Formal Verification of Distributed Algorithms using Distributed PlusCal

Student: Heba Al-Kayed
Supervisors : Stephan MERZ, Horatiu CIRSTEA
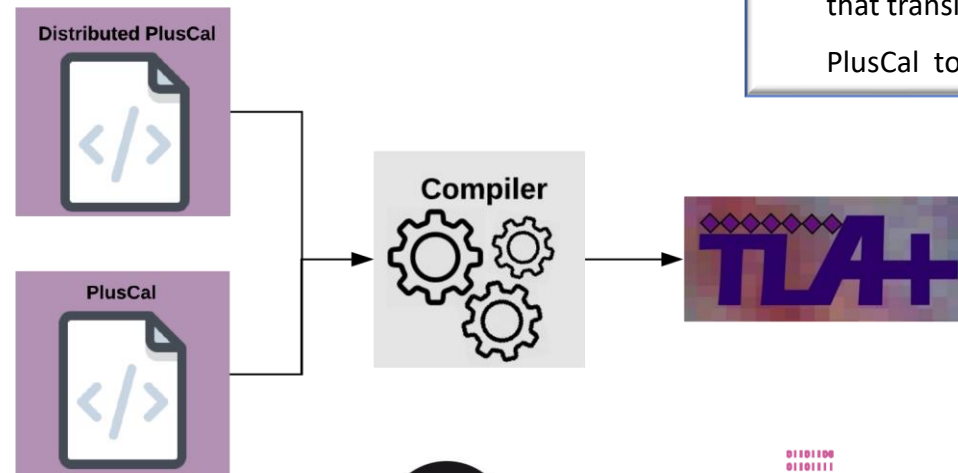
Introduction

- Designing sound algorithms for concurrent and distributed systems is subtle and challenging. These systems are prone to deadlocks and race conditions.

- As an aid for the design and implementation of distributed algorithms, formal verification methods have been employed successfully to model the system and its properties and then verify its correctness.

- TLA+[1] is a formal language used to describe algorithms, it provides a flexibility and an expressiveness that enables it to specify and verify complicated algorithms concisely.

- TLA+ relies on mathematical logic and formulas for structuring specifications, which may discourage programmers from using it; therefore, PlusCal[2] was designed as an algorithm language with a more familiar syntax that can be translated into TLA+ specifications.

Motivations

- When modeling distributed algorithms using PlusCal, it may enforce some limitation that make it difficult to express them in a natural way. For example PlusCal has a flat hierarchy of processes while distributed algorithms need to model sub-processes coexisting and communicating as a part of a distributed node.

Results

- We propose Distributed PlusCal, an extension of PlusCal that introduces constructs that can overcome some of the limitation of PlusCal.

- Distributed PlusCal is a language that provides algorithm designers with an interface in which distributed algorithms and their properties can be expressed naturally.

- We provide a backward compatible translator that translates from Distributed PlusCal and PlusCal to TLA+.

References

[1]Lamport, Leslie. (2000). Specifying Concurrent Systems with TLA+.
[2] Lamport, Leslie. (2009). The PlusCal Algorithm Language. 5684. 36-60.
10.1007/978-3-642-03466-4_2.