further syntactic operations may manipulate the tree at LF, but these changes are invisible and do not have an effect at PF.

\[
\text{ARCHITECTURE OF THE GRAMMAR}
\]

\[
\text{Lexicon} \rightarrow \text{Numeration} \rightarrow \text{Overt Syntax} \rightarrow \text{Spell-Out} \rightarrow \text{covert syntax/LF} \rightarrow \text{PF}
\]

(8)

3. THE ROLE OF ECONOMY

The Minimalist Program (MP; Chomsky 1991-2004) defines the contours of a family of theories of natural language syntax which embody the two central notions of *conceptual/methodological* and *substantive* economy. In contrast to earlier manifestations of (transformational versions of) generative grammar (such as the *Government and Binding* model; Chomsky 1981) theories that follow the guidelines of MP are characterized by two properties:

**CONCEPTUAL ECONOMY**

Driven by methodological, meta-theoretical considerations (*Occam’s Razor*): Analyses should employ a minimum of principles and grammatical constructs. A theory that uses fewer ingredients to explain a given set of phenomena is explanatory more adequate than a theory that uses additional assumptions.

- For instance, the Ptolemaic explanation of planetary motion needed four devices (deferents, epicycles, equants and eccentric, all embedded in a geocentric system), while Kepler’s theory reduced the phenomena to his Three Laws of Planetary Motion (and a heliocentric system, with additional empirical benefits)

  → Minimize number of levels or representation: DS and SS are eliminated as levels of representation (NB: also has *substantive* effects; see below)

  → Minimize number of axioms of theory...

**SUBSTANTIVE ECONOMY**

Syntactic derivations for natural language sentences are evaluated by **ECONOMY PRINCIPLES**. These principles compare sufficiently similar derivations along various dimensions (amount of operations, length of movement paths; computational cost;...) and eliminate all but the most economical one.

→ Principles of derivational and representational economy in (12) & (13)
3.1. INGREDIENTS OF AN ECONOMY ACCOUNT

In order to define the role of economy in restricting the space of possible derivations it is instrumental to understand four factors:

I. DEFINING THE DOMAIN OF COMPARISON. Which linguistic objects are compared by the apparatus that evaluates competing derivations (a.k.a. the ECONOMY METRIC)? Clearly, the economy principles need to apply to a multitude of objects, i.e. a set. (Questions such as *Is the Pope the tallest among all individuals that presently qualify as Popes?*, where comparison accesses a singleton set, do simply not make much sense.) Call this collection the REFERENCE SET (RS). In MP), the members of the reference set are syntactic derivations of natural language sentences. A hypothetical RS could look as follows:

\[
(9) \quad RS = \{D_1, D_2, D_3\}
\]

where

- $D_1$ is the derivation of *There is someone,*
- $D_2$ is the derivation of *John saw Mary,*
- $D_3$ is the derivation of *Seventeen frogs crossed the border the night Salomon died.*

The exact restrictions on the RS is subject of intense current debate, and no single solution definition has emerged yet. Still, it is possible to outline some initial requirements, along with a couple of remarks on the interaction between the RS and the numeration, which will be taken up in section 3.2.

II. SELECTING THE DIRECTION OF OPTIMIZATION. Given the architecture of MP, in which direction in the grammar model does optimization apply? Which areas or domains are affected by economy? Does the grammar strive to provide the most economical LF or PF representations, or does it optimize the computations that lead up to these representations? Clearly, if any, the latter must be true. Otherwise, all utterances would end up as something like *baba* at PF, which has been claimed to represent the optimal expression from a phonetic perspective (it maximized contrasts, thus minimizing perception cost). In MP, this is expressed by assuming that the LF and PF interfaces are not in the core syntactic component itself, making them impervious to optimization; rather, LF and PF impose external restrictions on the output of the grammar (so called bare output conditions).

A simile: Similar considerations arise in other computational procedures that interface with external systems. Take e.g. optimization of the process that leads to production of newspaper, whose abstract computational representation can be thought of as $C_{III}$. Readability, i.e. the need of the final print to be perceptible by human eyes serves as an external constraints imposed by the (visual) interface. Optimization can either minimize the efforts/costs of the printing process, e.g. by minimizing font size to save ink. On the other side, if the economy metric aims at optimizing compatibility with the interface, it should maximize font size for reasons of (verbatim) readability.

III. DEFINING SCALES (DIMENSIONS). What counts as economical depends on the scale that defines the ranking of the competing derivation w.r.t. each other. But which types of optimization algorithms does one expect to find in the computational system? In principle, the economy metric could impose quite unnatural conditions e.g. by counting and comparing the
number of words in a sentence, or comparing the arity\(^4\) of the predicates. Both of these hypothetical ways of implementing economy yield unreasonable results. Applied to (10) and (11), the metric would (of course wrongly) predict the b-examples to block the a-examples.

(10)  
  a. Sally arrived yesterday  
  b. It rains  

(11)  
  a. The big fish died  
  b. Sally ate the fish

Generally, the existence of such principles would be rather surprising, given the way other biological systems operate (counting does not seem to be a natural evolutionary mechanism).

The specific cases of competition in (10) and (11) can be avoided by including the requirement that the RS only contains candidates with identical numeration (or meanings). But even among the more plausible scales, it is necessary to make a choice as to whether they should be seen as part of the grammar. The list in (12) provides a number of scales that have been suggested in the grammar, as well as principles that have been suggested to operate on these scales (italics):

(12) **PRINCIPLES OF DERIVATIONAL ECONOMY**

  a. Preference for shorter movements (as opposed to longer ones)  
     → **formalized in principles called** *Shortest (Move)* and *Minimal Link Condition*  
  b. Preference for derivations that employ fewer (as opposed to more) operations  
     → **Fewest Steps**  
  c. Delay operations as long as possible  
     → **Procrastinate**  
  d. Move only if it cannot be avoided  
     → **Last Resort**  
  e. Move only if required by the category that undergoes movement  
     → **Greed**  
  f. Move only if required by the item that triggers movement  
     → *Altruism, Enlightened Self-Interest*  
  g. Move only if movement creates a new interpretation  
     → **Scope Economy**  
  h. Competition among insertion and movement  
     → **Merge over Move**  
  i. Prefer deletion in higher movement copies  
     → **Minimize Restriction**

As a result, a shorter derivation will be more economical than a longer one, one which employs fewer movement steps will win over a competing one with more operations, etc...

In addition, there is also a principle that applies to representations, the principle of *Full*
Interpretation (FI), and the requirement to eliminate unnecessary levels of representation (Although the latter has a strong methodological flavor, it might also be made an explicit part of the theory, and is included in the discussion for this reason.)

(13) REPRESENTATIONAL ECONOMY

a. Minimize the number of elements in a representation. The syntactic representation cannot contain contentless, spurious elements. The syntactic representation cannot contain elements that are not required for convergence.

→ Principle of Full Interpretation

b. Minimize the number of levels of representations (Deep Structure, Surface Structure,...). The theory of the language faculty cannot contain spurious levels of representation.

→ Elimination of Deep Structure and Surface Structure

IV. INTERPRETATION OF THE RESULTS OF THE ECONOMY METRIC. What are the consequences for a set of sentences to be evaluated by the economy metric? Which point of the scale triggers which effect? Are the less economical results e.g. supposed to be less often used or less acceptable? Or is the third-ranked candidate the only one that can be used in an embedded context? While such absurd outcomes are in principle imaginable (e.g. in a game designed by humans), it is highly unlikely that they can be used to describe natural language. Still, choice among the more plausible options is crucial for the design of the theory. MP adopts the following specific assumption:

(14) For any reference set RS, the most economical derivation RS generates a string that is judged to be well-formed by native speakers. All other members of RS are ill-formed.

→ UNGRAMMATICALITY can be the result of two sources:

○ the derivation loses out to a more economical competitor

○ the derivation fails to converge

Related to these considerations is the concept of INEFFABILITY: In certain cases, none of the competing derivations yields a well-formed result, because none of them converges, even though the string could in principle be assigned a consistent meaning. As an example think of the numerations \( N_1 = \{\text{John}_1, \text{dined}_1, \text{the}_1, \text{meat}_1\} \), which - in contrast to \( N_1 = \{\text{John}_1, \text{ate}_1, \text{the}_1, \text{meat}_1\} \) - does not support a well-formed representation:

(15) a. John ate the meat

b. John ate the meat

(16) a. John dined

b. *John dined the meat
3.2. The Numeration and the Reference Set

(partially from Hornstein, Nunes & Grohmann 2005: 330ff)

A. On the one side, reference sets have to be defined relative to a given numeration. The derivation should access a numeration, instead of the full lexicon. This decision is motivated by two considerations:

A1. Economy principles select among the cheapest derivation relative to a given reference set. Even though they have not been specified in detail yet, it is clear that the individual constraints in the list in (12) generally have the effect of selecting those derivations that minimize the number of operations. Assume now that derivations have free access to the lexicon, drawing any number of lexical items into the computational system in order to form a syntactic derivations. On this view, derivations that use the smallest number of items should be preferred, since selection from the lexicon counts as an operation. It follows now also that all derivations should involve the minimal number of lexical items (zero), resulting in an empty string (silence). Assuming that derivations compete only if they use the same given set of lexical items, which is provided by the numeration, removes this problem.

A2. Second, in a system where the derivations to be compared have access to the full lexicon, derivations that employ fewer operations in the syntactic component should be preferred over ones that use more. (17)a should e.g. block (17)b, because the latter involves more instances of Merge. Again, this does not seem right.

(17) a. John saw Mary
   b. John saw Mary in the park

Restricting competition to derivations that use the same numeration solves this problem. (17)a and (17)b start out with different numerations, and (17)a can therefore not block (17)b. Similarly, this accounts for the fact that the derivation for (18)a does not compete with (18)b, although the latter involves an additional movement step.

(18) a. Mary saw John
   b. What did Mary see

B. Competing derivations do not only have to come from the same numeration, but also need to satisfy another requirement: they have to converge. If all derivations in a given RS were to be compared, economy would sanction the cheapest operation, irrespective whether it converges or not. But this would leave unaccounted for the ill-formedness of non-converging derivations. On this view, a more economical non-converging derivations should always block converging competitors that use additional operations. Applied to an example, (19)b should block (19)a, even though only the former converges ((19)b violates the EPP, among others):

(19) a. There, seems t, to be a mistake
   b. *seems there to be a mistake

Thus, only convergent derivations are included in the RS (for a diverging view see below).
4. **Bare Phrase Structure**

One of the most important properties of NL is its *recursive part whole structure*, subject of the Phrase Structure component, which allows generation of strings such as (20):

(20) They met the sister of the brother of the youngest daughter of the first born son of the ...

The MP simplifies earlier versions of PS-theory by eliminating redundancies, resulting in a more parsimonious, and therefore conceptually more economical system.

4.1. **Generalized Transformations**

Logical Structure of Linguistic Theory (LSLT; Chomsky 1955). Merge any two syntactic objects by **Generalized Transformations** (GT) At the starting point of derivation, there is no single constituent structure tree, but syntactic structures are built by joining little sub-trees.

4.2. **Phrase Structure Rules**

Aspects (Chomsky 1965). Generate trees (phrase markers) by PS-rules.

(21) \[ NP \rightarrow NP \text{ PP} \]

At the beginning of the derivation, at DS, the syntactic tree is fully projected by PS rules. Lexical items are added as terminal symbols by lexical insertion.

**Problem for PS-rules:** the part of phrase structure that fixes the number of internal arguments is coded twice (\( \rightarrow \) violates the spirit of conceptual economy)

- in the subcategorization frame of the lexicon - e.g. transitive vs. intransitive verbs and
- in PS-rules

(22) a. \[ VP \rightarrow V \]

b. \[ VP \rightarrow V \text{ NP} \]

The information provided by PS-rules is redundant (for internal arguments!)

4.3. **X'-theory**

Basic properties of X'-theory (Chomsky 1973):

- XP is a projection of its head
- Cross-categorial harmony: VP, NP, AP, CP,... are internally structured in the same way
- Binary branching
- Binary identification of number of levels: one maximal, one minimal, one non-maximal/non-minimal (= X').
- Recursiveness results from combining categories in such a way that the output conforms with X'-theory (e.g. by inserting a CP in the complement position of V). The combinatorial system thus also employs a rule that is an instantiation of Merge.

4.4. **Bare Phrase Structure**
MP uses **Bare Phrase Structure** (BPS; Chomsky 1995). In BPS, there is only a single structure building operation which essentially qualifies as a Generalized Transformation: **Merge**.

\[(23) \text{Merge}(\alpha, \beta) := [\lambda, \alpha \beta] \quad \text{(where } \lambda \text{ is the label of the resulting tree)}\]

Note that Merge does not impose any restrictions on the output order of the two terms that are merged. It does not matter whether *read* is merged with *the book* resulting in \([\text{read}, \text{read the book}]\) (where \(\alpha\) is instantiated by the verb and \(\beta\) by *the book*) or \([\text{read the book}, \text{read}]\) (where \(\alpha\) is instantiated by *the book* and \(\beta\) by *read*).

- There are two types of configurations in which Merge can apply, yielding two subtypes of the operation. **External Merge** leads to extension of the structure (classic Merger), while **Internal Merge** results in movement configurations (Chomsky 2004).

- Formally, Merge simply forms pairs - it does so by forming the set of the two objects \(\alpha\) and \(\beta\), and specifying which of the two is the head. One way to do this is to define the head as the *first* element in a pair:

\[(24) \text{Merge}(\alpha, \beta) = <\alpha, \beta> \quad \text{\(\alpha\) provides the label and ‘projects’}\]

A notational variant, the so-called *Wiener-Kuratowski* notation of ordered pairs is the more commonly used in the literature:

\[(25) \text{Merge}(\alpha, \beta) = \{\alpha, \{\alpha, \beta\}\} \quad \text{\(\alpha\) provides the label}\]

A simple proof for the equivalence of (24) and (25): Formally, \(\{\alpha, \{\alpha, \beta\}\}\) is another way of writing \(<\alpha, \beta>\). Assuming that (i) \(\gamma\) in \(\{\gamma, \{\alpha, \beta\}\}\) denotes the label of the phrase, and that (ii) \(\gamma\) either equals \(\alpha\) or \(\beta\) (Endocentricity) yields the two options \(\{\alpha, \{\alpha, \beta\}\}\) and \(\{\beta, \{\alpha, \beta\}\}\). These two possible configurations in turn collapse into \(<\alpha, \beta>\) and \(<\beta, \alpha>\). Thus, if Merge applies to \(\alpha\) and \(\beta\) (in that order), it will result in a tree containing \(\alpha\) and \(\beta\) (in any order) with label \(\alpha\); if Merge applies to \(\beta\) and \(\alpha\) (in that order), it will result in a tree containing \(\alpha\) and \(\beta\) (in any order) that projects \(\beta\). Note that which option is eventually chosen by the grammar must be determined by independent factors.

BPS differs from earlier versions of the PS component in two ways: first, it adopts different assumptions as to the well-formedness conditions on certain syntactic configurations. Second, it eliminates a number of redundancies from earlier models.

### 4.4.1. Changes in Tree Design

Before tracing the most important developments, it is important to observe some changes in the basic design of tree representations.

- There is now a designated node for the base position of the subject. This node, vP (“little VP”) contains the subject in its specifier position and takes VP as its complement. Thus, subjects no longer move from SpecVP to SpecTP, but from SpecvP to SpecTP.
There are a variety of alternative proposals (Agree, Feature Movement,...) for Case assignment. For present purposes, nothing bears on the specific choice.

Objects are licensed for Case by (covertly) moving into the outer specifier of \(v^o\). Thus, little \(v^o\) serves the dual function of introducing the subject (e.g. by assigning a \(\Theta\)-role to the subject), and of providing Case information for the direct object:

**4.4.2. Elimination of Redundancies**

BPS eliminates various superfluous notions from X'-Theory, and is therefore descriptively more adequate (more economical) than earlier models of PS. Four of these advantages are listed below (the discussion partially follows Lasnik, Nunes & Grohmann 2005: 196ff).

I. Diacritics that denote bar level (\(\circ\) in \(X^o\), ‘‘’ in \(X\), ‘P’ for XP) are not part of the lexicon → violation of Inclusiveness.

(28) **Inclusiveness**

The computation does not introduce new items apart from the lexical items and their features. (Corollary: all features are part of the lexicon.)

In BPS, the levels are identified by functional determination: just like the notion ‘object’ and

---

5There are a variety of alternative proposals (Agree, Feature Movement,...) for Case assignment. For present purposes, nothing bears on the specific choice.
‘subject’ are relational notions, that are specified by the syntactic context (SpecTP vs. sister to head), the phrase structural status of a category (maximal, minimal, intermediate) is determined by its respective position w.r.t. other categories. Thus, BPS conforms better with Inclusiveness.

\[(29)\]
\[
a. \ \alpha \text{ is a maximal projection iff } \alpha \text{ does not project}
\]
\[
b. \ \alpha \text{ is a minimal projection iff } \alpha \text{ is selected from the numeration}
\]
\[
c. \ \alpha \text{ is an intermediate projection iff } \alpha \text{ is neither a maximal nor a minimal projection.}
\]

II. BPS derives Endocentricity, which needs to be stipulated in X’-theory by stating that \(X^o\) projects to \(X\), and not \(Y\) (where \(\neg X=Y\)). It does so by the format of Merge in (24).

III. BPS eliminates spurious, superfluous projections by getting rid of branching nodes that only dominate a single contentful node. Instead of being parsed as in (30), a name such as *Mary* is now reduced to (31)a (to be further simplified), while a VP is matched on the tree (31)b:

\[(30)\]
\[
\begin{array}{c}
\text{DP} \\
\text{D'} \\
\text{D}^o \\
\text{NP} \\
\text{N'} \\
\text{N}^o \\
\text{Mary}
\end{array}
\]

\[(31)\]
\[
\begin{array}{c}
\text{a. } \text{N} \\
\text{b. see the book}
\end{array}
\]

\[
\begin{array}{c}
\text{V} \\
\text{D} \\
\text{see} \\
\text{D} \\
\text{N} \\
\text{the} \\
\text{book}
\end{array}
\]

IV. Finally, BPS eliminates categories all together, and thereby reduces a redundancy between lexicon and PS (see discussion of X’-theory, where the change from PS-rules to X’-theory was motivated by a similar search for reducing redundancies.) More specifically, the fact that *Mary* is a noun is already stored in the lexicon. Thus, introducing a label N in the derivation does not add new information. Thus, (31)a is further simplified to (32)a, which can - in fact, has to - be collapsed into (32)a’ while (31)b is now transposed into (32)b (intermediate representation omitted).

\[(32)\]
\[
\begin{array}{c}
\text{a. } \text{Mary} \\
\text{a’} . \text{Mary}
\end{array}
\]

\[
\begin{array}{c}
\text{b. see} \\
\text{see} \\
\text{the} \\
\text{the book}
\end{array}
\]

As an additional bonus, rewriting (32)a as (32)a’ eliminates the distinction between *lexical item* and *terminal nodes* - they can now no longer be distinguished.
**QUESTION:** Why does (32)a collapse into (32)a’? The answer runs as follows. Merge applies to two terms. The first member of the pair serves a dual function: it denotes a term that merges, and the term that projects (label). The second member of the pair is the non-projecting second term that the first member merges with. Now in trees such as (32)a, *Mary* is the label, so should be the first member of the set. But since it does not merge with anything, the resulting pair is not defined. Merge by definition only applies to two terms.

NB: *Mary* does not merge with the empty set. This would yield <*Mary*, {}>, or {*Mary*, {*Mary*, {}}}, a branching tree that combines *Mary* and {}, formally an object different from (32)a.

Thus, a system that just admits Merge cannot even represent (32)a. No such problems arise with (32)a’, which contains the same syntactic information, but represents a tree theoretic object of its own (Chomsky calls them 'syntactic objects'), and therefore can combine with other syntactic objects.

NB: Formally, trees are pairs <A, D>, where A is a set of nodes and D a set of pairs capturing the dominance relation. Trees can be trivial in that D is empty: <a, {}>. Thus, *Mary* is a tree/phrase marker.

**REFERENCES**


