

An External Module for Implementing Linear Tabling in Prolog

Cláudio Silva, Ricardo Rocha and Ricardo Lopes

DCC-FC & LIACC

University of Porto, Portugal

ccaldas@dcc.online.pt

{ricroc,rslopes}@ncc.up.pt

Motivation

- This work was motivated by our recent proposal of combining **tabling** with the **Extended Andorra Model (EAM)** [Rocha *et al.*, ICLP'2005].
- The first task in that proposal was to develop an external module for implementing tabling primitives that provide direct control over the search strategy.
- The external module will serve as the basis to study how tabling interacts with the EAM and to detect in advance the potential integration problems before incorporating tabling within the EAM environment.

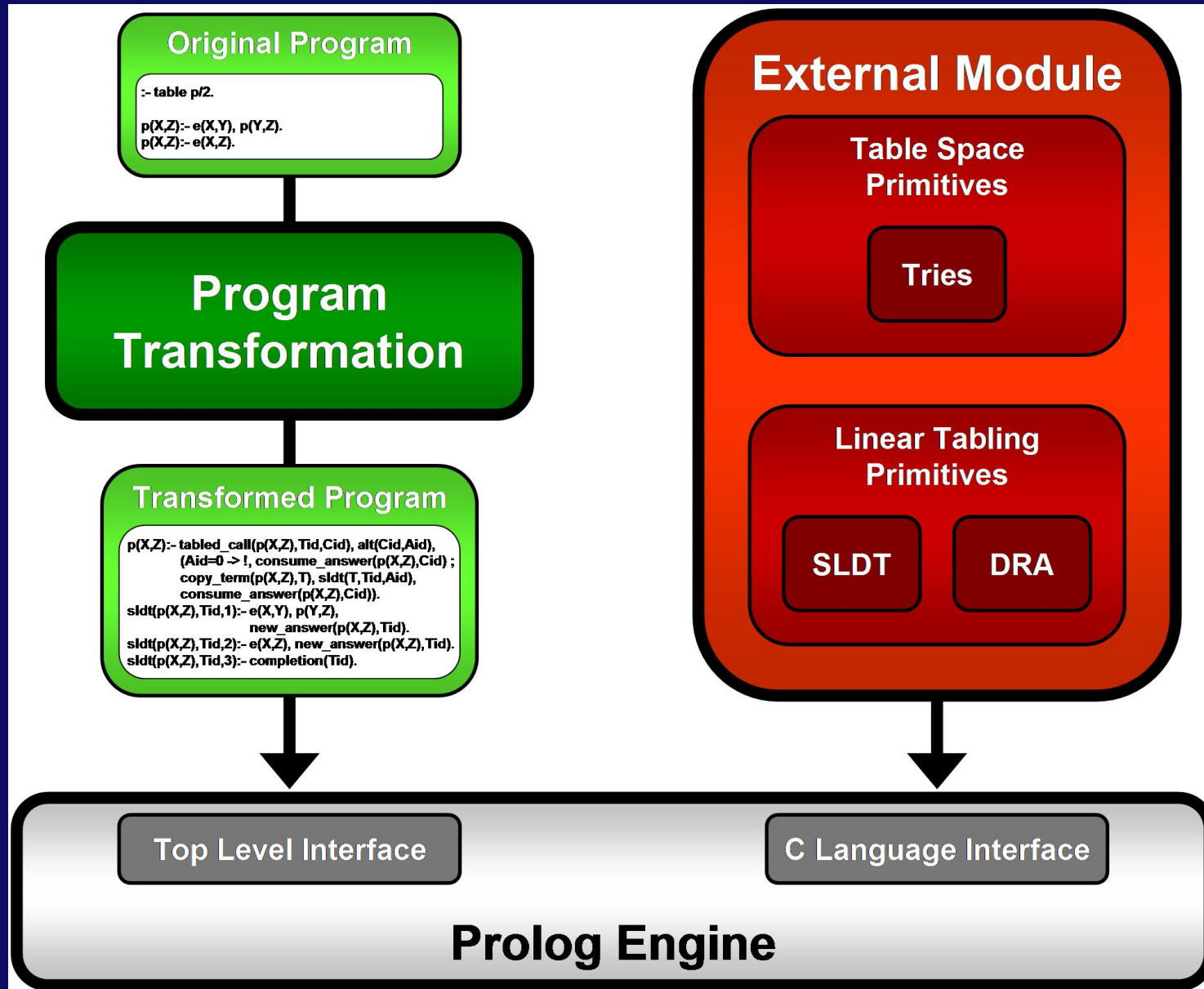
Tabling Mechanisms

- We can distinguish two main categories of tabling mechanisms:
 - ◆ **Delaying-Based Tabling** where tabled calls can be suspended and later resumed (we need to preserve the computation state of suspended calls).
 - ◆ **Linear Tabling** where tabled calls always compute starting from the previous call (only a single tree is maintained in the execution stacks).
- Linear-based mechanisms are considered to be less efficient but are easier to implement.

Our Module

- Our module implements the two available mechanisms that, to the best of our knowledge, implement linear tabling:
 - ◆ **SLDT Strategy** [Neng-Fa Zhou *et al.*, PADL'2000]
 - ◆ **DRA Technique** [Hai-Feng Guo and G. Gupta, ICLP'2001]
- The module uses the **C language interface** of the Yap Prolog system to define the **tabling primitives** that implement each tabling mechanism.
- To use the module, a tabled logic program is first **transformed** to include the corresponding tabling primitives and only then the resulting program is compiled.

Module Architecture



Concluding Remarks

- Our module is independent from the Yap Prolog's engine which makes it easily portable to other Prolog systems with a C language interface.
- Starting from these two tabling mechanisms, we are now working on a new proposal that tries to combine the best features of both.