Distributed LNT Compiler (DLC): Compiling a Concurrent System Formal Specification to a Distributed Implementation

Hugues EVRARD

Team CONVECS
Inria Grenoble Rhône-Alpes & LIG, France

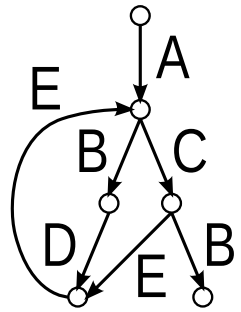
April 6, 2016

TACAS'16 - Eindhoven

Introduction

- Concurrency: *several* processes which *interact*

- Formal model in **process algebra** style
  - process observable event: **action** (on *gate*)
  - formal semantics: *Labelled Transition System*
  - **non-deterministic** process: ready on several actions at the same time
  - process interaction: **multiway rendezvous** upon action

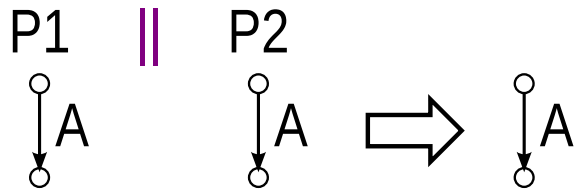


- Main goal: from the **same formal model** (LNT), both **verification** (CADP) and **code generation** (DLC)
  - **LNT:** modern process algebra
  - **CADP:** verification toolbox
  - **DLC: Distributed LNT Compiler**
    - ★ implements multiway rendezvous over asynchronous message-passing (distributed: TCP sockets)

Interaction by Multiway Rendezvous

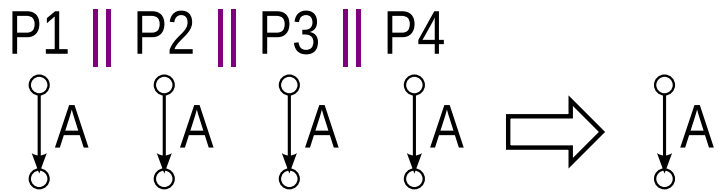
- Binary rendezvous

```
par A in
   P1 || P2
end par
```



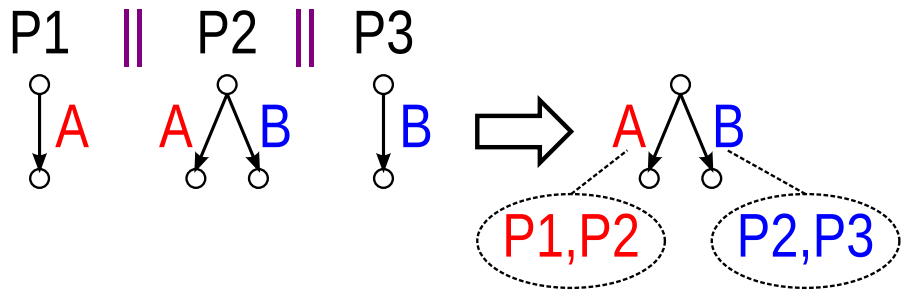
- Multiway rendezvous

```
par A in
  P1 || P2 || P3 || P4
end par
```



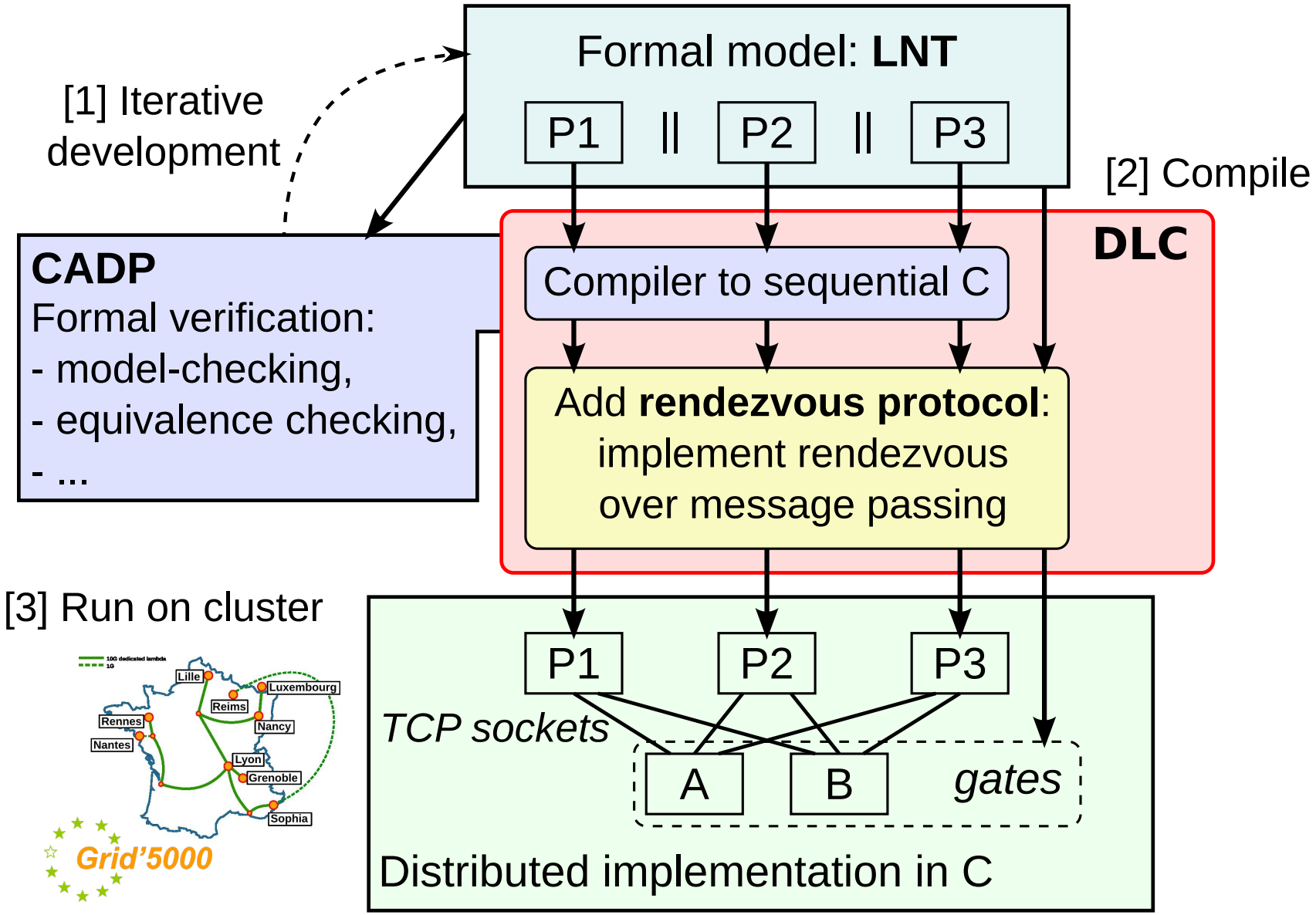
- Rendezvous in **conflict**

```
par
   A     -> P1
|| A, B  -> P2
|| B     -> P3
end par
```



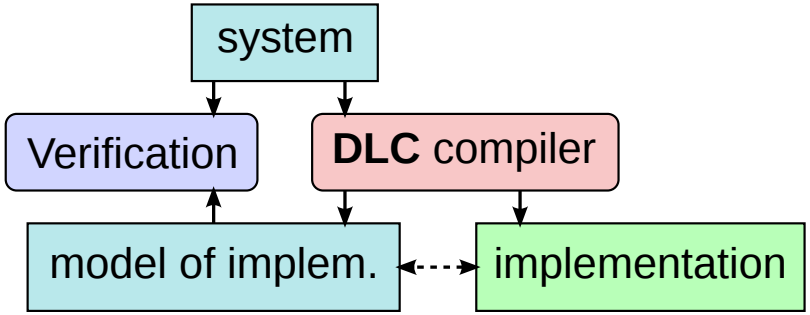
  - *Mutual exclusion* between rendezvous in conflict

Distributed LNT Compiler (DLC) — Overview



Rendezvous Protocol

- Based on *Parrow-Sjödin-96*
- We **verify** rendezvous protocol
  - detected possible deadlocks

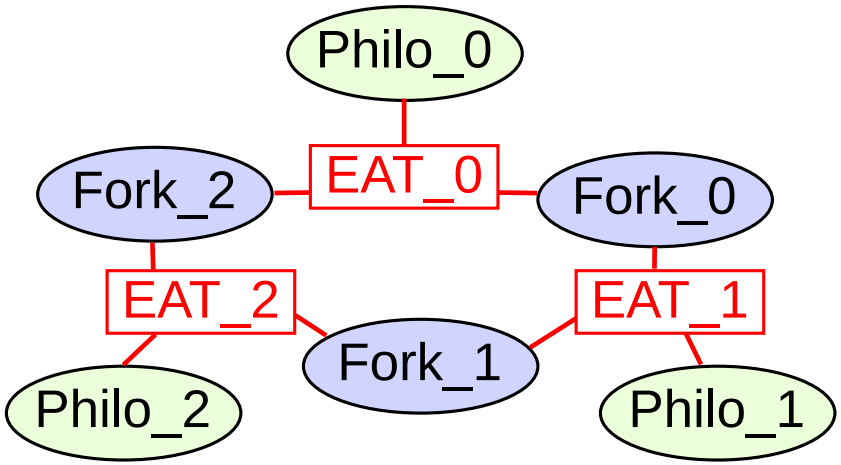


- Iterative **improvements** (verif. at each step), including:
  - **correction** to avoid deadlocks
  - **reduce** message types from 9 to 4
  - **broadcast** of results
  - **autolock** and **purge** optimization
- Protocol complexity

Protocol	Total messages
Parrow-Sjödin-96	$5n$
Perez-Corchuelo-Toro-04	$4n - 2k + 2\sum_{i=1}^{n}(p_i - 1)$
DLC	$3n - k$

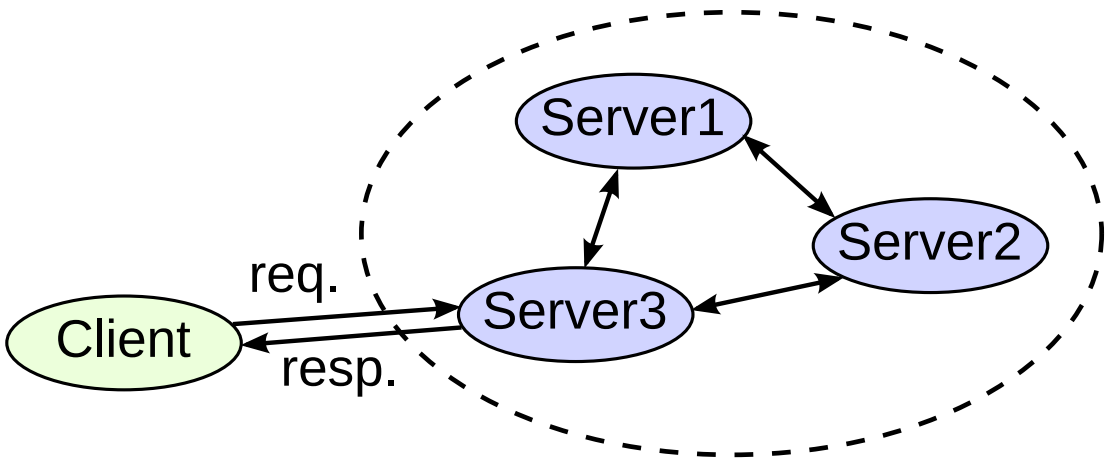
Demo 1. Dining Philosophers

- Philosophers with multiway rendezvous
  - 3-way rendezvous between a philosopher and its surrounding forks



Demo 2. Raft Consensus

- Raft Consensus algorithm *Ongaro-Ousterhout-14* (Stanford)
  - Paxos-like
  - quick adoption: etcd (CoreOS), Consul (Hashicorp), ...
- Fault-tolerant service (replicated state-machine)
  - maintain a replicated log of client command
  - Raft: Leader election, Heartbeat
  - toy application: store of a boolean value



- Detected issue in author's original specification (TLA+)

Summary

- From the same formal model (LNT):
  - perform formal verification (CADP)
  - automatically get a distributed implementation (DLC)
- New tool: **Distributed LNT Compiler**
  - `http://hevrard.org/DLC`
  - state-of-the-art **verified** and **efficient** rendezvous protocol
- References
  - H. Evrard, *DLC: Compiling a Concurrent System Formal Specification to a Distributed Implementation*, TACAS 2016
  - H. Evrard, *Génération automatique d'implémentation distribuée à partir de modèles formels de processus concurrents asynchrones*, (PhD Thesis) Université Grenoble-Alpes, 2015
  - H. Evrard, F. Lang, *Automatic distributed code generation from formal models of asynchronous concurrent processes*, PDP-4PAD 2015
  - H. Evrard, F. Lang, *Formal verification of distributed branching multiway synchronization protocols*, FORTE 2013

Thank you for your attention