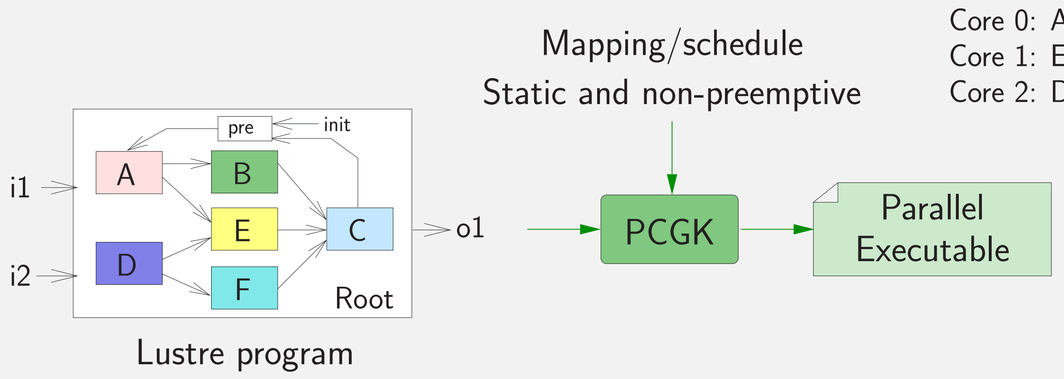
Amaury Graillat[1,2], Matthieu Moy[3],
Pascal Raymond[1], Benoît Dupont de Dinechin[2]

[1] Univ. Grenoble Alpes / Verimag, France
[2] Kalray, Montbonnot, France
[3] Université de Lyon, Université Claude-Bernard Lyon 1, Inria, CNRS, ENS de Lyon, LIP

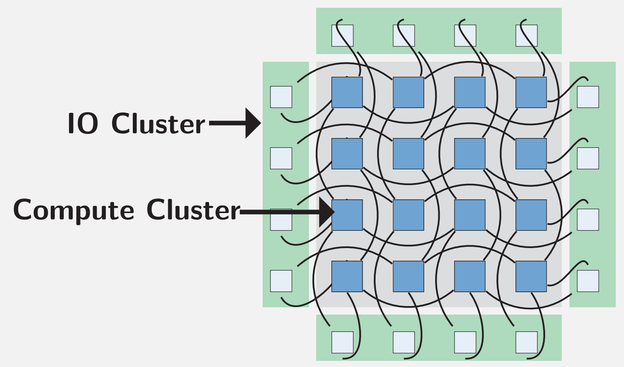
Parallel Code Generation of Dataflow Synchronous Programs for a Many-core Architecture

Parallel Code Generation from Lustre/Scade

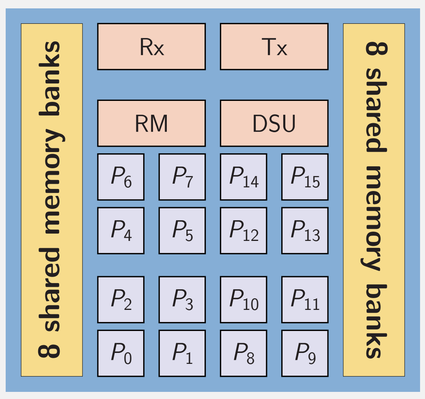


- Parallel execution
- WCET guarantee accounting for memory interference
- Time-triggered execution
- Code Traceability

The Kalray MPPA2 Many-core



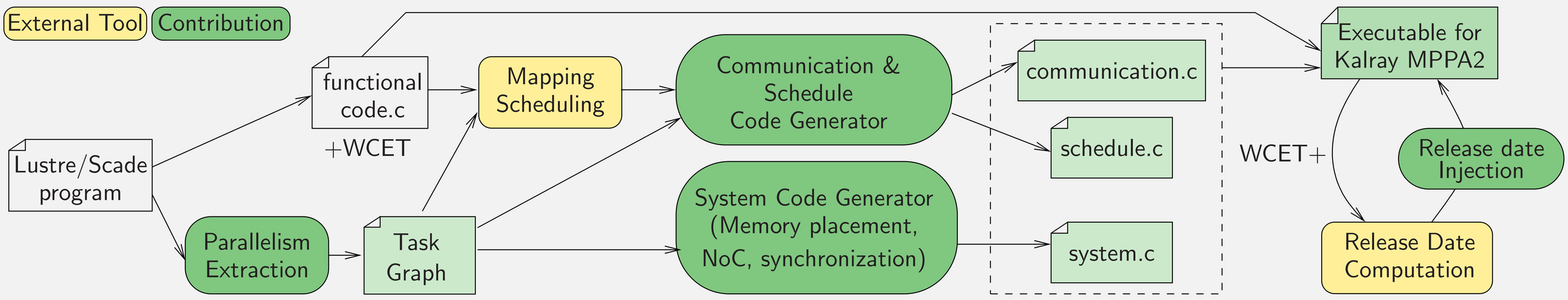
The Kalray MPPA2



Compute Cluster

- Good timing properties of cores
- Banked memory to minimize interferences
- Synchronizable mesochronous clocks on the cores

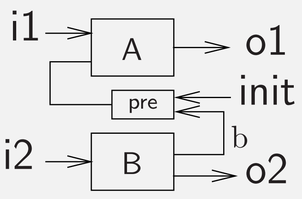
Workflow of the Parallel Code Generator for Kalray (PCGK)

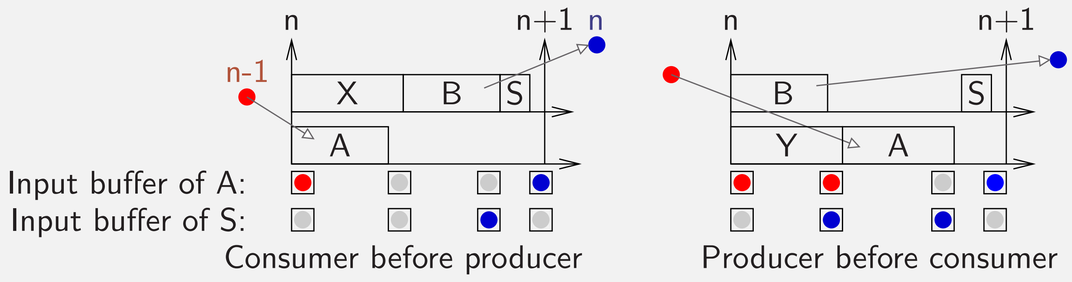


- PCGK generates code for system configuration, shared-memory and NoC communications, synchronizations and task creation
- Schedule is computed with Mapschedule [2]
- Worst-Case Execution Time (WCET) in isolation for each task computed with Otawa [1]
- MIA [3] computes release dates for the tasks taking into account memory interference and dependencies.

Implementation of Delayed Communications

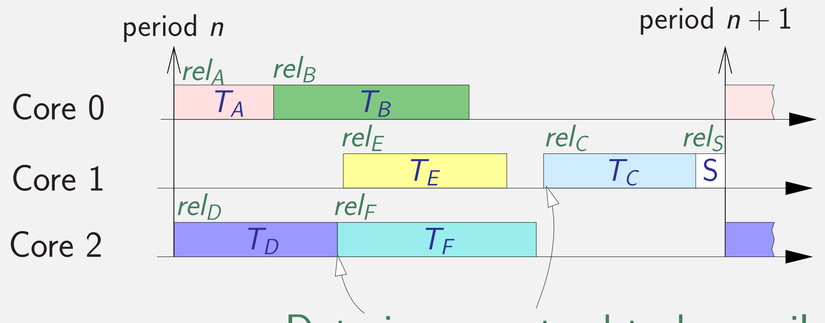
- Delay replaced with a SWAP task: `A.i = S.i`
- S executed after both A and B
- Example when A before B and B before A:





Time-Triggered Execution

- Increases predictability of memory congestion
- Release dates to begin *computations*



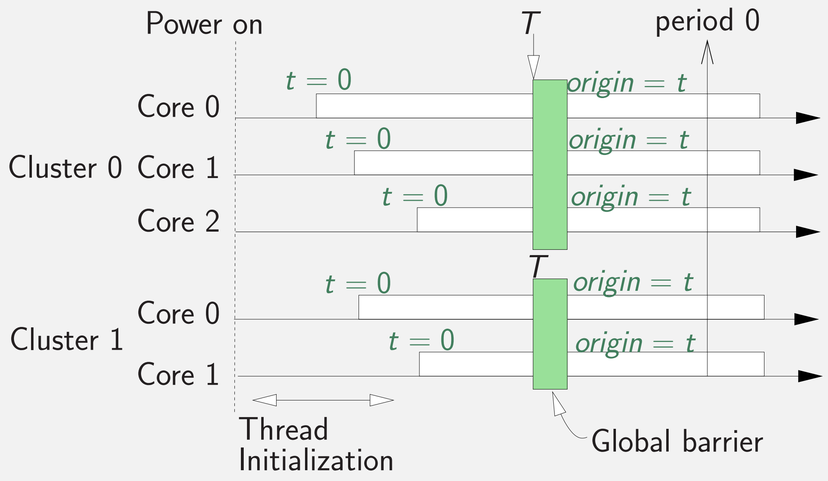
- Example: thread core 0:

```
while(true) {
    while(t<releaseA){};
    T_A();
    send_outputs_A();
    while(t<releaseB){};
    T_B();
    send_outputs_B();
}
```

1 period

Initial Clock Synchronization

- Hardware global timer to synchronize clusters
- Barrier to synchronize cores local timer



[1] Clément Ballabriga, Hugues Cassé, Christine Rochange, and Pascal Sainrat.
Otawa: an open toolbox for adaptive wcet analysis.
In *IFIP, SEUS 2010*, pages 35–46. Springer, 2010.

[2] Viet Anh Nguyen, Damien Hardy, and Isabelle Puaut.
Scheduling of parallel applications on many-core architectures with caches: bridging the gap between WCET analysis and schedulability analysis.
In *JRWRTC 2015*, Lille, France, November 2015.

[3] Hamza Rihani, Matthieu Moy, Claire Maiza, Robert I Davis, and Sebastian Altmeyer.
Response time analysis of synchronous data flow programs on a many-core processor.
In *RTNS'16*, pages 67–76. ACM, 2016.

amaury.graillat@univ-grenoble-alpes.fr