

## **Notas sobre el formalismo de la GPSG.**

*J.A.Alonso, J.C.Ruiz*

## NOTAS SOBRE EL FORMALISMO DE LA GPSG

J.A. Alonso (SIEMENS)  
J.C. Ruiz Anton (EUROTRA)

### 1.1. - Fundamentos de la GPSG

La Gramática Sintagmática Generalizada (*Generalized Phrase Structure Grammar* o GPSG) es una gramática generativa en el sentido clásico, i.e. una teoría formal que asigna descripciones estructurales a las oraciones. Una Gramática generativa consta de reglas que determinan explícitamente todas y (sólo) las oraciones bien formadas de una lengua.

La GPSG (Gazdar, Klein, Pullum y Sag 1985, a partir de aquí abreviado en GKPS) insiste en la preocupación por el rigor formal y la precisión matemática que a juicio de los autores ha de tener la teoría gramatical. Uno de los objetivos de este formalismo es alcanzar un metalenguaje restringido capaz de definir las gramáticas de las lenguas naturales.

Básicamente, la GPSG asume lo siguiente:

- Una gramática con reglas *sintagmáticas* es capaz de generar los lenguajes naturales. Al asumir esta hipótesis, la GPSG rechaza la conocida opinión contraria de Chomsky (1956), en la cual se fundamentó la introducción de transformaciones en la teoría generativa.
- En consecuencia, la GPSG no hace uso de transformaciones, entendidas como reglas que transforman (proyectan) representaciones en otras transformaciones.
- La GPSG parte de un único nivel de representación oracional. Se trata de una estructura de constituyentes, muy similar a la estructura superficial de Chomsky (1965). La GPSG es, por tanto, una teoría *monoestratal*, frente a otros modelos como la Gramática Léxico Funcional (LFG)<sup>1</sup> o la Teoría de la rección y del ligamen (GB)<sup>2</sup>, que admiten varios niveles de representación.

A diferencia de las versiones clásicas de la Gramática Generativa, la GPSG no define directamente sus reglas sintagmáticas. En su lugar, recurre a técnicas cuyo efecto es hacer la gramática más compacta de lo que sería una simple lista de reglas, y a la vez permiten extraer importantes generalizaciones lingüísticas. Estas técnicas son:

<sup>1</sup> cf. Kaplan / Bresnan (1982).

<sup>2</sup> *Government and Binding Theory* (GB) cf. Chomsky 1981, 1985.

- una teoría muy elaborada de los rasgos sintácticos.
- factorización de las reglas sintagmáticas en reglas de dominio, reglas de precedencia y metarreglas.
- principios de instanciación de rasgos.

## 1.2.- El concepto de categoría en la GPSG; Teoría de Rasgos

En GPSG, una categoría se define como un símbolo complejo compuesto de un conjunto de pares atributo-valor (rasgos). Es importante tener en mente que esto no significa que una categoría contenga un conjunto de rasgos, sino que es éste el que conforma una categoría dada. Las etiquetas utilizadas en las reglas para referirse a diferentes categorías (NP, VP, PP, etc.) son únicamente siglas mnemotécnicas que se refieren a determinados conjuntos de rasgos. Así por ejemplo:

NP se refiere al conjunto [+N,-V,BAR 2,...]

VP se refiere al conjunto [-N,+V,BAR 2,...]

etc.

Cada rasgo se compone de un atributo, que representa una cierta propiedad gramatical relevante para la teoría (género, número, persona, tiempo, nivel de barra, etc.), y un valor asociado (singular, primera, presente, 2, etc.).

Hay diferentes tipos de rasgos, según éstos tomen valores atómicos o categoriales. Algunos ejemplos de rasgos con valores atómicos son:

PER [1,2,3]

PAST [+,-]

BAR [0,1,2]

etc.

Los rasgos con valores categoriales, como por ejemplo SLASH, AGR, RE o WH, toman como valor un conjunto de rasgos que especifican (en menor o mayor medida) una categoría. Así, por ejemplo:

AGR[+N,-V,PER 3,-PLU,CA NOM,...]

se refiere al rasgo AGR (concordancia) con un valor que representa un NP singular acusativo.

## 1.3.- Concepto de unificación.

La GPSG se apoya extensamente en el concepto de unificación. Se denomina unificación a la unión de dos categorías (conjuntos de rasgos) en el caso de que los valores de sus respectivos rasgos no sean incompatibles. Por ejemplo; sean las categorías (1), (2) y (3):

(1) [+N,-V,-PLU,CA NOM]

(2) [+N,BAR 2, PER 3]

(3) [+N,-V,+PLU]

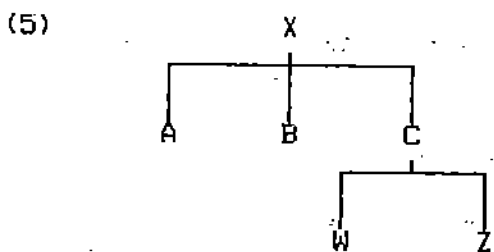
(1) unificaría con (2) en la categoría extendida (4), mientras que en (1) y (3), al tener un valor incompatible para PLU fallaría la unificación.

(4) [+N,-V,-PLU,CA NOM,BAR 2,PER 3]

#### 1.4.- Rasgos instanciados y rasgos heredados.

Un sub-árbol formado por un nodo raíz y todos y sólo los nodos dominados inmediatamente por éste recibe el nombre de árbol local.

Así, en (5), A,B,C conforman un árbol local dominado por X. Lo mismo sucede con W,Z respecto a C.



Un rasgo que forme parte de una categoría en un árbol local se dice que es heredado si la regla ID que induce ese sub-árbol introduce explícitamente dicho rasgo.

Por el contrario, un rasgo se dice que es instanciado cuando su presencia en una categoría de un árbol local se debe a la aplicación de mecanismos como FCRs, FSDs, o los principios de instanciación de rasgos FFP, CAP o HFC, no siendo introducido directamente por ninguna regla ID (cf. punto 2). Así, en el árbol

(6): VP[SLASH[NP]] (= VP/NF)  
V[40]  
VP[FIN,-PLU]

inducido por la regla ID

(7): VP/NF --> H[40], VP[FIN]

los rasgos SLASH y FIN son heredados, mientras que [-PLU] es instanciado.

#### 2.- Mecanismos del formalismo.

Los mecanismos de construcción y validación de árboles con los que trabaja la GPSG son los siguientes:

- \* Reglas de dominancia (ID)
- \* Reglas de precedencia (LP)

- \* Metarreglas
- \* Restricciones de coocurrencia de rasgos (FCR)
- \* Principios universales de instanciación de rasgos (FFP, CAP y HFC)
- \* Especificaciones de rasgos por defecto (FSD)

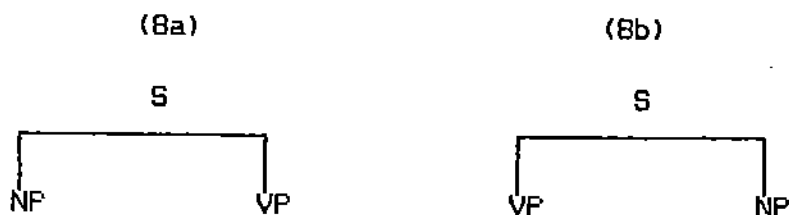
Seguidamente, daremos un breve resumen de cada uno de estos mecanismos:

### 2.1.- Reglas de dominancia (reglas ID).

Las reglas ID (*Immediate Dominance*) indican relaciones de dominancia entre una categoría raíz y una serie de categorías a las que ésta domina, sin dar ninguna información sobre el orden de precedencia de las categorías hijas. Por ejemplo, la siguiente regla ID:

(8): S  $\rightarrow$  NP, VP

generaría dos posibles árboles locales:

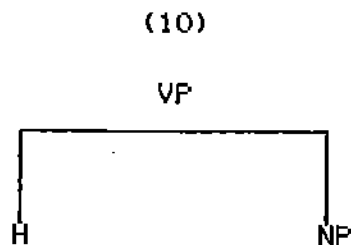


Como convención, la GPSG supone que toda regla ID introduce un núcleo (HEAD) en los árboles locales que induce. Se define como núcleo de un árbol local una categoría H cuyos rasgos N y V coincidan en valor con los de la categoría raíz, y cuyo valor para el rasgo BAR sea igual o menor que el de ésta.<sup>3</sup>

Así, la regla ID

(9): VP  $\rightarrow$  H, NP

introduce árboles del tipo



<sup>3</sup> Esta definición está simplificada respecto a la dada en GKPS (pág. 51). Más estrictamente, H es una abreviatura que representa un conjunto de rasgos que se instanciarán por la aplicación ulterior de la convención HFC (cf. 2.5.3).

donde H es una categoría [+V,-N,BAR 0,...], es decir, un verbo léxico.

## 2.2.- Reglas de precedencia lineal (reglas LP)

Las reglas LP establecen relaciones de precedencia lineal entre constituyentes dentro de árboles locales.

Así, una regla LP como:

(11): [+N] < V[BAR 2] < P[BAR 2]

indica que cualquier constituyente nominal debe preceder a cualquier VP, y ambos deben preceder a cualquier PP dentro de un árbol local. Una regla LP como ésta restringiría los árboles posibles generados por (8) a (8a), ya que (8b) viola (11).

## 2.3.- Metarreglas.

Las metarreglas son un mecanismo para ampliar el conjunto de reglas ID del que dispone la gramática para evitar tener que especificar todas ellas explícitamente.

Una metarregla se aplica sobre una regla ID léxica para dar como resultado otra regla ID léxica, sin anular la primera. Las metarreglas se diferencian de las transformaciones estructurales típicas de las gramáticas generativas en cuanto que:

- \* No se aplican sobre árboles, sino sobre reglas.
- \* No convierten una estructura en otra, sino que a partir de unas reglas ID ya existentes, amplían el conjunto de éstas creando otras nuevas.

Un ejemplo de metarregla es la llamada Metarregla de pasivización, que para el inglés se define como:

(12):           VP --> W, NP  
                  |  
                  V  
          VP(PAS) --> W, (PP[by])

Esta metarregla especifica que para cada regla ID que permita a un VP dominar un NP y algo más (W), debe haber otra regla ID que introduzca una categoría VP pasiva que domine las categorías englobadas en W, junto con (opcionalmente) un PP con un valor by para el rasgo FF0RM.

## 2.4.- Restricciones de Coaparición de Rasgos (FCR)

Las FCRs (*Feature Co-occurrence Restrictions*) establecen restricciones sobre especificaciones de rasgos.

Por ejemplo, una FCR como

(13): FCR 1: [+INV] > [+AUX,FIN]

establece que si se da una especificación de rasgos [+INV] en una categoría, forzosamente se tienen que dar las especificaciones [+AUX,FIN] para esa categoría. En otras palabras, siempre que se dé un caso de cláusula con el verbo en primera posición (inversión), éste deberá ser finito.

## 2.5.- Principios universales de instanciación de rasgos.

La GPSG define tres principios que determinan la buena formación de árboles locales en cuanto a presencia, ausencia o coaparición de rasgos en ciertos nodos clave de dichos árboles. Estos tres principios son los llamados Foot Feature Principle (FFP, o principio de rasgos "foot"), el Control Agreement Principle (CAP, o principio de concordancia de control) y el Head Feature Convention (HFC, o convención de rasgos nucleares).

Seguidamente daremos un breve resumen de cada uno de ellos, utilizando más ideas intuitivas sobre su función y utilidad que el formalismo matemático que se utiliza en el libro para definirlos.

### 2.5.1.- FFP (Foot Feature Principle).

El FFP establece la obligación de que en todo árbol local válido la categoría raíz y todas las categorías hijas unifiquen entre sí sus respectivas especificaciones para los rasgos que pertenecen al subconjunto FOOT, esto es [SLASH,WH,RE].

Una importante restricción del principio es que es válido sólo para rasgos del subconjunto FOOT que estén instanciados, no aplicándose para ningún rasgo heredado (cf. 1.4).

El FFP se ocupa de asegurar la presencia en los nodos necesarios de los rasgos que determinan el control de fenómenos anafóricos, de topicalización o de movimiento (en frases WH, por ejemplo).

### 2.5.2.- CAP (Control Agreement Principle).

El CAP es un principio de buena-formación de árboles que asegura la concordancia entre constituyentes dentro de una frase.

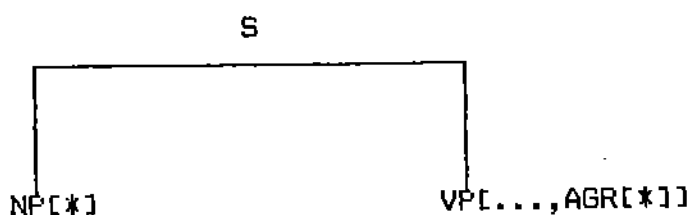
Para ello, se define el concepto de categorías controladoras y controladas, en términos del tipo semántico que cada una de ellas tenga asignado. Sin entrar en detalles, baste mencionar que dentro de la teoría, sólo las categorías nominales pueden ser controladoras, mientras que categorías controladas son todas aquellas que en la forma semántica sean funtores cuyos argumentos sean categorías controladoras, es decir, verbos, adjetivos o determinantes.

Dentro de un árbol local, una categoría hija dada puede tener un controlador local o no tenerlo. En el primer caso, el CAP obliga a que el valor del rasgo de control (AGR o SLASH) de la categoría controlada se unifique con la categoría controladora local. En el segundo caso, el CAP obliga a que las

especificaciones para el rasgo de control de la categoría controlada coincide con las de la categoría raíz.

Así, en el árbol (14a)

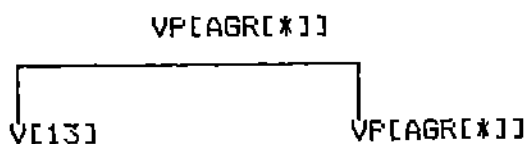
(14a):



el nodo NP controla a VP, y el árbol sólo cumplirá el CAP sólo si las especificaciones de NP, (\*), coinciden con las del rasgo AGR de VP, (\*).

Por otra parte, el árbol (14b)

(14b):



al no tener controlador local, sólo cumplirá el CAP si el valor para el rasgo AGR en el nodo VP raíz y el nodo VP hijo son el mismo.

### 2.5.3.- HFC (Head Feature Convention).

El objetivo del HFC es asegurar la unificación de los rasgos pertenecientes al subconjunto HEAD entre las categorías raíz y núcleo dentro de un árbol local.

Básicamente, el HFC establece que, para todo árbol local que cumpla el FFP y el CAP, todos los rasgos HEAD presentes en la raíz que sean libres en el núcleo del árbol (cf. 2.1) (es decir, que puedan pasar a formar parte de la categoría núcleo sin que haya conflictos de oposición de valores con rasgos ya existentes) deben aparecer en el núcleo, y viceversa: todas las especificaciones de rasgos HEAD que aparezcan en el núcleo del árbol y sean libres en la raíz, deben pasar a formar parte de ésta.

### 2.5.6.- Especificaciones de rasgos por defecto (FSD)

Las FSDs son mecanismos que dotan a las categorías de unas determinadas especificaciones de rasgos en caso que ningún otro mecanismo especifique lo contrario (ya sea otro valor que el marcado por la FSD, o bien la no aparición o aparición del rasgo en cuestión).

Por ejemplo, la siguiente FSD

(15): FSD:     -[CONJ]



establece que si es posible que una categoría carezca del rasgo CONJ (es decir, si ningún otro mecanismo de la teoría obliga a ello), el rasgo CONJ debe estar en efecto ausente de la categoría en cuestión.

### 3. Implementación de GPSG

Veremos aquí las formas de tratar computacionalmente las reglas y principios de la GPSG, y algunos de los problemas que plantean.

#### 3.1.- Reglas ID/LP

Hay dos posibilidades de tratar computacionalmente el reparto de trabajo entre reglas ID y reglas LP:

\* compilar una gramática objeto extendida, que exprese el efecto de combinar ambos tipos de generalización en el formato más familiar de las reglas libres de contexto (CF), que disponen de algoritmos de *parsing* probados y eficaces (Earley, CYK, etc).

Este punto de vista es el adoptado por Phillips y Thompson (1987) del Departamento de IA de la Universidad de Edimburgo. Su proceso de expansión es el siguiente:

- las reglas ID se aumentan por metarreglas
- las reglas ID resultantes se combinan (teniendo en cuenta las reglas LP) en sus posibles órdenes.
- el nuevo conjunto de reglas se completa por medio de las convenciones de instanciación de rasgos.

El problema de esta estrategia es la explosión combinatoria que se produce al compilar gramáticas grandes. Al tener que calcular todas las permutaciones de las reglas ID (filtrando algunas de ellas por las reglas LP), la expansión puede producir listas de miles de reglas CF.

\* El enfoque opuesto consiste en diseñar un *parser* que procese directa y simultáneamente las reglas ID y las reglas LP. Shieber (1983) ha adaptado con algunos retoques el algoritmo de Earley a esta nueva función. En este análisis, el algoritmo no genera todas las combinaciones de una regla ID, sino que examina las posibilidades un paso cada vez, a medida que las va necesitando.

La implementación de Evans y Gazdar (1984) sigue esta línea. Su técnica es compilar primero las reglas ID básicas, las metarreglas, y una versión simplificada de los principios HFC y CAP en un conjunto sencillo de reglas ID, dejando al *parser* la comprobación de las reglas LP y de las restantes convenciones de rasgos.