Evolution of the MARGe Project

Joshua Dourado Capistrano
Computational Engineering — University of Aveiro

The MARGe project has evolved from a simple reactive graph animation tool into an integrated environment for modeling, simulation, and formal verification of reconfigurable systems.

This evolution has surpassed the initial goals, incorporating dynamic modal logic and advanced semantic extensions.

Fundamental Axes of Contributions

1. Extension of the core formalism (guards and time)

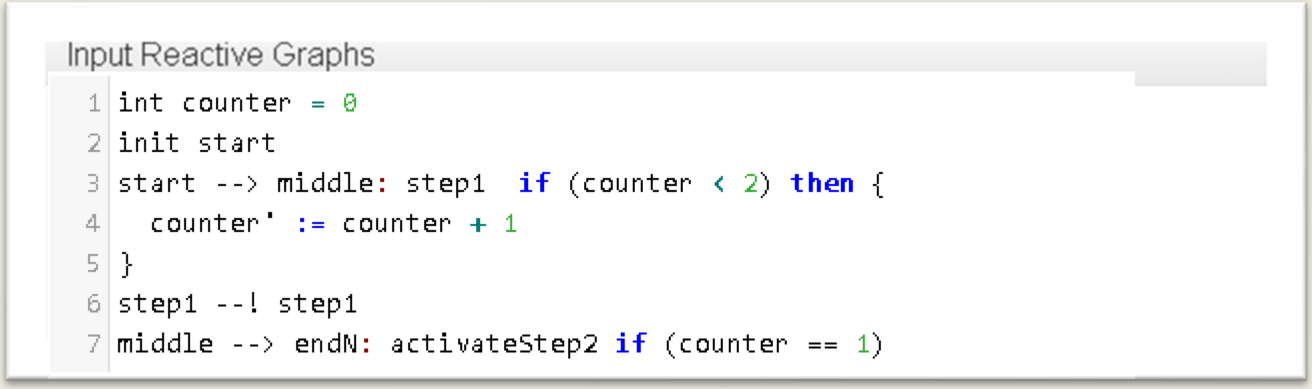
2. Introduction of formal verification capabilities.

3. Formal mappings and interoperability with other tools.

Extension of the Reactive Graph Formalism

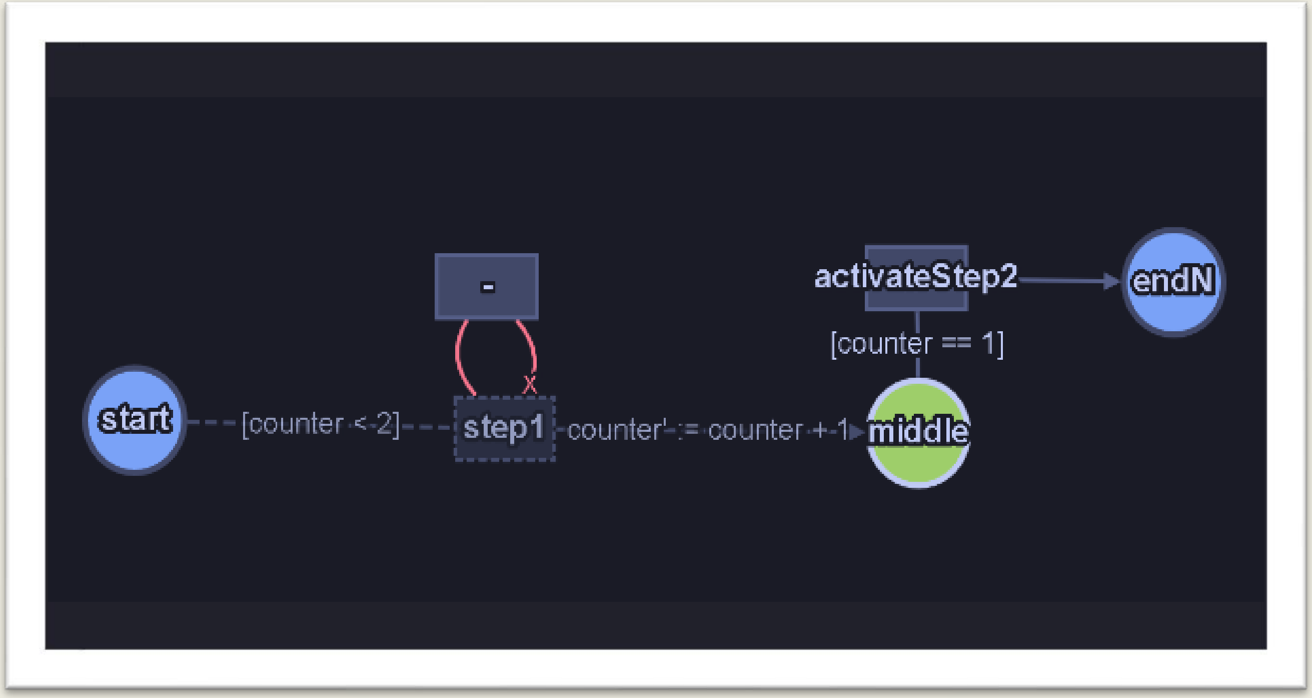
- The Reactive Graphs formalism was expanded with guards and time, giving rise to two new formalisms:
  - Guarded Reactive Graphs (GRG): enable logical conditions and atomic updates.
  - Timed Reactive Graphs (TRG): introduce clocks and temporal invariants.

Guarded Reactive Graphs (GRG)

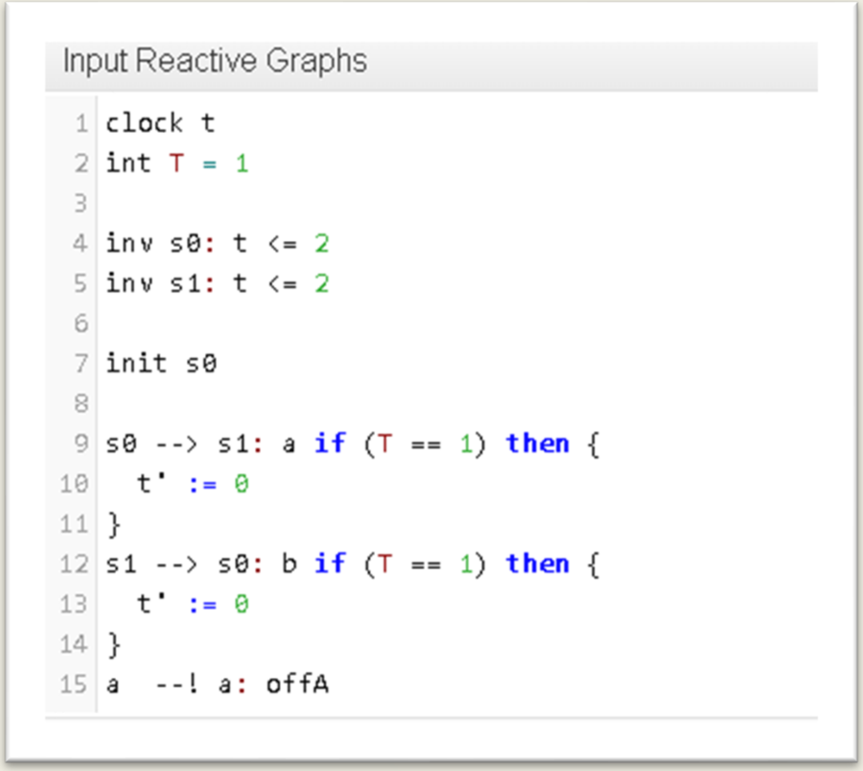


Input Reactive Graphs

```
int counter = 0
init start
start --> middle: step1  if (counter < 2) then {
  counter' := counter + 1
}
step1 --! step1
middle --> endN: activateStep2 if (counter == 1)
```

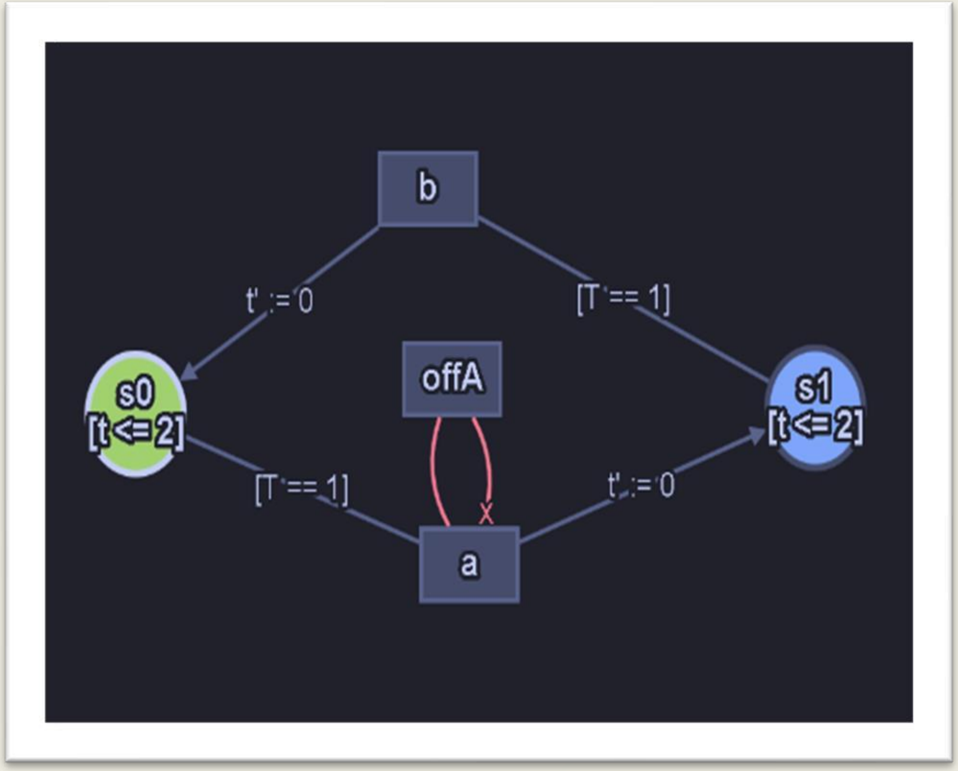


Timed Reactive Graphs (TRG)



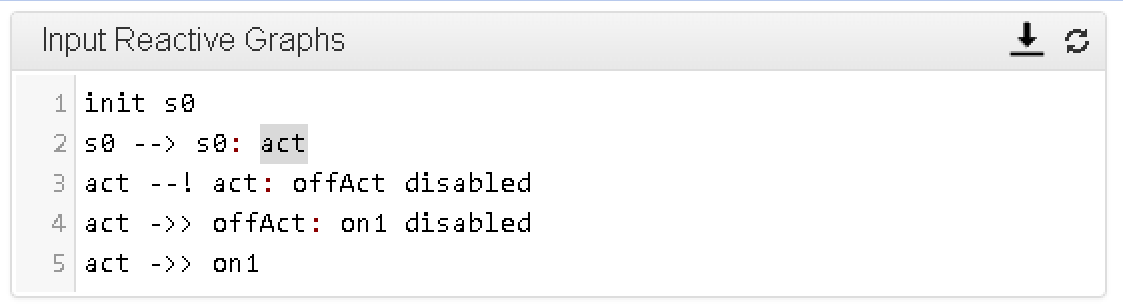
Input Reactive Graphs

```
clock t
int T = 1

inv s0: t <= 2
inv s1: t <= 2

init s0

s0 --> s1: a if (T == 1) then {
  t' := 0
}
s1 --> s0: b if (T == 1) then {
  t' := 0
}
a  --! a: offA
```

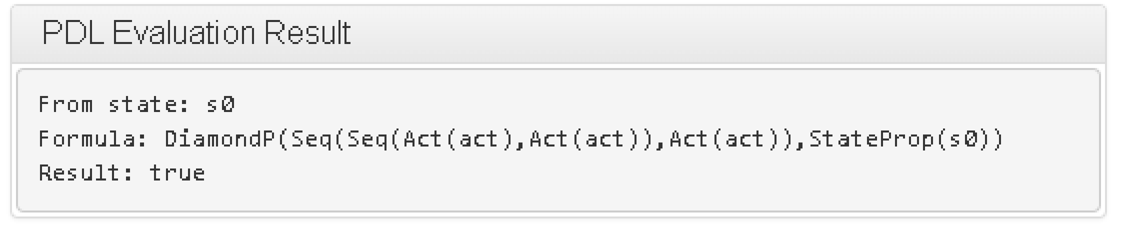


2. Formal Analysis and Property Verification

- Implementation of a verifier based on Propositional Dynamic Logic (PDL).



```
init s0
s0 --> s0: act
act --! act: offAct disabled
act ->> offAct: on1 disabled
act ->> on1
```



```
From state: s0
Formula: DiamondP(Seq(Seq(Act(act),Act(act)),Act(act)),StateProp(s0))
Result: true
```

Thank you!

Questions?