

UNIVERSIDAD POLITECNICA SALESIANA

Narcisa Araujo

El Uso de Prolog

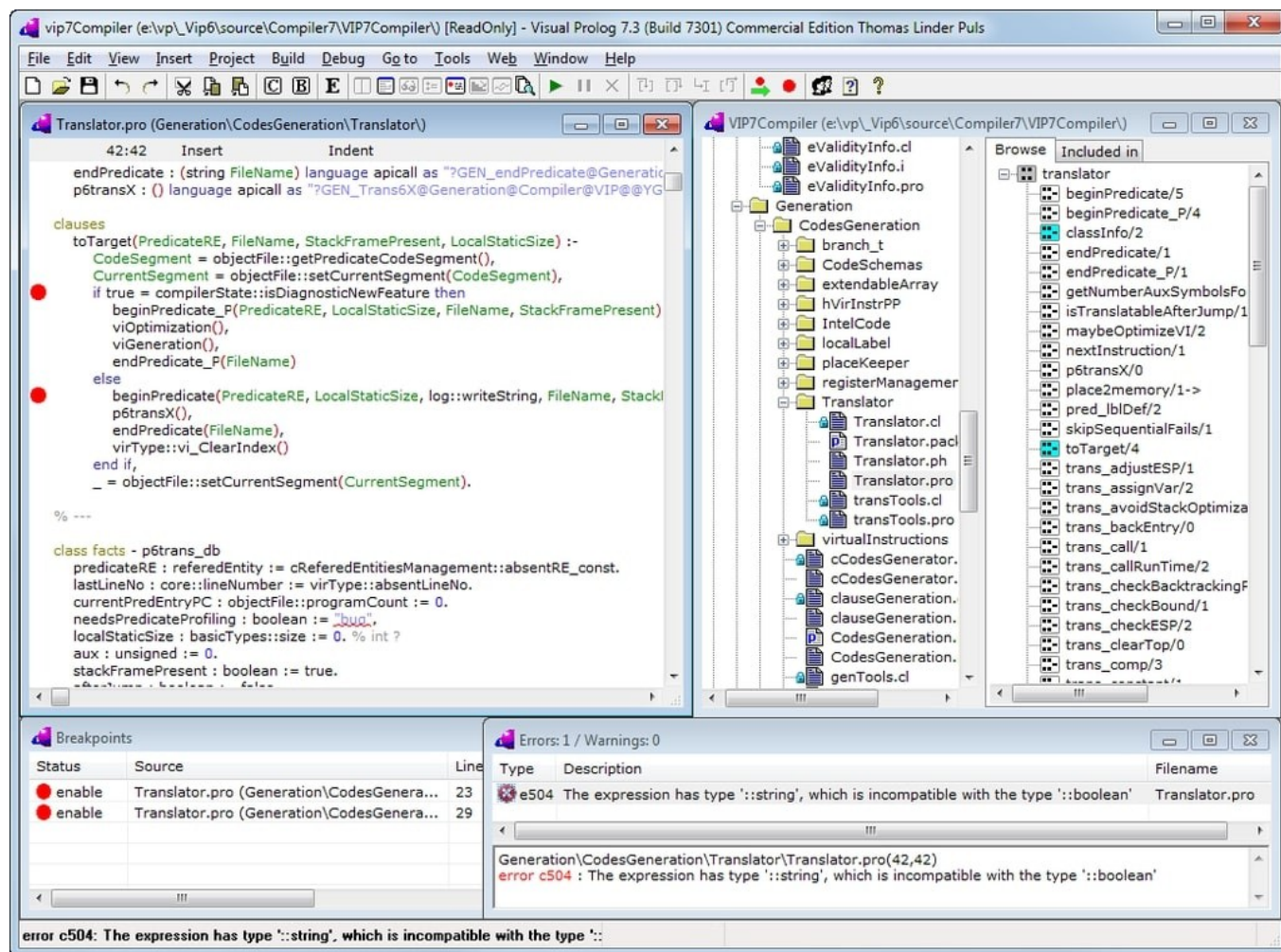


¿Qué es Prolog?

Prolog (derivado de "PROgrammation en LOGique") nació a comienzos de la década de los 70, de la mano de un investigador de la Universidad de Marsella (Alain Colmerauer), mientras dirigía un proyecto de tratamiento algorítmico de lenguajes naturales.

Inicialmente se trataba de un lenguaje interpretado hasta que, una década más tarde, se creó un compilador capaz de traducir Prolog en un conjunto de instrucciones de una máquina abstracta denominada WAM ("Warren Abstract Machine"), lo que lo convirtió desde entonces en un lenguaje semi-interpretado.

Los ámbitos de la computación donde más se usa Prolog son los de la inteligencia artificial y aspectos relacionados con la misma como el machine learning, procesamiento de lenguaje natural, construcción de sistemas expertos, etc.



Sintaxis de Prolog

Un programa Prolog se compone de un conjunto de hechos (afirmaciones simples) y de reglas (que sirven para afirmar la veracidad de un hecho en base a otros). El conjunto de hechos de un programa viene a ser el equivalente a contar con una base de datos, aunque en este contexto se habla de 'base de conocimientos'.

Ejemplo de hechos

`es_hijo(Juanito, Juan)`

`suma(10,5,15)`

Una vez definidos estos hechos (el primero significa "Juanito es hijo de Juan" y el segundo "15 es la suma de 10 y 5"), se pueden realizar preguntas al programa, que terminarán siempre en un punto, como el siguiente:

`?es_hijo(x,Juan).`

Dicha pregunta significa "¿Existe un X que sea hijo de Juan?"; la respuesta del programa sería positiva, y nos arrojaría todas las respuestas posibles ("Juanito"... y el resto de sus hermanos, si los hubiera y los hubiéramos definido como hechos).

Igualmente, a partir de lo anterior, podemos también establecer reglas, que definen nuevas relaciones a partir de relaciones ya dadas. Por ejemplo:

```
es_padre(X,Y) :- es_hijo(Y,X), es_varón(X).
```

Este código Prolog se correspondería con la fórmula lógica " $\forall x \forall y ((\text{es_hijo}(y,x) \wedge \text{es_varón}(x)) \rightarrow \text{es_padre}(x,y))$ ".

Prolog cuenta también con objetos estructurados:

```
en_stock( libro(harry_potter_1, autor(j.k., rowling)) ).
```

Lo anterior nos permite preguntar al programa por varios elementos de la entidad: "¿Tenemos 'harry_potter_1' en stock?", "¿Tenemos algún libro de 'j.k. rowling' en stock?", etc.

No pretendemos con este artículo más que echar un leve vistazo a un enfoque minoritario de la programación y a uno de sus principales ejemplos. Prolog es mucho más complejo y potente que lo que dejamos entrever aquí: si te pica la curiosidad estás de enhorabuena, porque una visita rápida a Google te revelará grandes cantidades de documentación al respecto.

Bibliografía

<https://www.genbeta.com/desarrollo/lenguaje-prolog-ejemplo-paradigma-programacion-logica>
(<https://www.genbeta.com/desarrollo/lenguaje-prolog-ejemplo-paradigma-programacion-logica>).

In []: