



**Universidad
Europea de Madrid**

LAUREATE INTERNATIONAL UNIVERSITIES

ANÁLISIS SEMÁNTICO

ETDS VS DDS

© Todos los derechos de propiedad intelectual de esta obra pertenecen en exclusiva a la Universidad Europea de Madrid, S.L.U. Queda terminantemente prohibida la reproducción, puesta a disposición del público y en general cualquier otra forma de explotación de toda o parte de la misma.

La utilización no autorizada de esta obra, así como los perjuicios ocasionados en los derechos de propiedad intelectual e industrial de la Universidad Europea de Madrid, S.L.U., darán lugar al ejercicio de las acciones que legalmente le correspondan y, en su caso, a las responsabilidades que de dicho ejercicio se deriven.

Índice

Presentación	4
Descripción de la necesidad	5
Definiciones dirigidas por la sintaxis	6
DDS. Evaluación de la gramática I	8
DDS. Evaluación de la gramática II	9
Esquemas de traducción dirigidos por la sintaxis	11
ETDS. Atributos heredados por la izquierda y evaluación descendente	13
Eliminación de la recursividad	13
ETDS. Atributos heredados por la izquierda y evaluación ascendente	15
Incorporación de símbolos marcadores	15
Creación del árbol sintáctico	17
Resumen	18

Presentación

El objetivo de este tema es comprender las tareas necesarias para realizar los controles semánticos y sus acciones correspondientes, así como las dos notaciones utilizadas para llevarlos a cabo.

En este tema aprenderemos a distinguir entre las **definiciones dirigidas por la sintaxis (DDS)** y **los esquemas de traducción dirigidos por la sintaxis (ETDS)**, y para ello se alcanzarán los siguientes objetivos:

- Entender en qué consiste la traducción dirigida por la sintaxis y las dos notaciones existentes.
- Conocer las definiciones dirigidas por la sintaxis y su funcionamiento con atributos sintetizados y heredados.
- Comprender cómo se evalúa una DDS con atributos sintetizados y heredados.
- Estudiar las características de los ETDS tanto para atributos sintetizados como heredados.
- Entender en los ETDS, los atributos heredados por la izquierda con una evaluación descendente.
- Entender en los ETDS, los atributos heredados por la izquierda con una evaluación ascendente.



Descripción de la necesidad

Como hemos visto, las gramáticas de atributo son gramáticas independientes del contexto, o gramáticas de tipo 2, a las que se añaden atributos y reglas de evaluación de los mismos. Cada atributo es una variable que representa una propiedad del símbolo, bien sea terminal o no terminal, pudiendo ser este sintetizado (se calcula a partir de los atributos de los hijos de dicho nodo) o heredado (se calcula a partir de los atributos del padre o de los hermanos).

Con el objeto de utilizar los atributos para realizar tareas, utilizamos reglas semánticas asociadas a las producciones de la gramática. Estas reglas semánticas se definen en función de lo que queramos conseguir con los atributos de cada producción.

Esta asociación entre reglas semánticas y la estructura sintáctica a través del árbol de análisis sintáctico se denomina **traducción dirigida por la sintaxis**. Para realizar la traducción dirigida por la sintaxis se utilizan dos notaciones:

Definiciones dirigidas por la sintaxis (DDS).	Son especificaciones de alto nivel para las traducciones, que ocultan muchos detalles de la implantación, y donde el orden en el que se especifican no tiene por qué ser el orden de ejecución de las reglas. Las acciones semánticas van siempre al final de la producción. En algunos textos aparecen separadas las acciones por " ".
Esquemas de traducción (ETDS) o simplemente esquemas de traducción.	Indican el orden en el que se deben evaluar las reglas semánticas. Las acciones van intercaladas entre los símbolos de la gramática, es por ello que se dice que se especifican algunos detalles de la implementación.

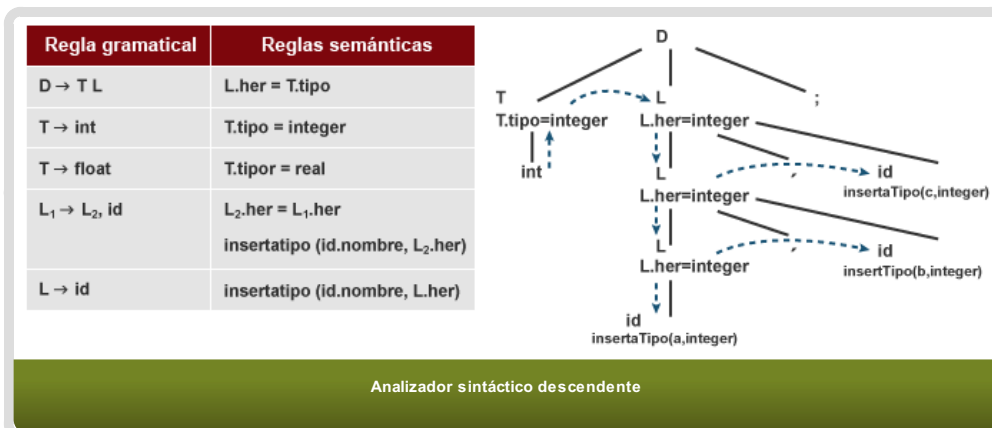
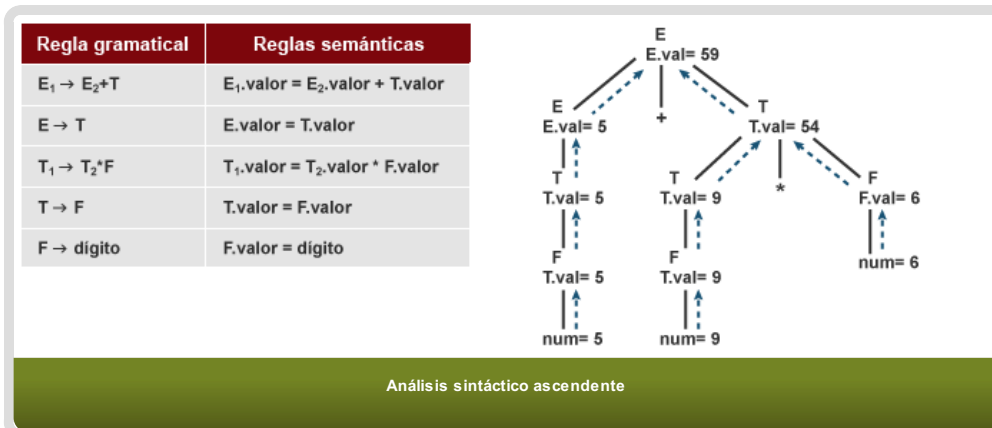
De las dos notaciones, la DDS es más fácil de entender y por tanto más útil para las especificaciones, aunque los ETDS son más eficientes y más útiles para las implementaciones (Aho et al, 2008).

Las **acciones semánticas** se encierran entre llaves, y la **posición** de una acción semántica en el lado derecho de una producción, indica el orden en el que se va a ejecutar la acción (va de izquierda a derecha).

La forma más fácil de decidir las acciones semánticas consiste en realizar un árbol sintáctico con una sentencia válida del lenguaje y en función de los controles semánticos que se quieran hacer, decidir los atributos y la producción donde se realizarán las acciones semánticas.

Definiciones dirigidas por la sintaxis

Tanto en las DDS como en los ETDS hay que tener en cuenta los tipos de atributos y si el análisis sintáctico es ascendente o descendente, principalmente en los ETDS puesto que el orden es importante. Puesto que el orden es importante, hay que establecer el orden en el que se evaluarán los atributos. Este orden dependerá de los valores de los atributos de los que depende el valor que queremos calcular.



Como vemos en la primera imagen, **todos los atributos son sintetizados y el análisis sintáctico es ascendente**, para una cadena de entrada: $5 + 9 * 6$.

En el caso de que estén mezclados los atributos heredados y sintetizados, se denomina **definiciones con atributos heredados**. Esta DDS es adecuada para usarse con un **analizador sintáctico descendente**, tal y como vemos en la segunda imagen.

Valores de los atributos

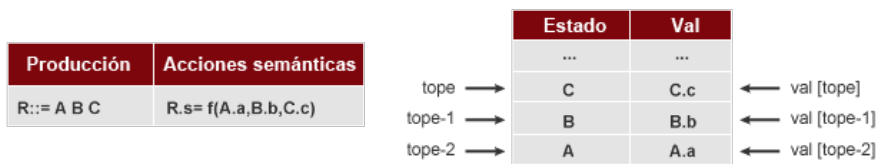
En el caso de que todos los atributos sean sintetizados, y puesto que por definición estos se calculan a partir de los atributos de los hijos de dicho nodo, entonces serán estos nodos hijo los primeros en evaluarse, y si el árbol de análisis se construye de forma ascendente, dará también igual que sea de izquierda a derecha (LR) o al revés. No es necesario, en este caso, construir el grafo de dependencias y en este caso se denomina **definiciones con atributos sintetizados**.

En el caso de que haya atributos heredados y sintetizados, hay que utilizar el grafo de dependencias y analizar caso por caso para establecer el orden del cálculo y por tanto las acciones semánticas a aplicar.

DDS. Evaluación de la gramática I

Con el objeto de ser lo más eficientes posibles, se debe intentar realizar todas las acciones semánticas durante el análisis sintáctico, puesto que esto nos permitirá evaluar la gramática en una sola pasada.

En el caso de las **definiciones con atributos sintetizados**, es por tanto una gramática con atributos sintetizados (S-A Grammar), el tipo de analizador sintáctico recomendado es el **ascendente** y la forma de evaluar esta gramática es utilizando una **pila**.



Las acciones asociadas serían (el atributo valor, se denomina val):

Regla gramatical	Reglas semánticas
$E_1 \rightarrow E_2 + T$	$ntope = tope - 2$ $val [ntope] = val [tope - 2] + val [tope]$ $tope = ntope$
$E \rightarrow T$	$E.val = T.val$
$T_1 \rightarrow T_2 * F$	$ntope = tope - 2$ $val [ntope] = val [tope - 2] * val [tope]$ $tope = ntope$
$T \rightarrow F$	$T.val = F.val$
$F \rightarrow digito$	$F.valor = digito$

Pila

La pila de estados LR tiene asociados los símbolos gramaticales, y mediante otra pila con los valores de los atributos sintetizados podemos ir realizando los cálculos (evaluación).

DDS. Evaluación de la gramática II

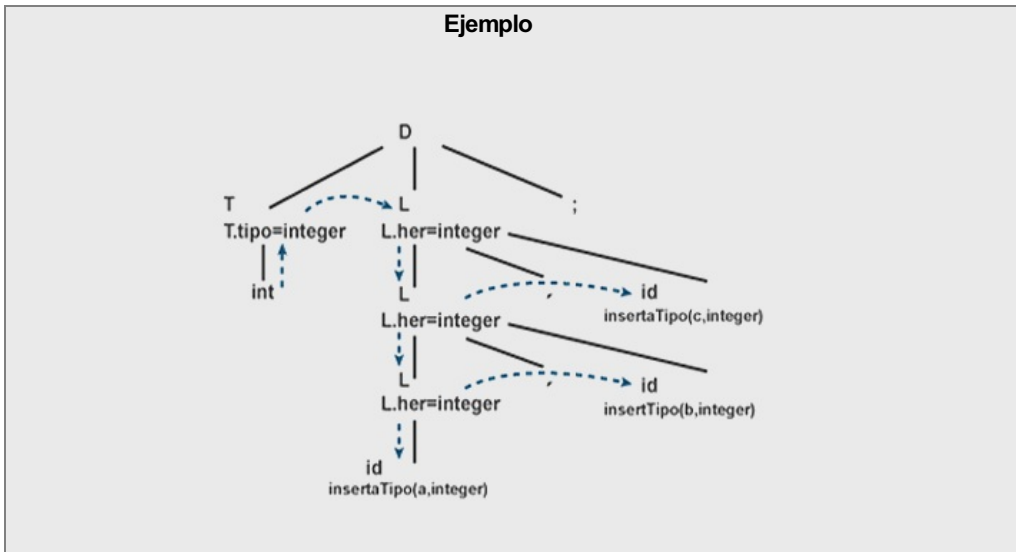
En el caso de las **definiciones con atributos heredados**, es por tanto una gramáticas con atributos por la izquierda (L-A Grammar, donde los atributos parecen ir de izquierda a derecha), como vimos en el [ejemplo](#), y el orden en el que se efectúa la evaluación de los atributos es el orden en el que se crean los nodos del árbol de análisis sintáctico.

A este tipo de gramáticas que utilizan atributos heredados, se las denomina **gramática de atributos por la izquierda (GAI)**, donde el flujo de atributos es de izquierda a derecha o de abajo a arriba y viceversa, pero nunca de derecha a izquierda. Recordemos que por convención, los atributos heredados siempre se colocan a la izquierda a la hora de construir el grafo de dependencias.

Si se diera el caso de que para calcular algún atributo, tuviéramos una dependencia de derecha a izquierda, habría que retrasar el cálculo del mismo hasta que se conozca el valor de los atributos de los que depende. Por tanto, habría que realizar una primera pasada para construir el árbol de análisis y una segunda pasada para realizar el cálculo de los atributos pendientes, lo que redundaría en una peor eficiencia. Si este fuera el caso, siempre se puede ver la posibilidad de modificar la gramática para que esta segunda pasada no sea necesaria.

Este tipo de gramáticas se evalúan adecuadamente con un analizador sintáctico descendente (no recursivo, preferentemente por cuestiones de rendimiento) como LL(1).





Esquemas de traducción dirigidos por la sintaxis

ETDS es la otra notación utilizada para realizar la traducción dirigida por la sintaxis. En este caso las acciones semánticas se intercalan entre los símbolos del lado derecho de la producción, de la siguiente manera:

- $X \rightarrow \alpha \{accion();\} \beta$

Como en el caso de la notación DDS, se presentan dos casuísticas:

EDTS solo con atributos sintetizados	Las acciones semánticas se incorporan al final de la producción, o bien después de los símbolos implicados.
EDTS con atributos sintetizados y heredados	Los atributos heredados provienen del lado izquierdo de la producción (padre) y los sintetizados provienen de los nodos hijo. En este caso las acciones relativas al cálculo de los atributos heredados deben colocarse antes de que aparezca el símbolo que utiliza este atributo en la parte derecha de la producción.

Puesto que el orden en los ETDS es importante, existen ciertas **restricciones**:

- Un atributo heredado para un símbolo en el lado derecho de una producción se calculará en una acción antes que dicho símbolo $A \rightarrow \alpha_1 B \alpha_2 \rightarrow \{B.h=f (...)\}$ (antes de B, en el lugar donde va el punto).
- Una acción no debe referirse a un atributo sintetizado de un símbolo que esté a la derecha de la acción $A \rightarrow \alpha_1 \alpha_2 \{A.a= \alpha_3.s\} \alpha_3$ (error, puesto que está a la derecha el símbolo α_3).
- Un atributo sintetizado para el no terminal de la izquierda solo puede calcularse después de que se hayan calculado todos los atributos a los que hace referencia (la acción se sitúa al final del lado derecho de la producción).



Ejemplo de ETDS incorrecto

Es importante destacar qué tipo de análisis (ascendente o descendente), puesto que es la forma en la que se crean los nodos del árbol de análisis sintáctico, y afecta al orden y a las acciones semánticas a realizar.

Ejemplo**Ejemplo de ETDS incorrecto**

- $S \rightarrow B_1 B_2 \{B_1.her=1; B_2.her=2\}$
- $B \rightarrow b \{imprimir(b.her)\}$

Como se puede observar estamos utilizando atributos heredados y las acciones las ponemos después del símbolo cuando tienen que estar a su izquierda. **¿Cómo lo solucionamos?**

- $S \rightarrow \{B_1.her=1;\} B_1 \{B_2.her=2;\} B_2$
- $B \rightarrow \{imprimir(b.her)\} b$

ETDS. Atributos heredados por la izquierda y evaluación descendente

Como nos indica Aho et al (1986), la mayoría de los operadores aritméticos son asociativos por la izquierda, es por tanto natural utilizar gramáticas recursivas por la izquierda para estas expresiones. Por ello se nos presentan dos problemas, primero el resolver la **recursividad con las acciones semánticas intercaladas** y por otro **pasar de atributos sintetizados**, que son los recomendables para hacer un análisis sintáctico ascendente, a la utilización de atributos heredados para poder realizar un análisis sintáctico descendente que es el que mejor encaja con una gramática de atributos heredados por la izquierda.

Eliminación de la recursividad

Partamos de la siguiente gramática y sus acciones asociadas (recuerda):

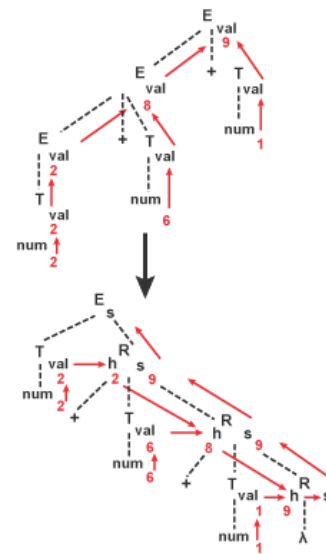
- $E_1 \rightarrow E_2 + T \{E_1.val = E_2.val + T.val\}$
- $E \rightarrow T \{E.val = T.val\}$
- $T \rightarrow \text{dígito} \{T.val = \text{dígito.val}\}$

Partimos del árbol de arriba y queremos llegar al de abajo utilizando atributos heredados y sintetizados realizando la operación: $2 + 6 + 1$

En este caso aplicando la regla general de eliminación de la recursividad, y la incorporación de atributos heredados, las acciones quedarían de la siguiente forma (recuerda):

- $E \rightarrow T \{R.h = T.val\} R \{E.valor = R.s\}$
- $R_1 ::= + T \{R_2.h = T.val + R_1.h\} R_2 \{R_1.s = R_2.h\}$
- $R ::= \lambda \{R.s = R.h\}$
- $T ::= \text{dígito} \{T.val = \text{dígito.val}\}$

Una vez que se tiene claro qué se quiere obtener, a partir del árbol se ubican las acciones semánticas teniendo en cuenta las restricciones mencionadas para los atributos sintetizados y heredados respecto a la realización de los cálculos.



Recuerda I

Recordemos que en una producción del tipo $A \rightarrow A\alpha \mid \beta$, la eliminación de la recursividad dejaba las siguientes producciones:

- $A \rightarrow \beta A'$
- $A' \rightarrow \alpha A \mid \lambda$

Recuerda II

Partíamos de $E \rightarrow E + T$, la eliminación de la recursividad deja las siguientes producciones:

- $E \rightarrow TR$
- $R \rightarrow + TR \mid \lambda$

ETDS. Atributos heredados por la izquierda y evaluación ascendente

Ahora estamos en el caso en el que tenemos atributos heredados por la izquierda y una evaluación descendente LL(1) y queremos realizar una evaluación ascendente mediante un LR (Aho et al, 1986 y Aho et al, 2008). Tenemos dos formas:

- **Incorporar símbolos marcadores:** así eliminamos acciones intercaladas. Estos marcadores son símbolos no terminales de los que se deriva la palabra vacía (λ) y que aparecerán solo una vez por cada atributo heredado que queramos eliminar.
- **Sustitución de atributos heredados por sintetizados:** esto implica modificar la gramática y por tanto hay que estudiarlo para cada caso.

Incorporación de símbolos marcadores

Cuando no hay atributos heredados, la transformación es inmediata:

$$A \rightarrow X_1 \dots X_{n-1} \{acción\} X_n \quad \Rightarrow \quad \begin{array}{l} A \rightarrow X_1 \dots X_{n-1} M X_n \\ M \rightarrow \lambda \{acción\} \end{array}$$

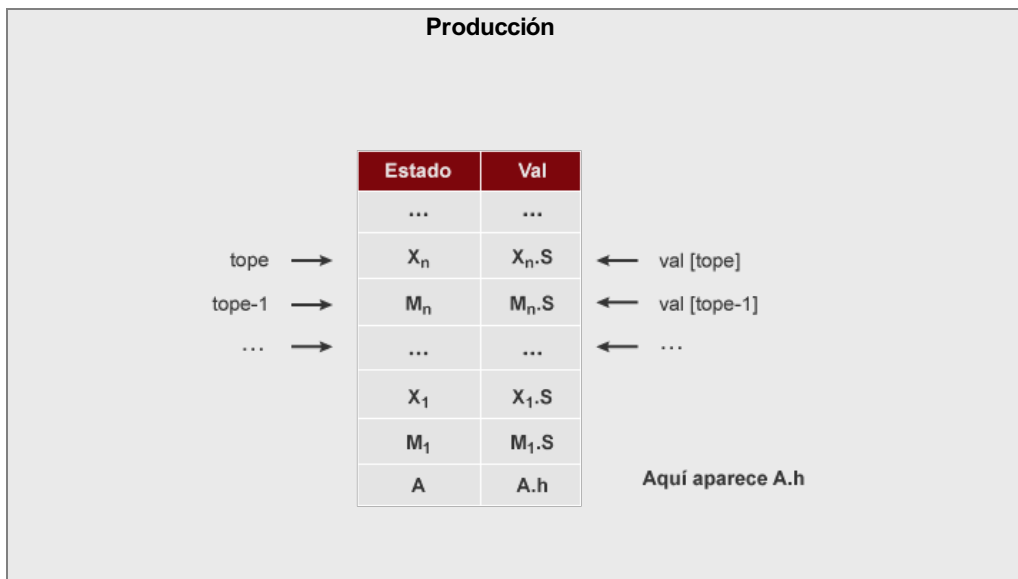
Ejemplo:

$\begin{array}{l} E ::= T E' \\ E' ::= + T \{imprime +\} E' \\ E' ::= \lambda \\ T ::= \text{dígito} \{imprime num.lex\} \end{array}$	\Rightarrow	$\begin{array}{l} E ::= T E' \\ E' ::= + T M E' \\ E' ::= \lambda \\ T ::= \text{dígito} \{imprime num.lex\} \\ M ::= \lambda \{imprime +\} \end{array}$
---	---------------	--

Cuando hay atributos heredados, se introduce un marcador por cada símbolo con atributo heredado:

$$A \rightarrow X_1 \dots X_n \quad \Rightarrow \quad \begin{array}{l} A \rightarrow M_1 X_1 \dots M_n X_n \\ \text{(si todo } X_i \text{ tiene atributo heredado)} \end{array}$$

Se debe tener en cuenta que todo atributo heredado está en el sintetizado del no terminal marcador, de tal forma que si utilizamos la pila sincronizada con la pila de estados de los LR, veremos que todos los atributos heredados están en posiciones predecibles en la pila (incluyendo la parte izquierda de la producción). Supongamos la siguiente producción: $A \rightarrow M_1 X_1 \dots M_n X_n$

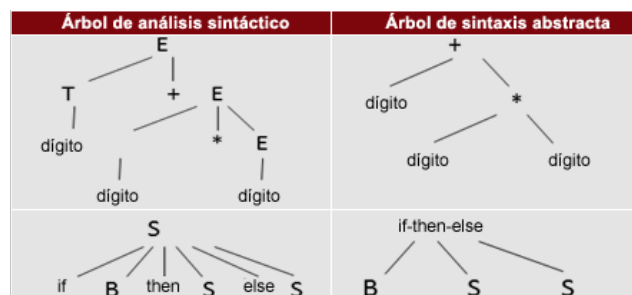


Creación del árbol sintáctico

Si no se puede evaluar la gramática a la vez que se realiza el análisis sintáctico, tendremos que evaluar en varias pasadas, en función del orden que nos marque el grafo de dependencias.

Como nos indica Aho et al (1986) el uso de árboles sintácticos como representación intermedia permite que la traducción se separe del análisis sintáctico. Es importante indicar, que los tipos de árboles que nos interesan son los **árboles de sintaxis abstracta (ASA)** que es un árbol de análisis sintáctico al que se le han eliminado los símbolos superfluos y es muy útil para representar las construcciones del lenguaje.

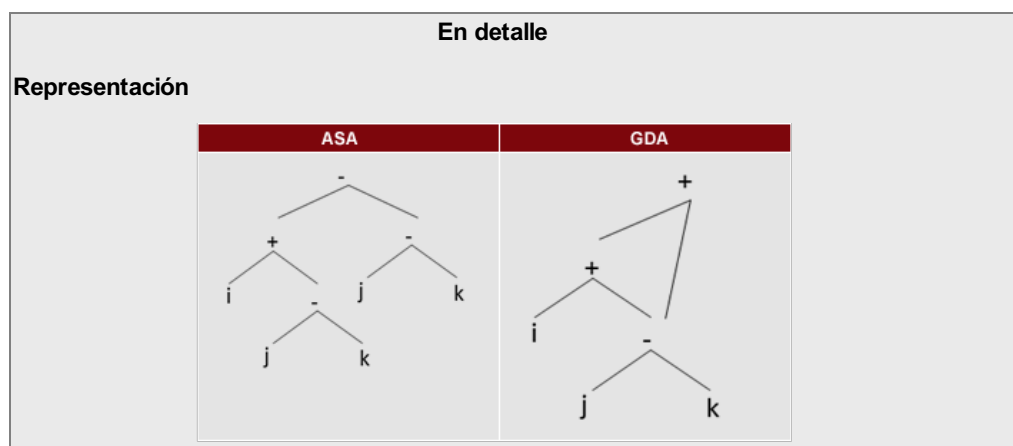
Veamos los siguientes ejemplos:



Como podemos ver, en un ASA nos interesan las operaciones y los operadores.

Otro tipo de gráfico que se suele utilizar es el **grafo dirigido acíclico (GDA)** cuyo objetivo es no repetir estructuras del ASA. En este tipo de grafos, sus nodos interiores representan a los operadores mientras que los nodos hijo representan a los operandos. Cada vez que se repite una estructura se referencia en lugar de crearla nueva otra vez. En este caso vamos a representar la siguiente operación: $i + (j - k) - (j - k)$

 [Representación](#)
En detalle



Resumen

En este tema hemos comprendido las tareas necesarias para realizar una traducción dirigida por la sintaxis, que no es otra cosa que la asociación de las reglas semánticas con la estructura sintáctica, utilizando dos notaciones: DDS y ETDS. Después se ha visto las características de los DDS y ETDS utilizando tanto atributos sintetizados como heredados, así como la necesidad en este último caso de los grafos de dependencias.

Hemos visto cómo se puede utilizar una pila sincronizada con la pila de estados para gestionar los atributos y poder realizar su evaluación a la vez que se realiza el análisis sintáctico.

Por otro lado, se ha podido observar que los atributos sintetizados son los más adecuados para un análisis sintáctico ascendente, mientras que los heredados encajan mejor en el análisis sintáctico descendente. Además se ha podido ver cómo en los ETDS el orden en el que se realicen las acciones es importante, sobre todo cuando hay atributos sintetizados y heredados en la gramática y por tanto hay que tener en cuenta una serie de restricciones a la hora de intercalar las acciones entre la parte derecha de las producciones.

Por último, es importante indicar, que los tipos de árboles que nos interesan son los árboles de sintaxis abstracta (ASA) que es un árbol de análisis sintáctico al que se le han eliminado los símbolos superfluos y es muy útil para representar las construcciones del lenguaje.