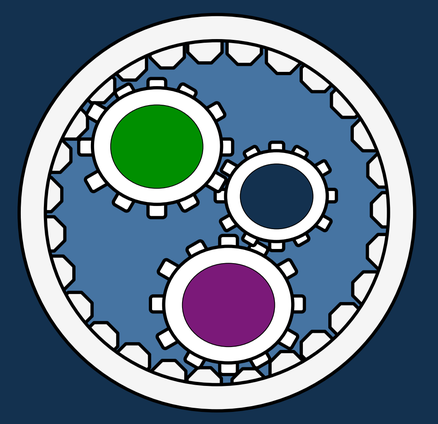
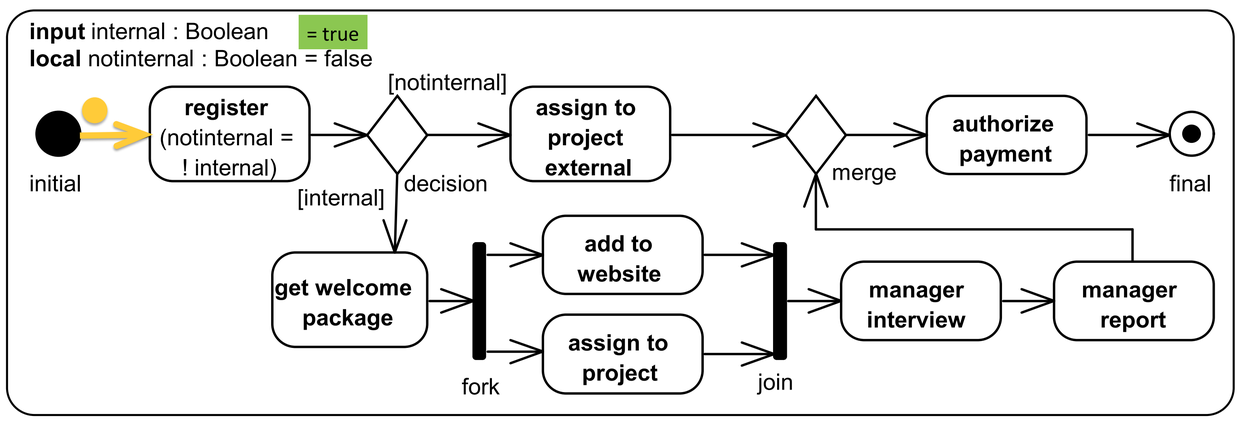
fUML Activity Diagrams with RAG-controlled Rewriting

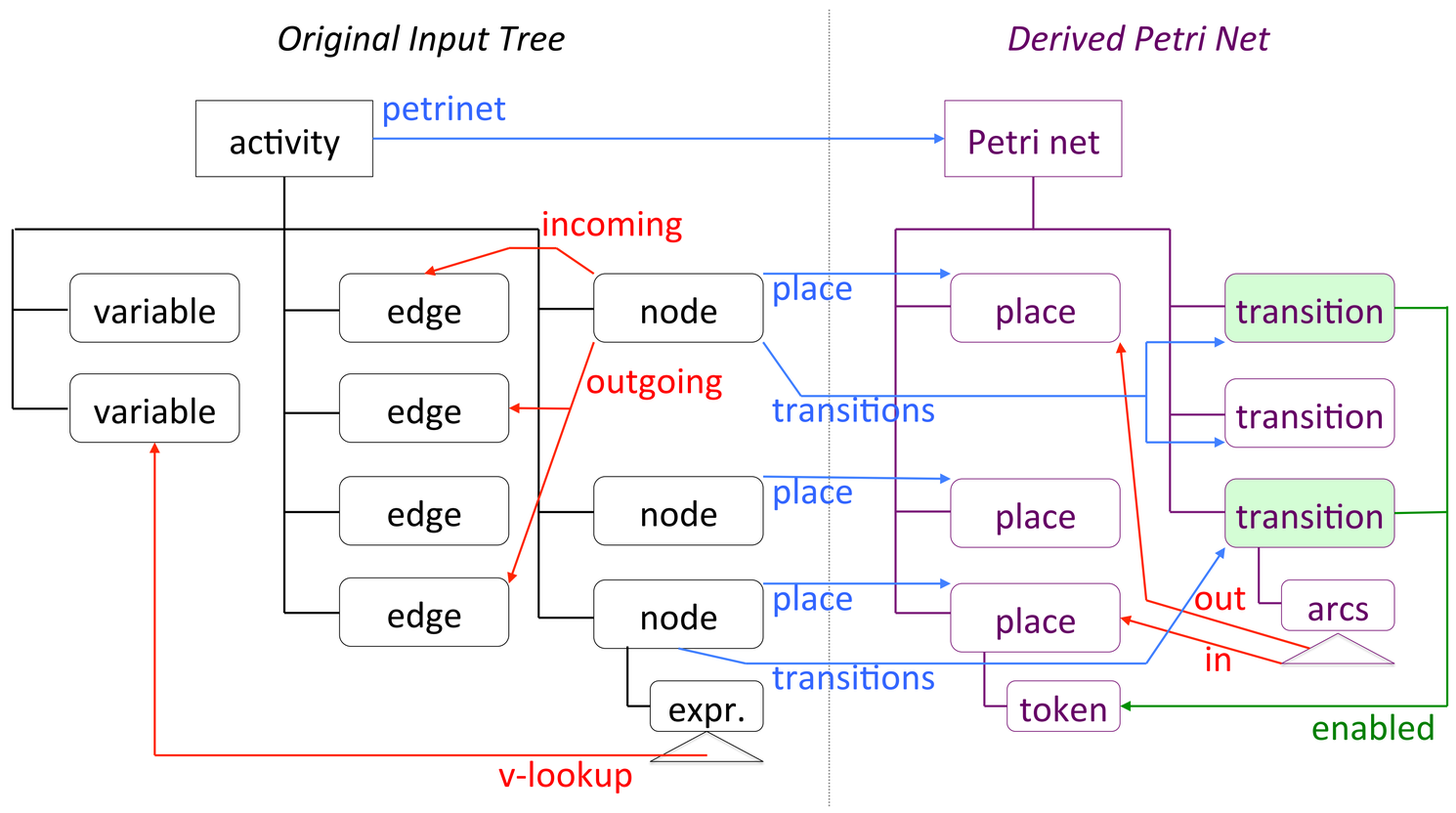


A *RACR* Solution of the *8th Transformation Tool Contest*

TTC 2015 task: execution of *fUML Activity Diagrams.*



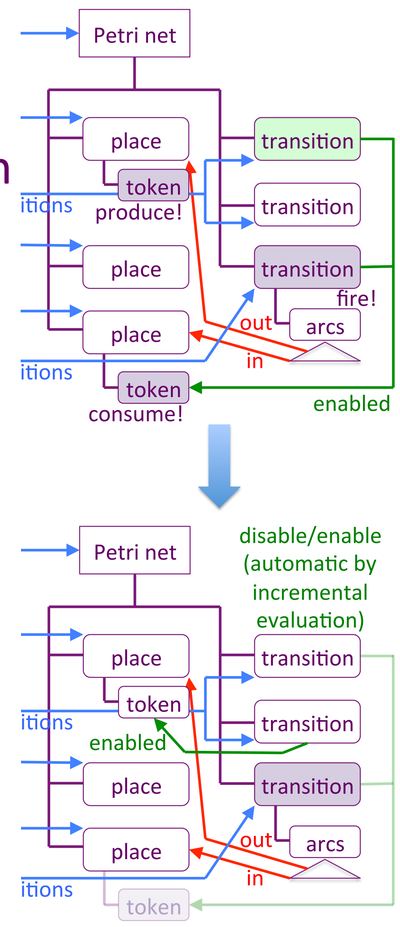
RACR Solution: diagram to Petri net interpreter, uses a reference attribute grammar to deduce memoized abstract syntax graph well-suited for execution by rewriting.



memoized, deduced abstract syntax graph

reuse of enabled analysis permits convenient, focused rewrite specification

incremental enabled analysis



RAG-controlled rewriting: declarative, seamless combination of reference attribute grammars & graph rewriting (*RACR*: reference implementation, *Scheme* library).

- transformation-aware RAG-based analyses (incremental evaluation)
- analyse-aware rewrite-based transformations (analyses deduce rewrites)

mutual control


8th Transformation Tool Contest

TTC 2015

Contest Award

This document certifies that the award for

THE OVERALL QUALITY AWARD FOR THE MODEL EXECUTION CASE STUDY

has been won by

RACR

Participating team members:

CHRISTOFF BÜRGER

Location: L'Aquila, Italy
Date: 24.07.2015
Organizing Committee:
Tassilo Horn, Filip Krikava, Louis Rose

Christoff Bürger (christoff.buerger@cs.lth.se, https://github.com/christoff-buerger/racr)
Dept. of Computer Science, Faculty of Engineering, LTH, Lund University, Lund, Sweden