# Communication Refinement for SOC Design

Hocine Mokrani,
Rabéa Ameur-Boulifa, Sophie Coudert, Emmanuelle Encrenaz-Tephène

LABSOC (Telecom ParisTech)
hmokrani@telecom-paristech.fr

SAFA-Sophia Antipolis 2010
07-10-2010

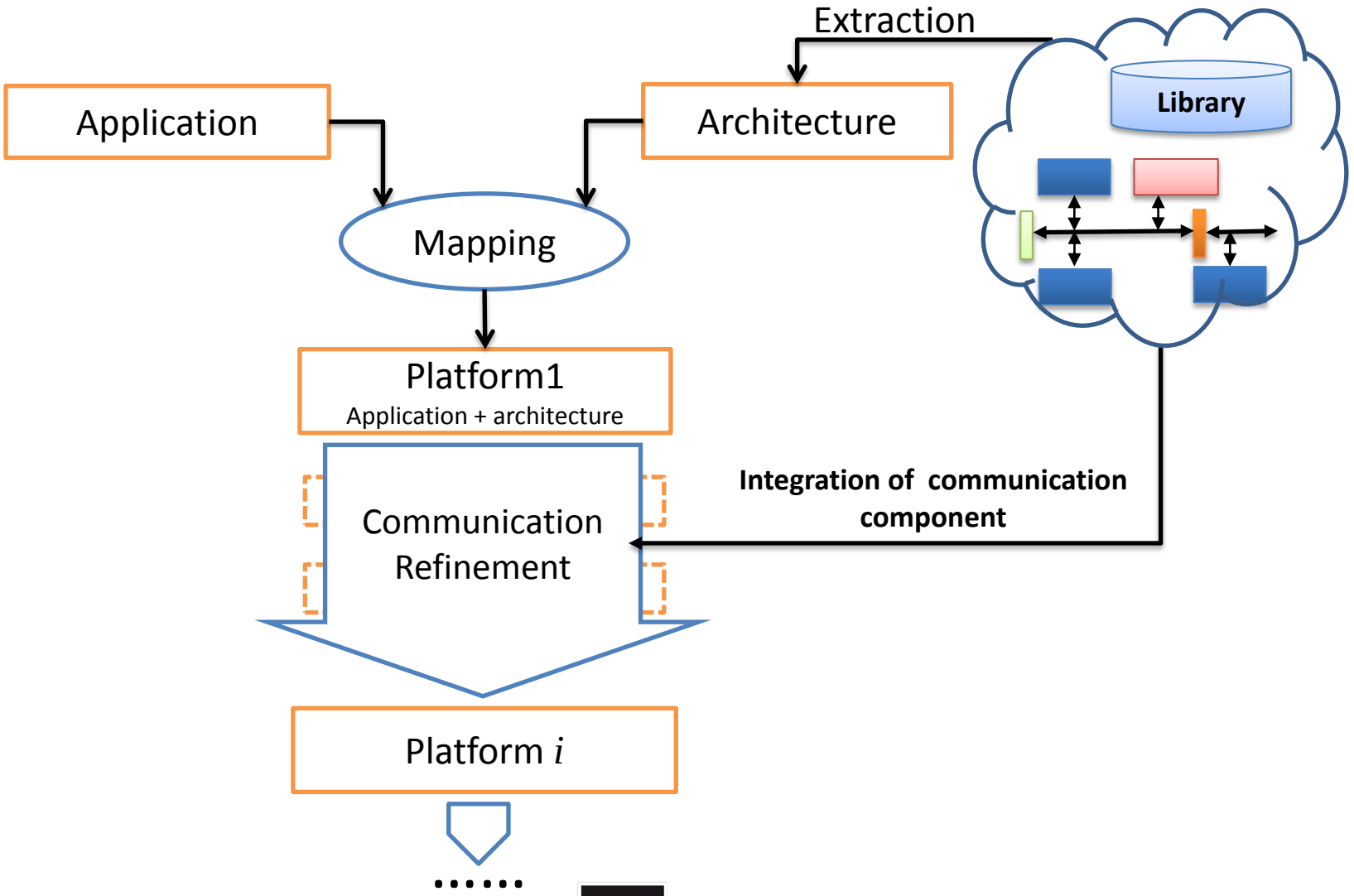
# Objective

Our work aims to improve the architecture exploration phase by incremental development of SoCs and the use of formal verification techniques.

# Plan

- Design methodology.
- Our approach.
- Case study.
- Conclusion.

# Design Methodology

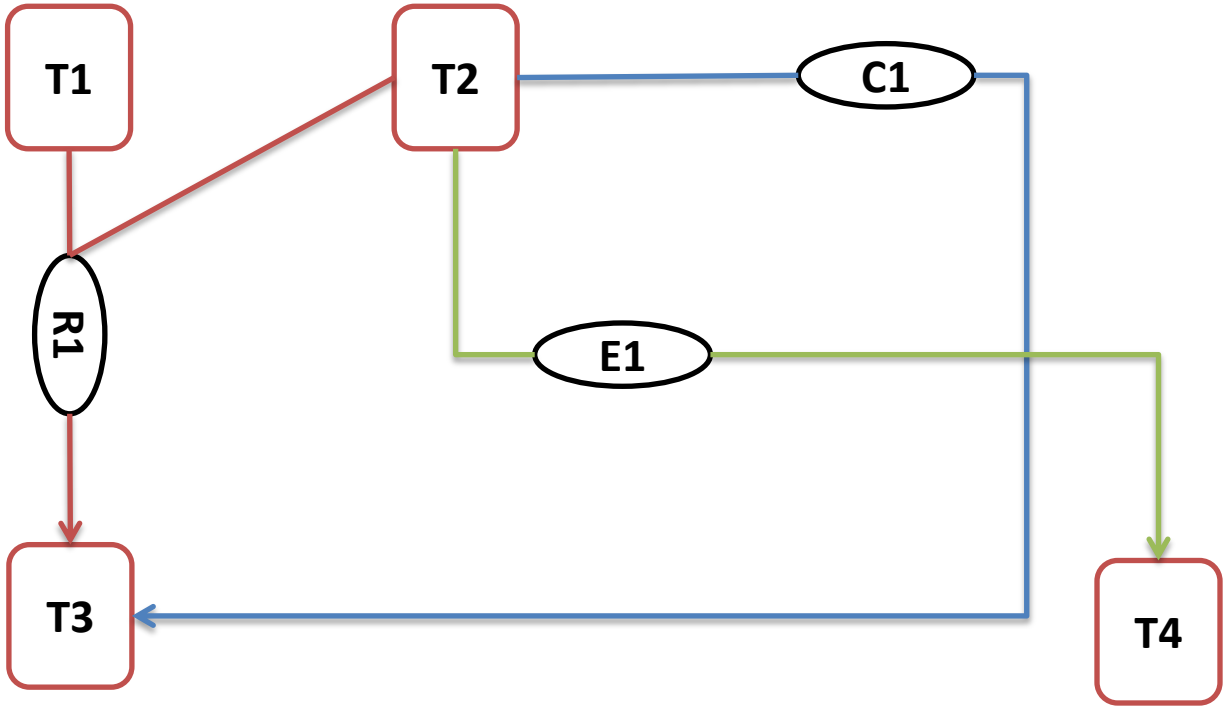


# Application

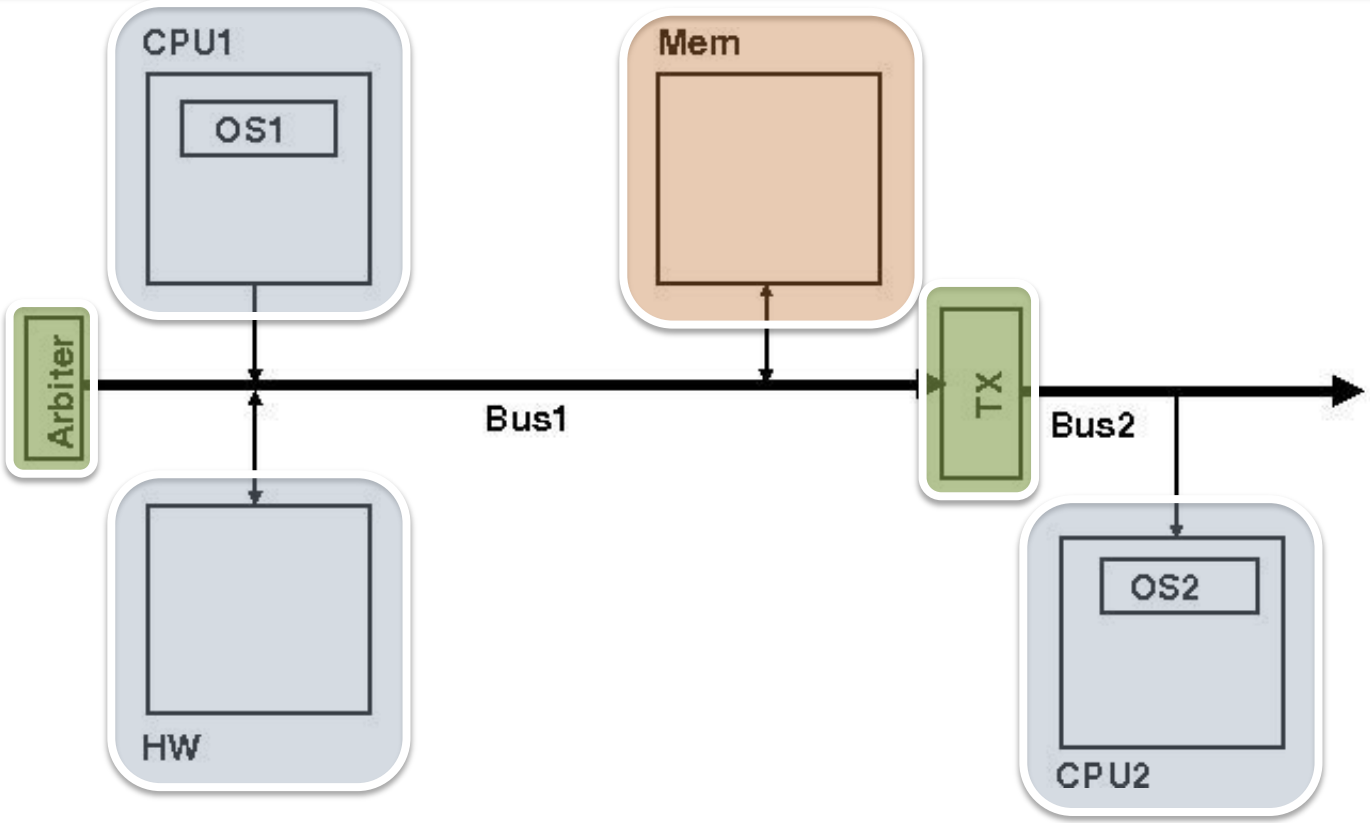
- Separation of application and architecture.
- Data abstraction.
- TML Modeling the system by a tasks network communicating via channels.
  - ***Task***: "Calculus/communication" Instructions, variable Setting, Tests and loops.
  - ***Channels:***
    - Channels: BR-NBW, NBR-NBW, BR-BW
    - Events: FIFO fini, FIFO infini.
    - Requests: FIFO infinie.

[1] W.Muhammad and al. *Abstract application modeling for system design space exploration*. August 2006.

# Application (Cont.)



# Architecture




**PE** (Processing Element)

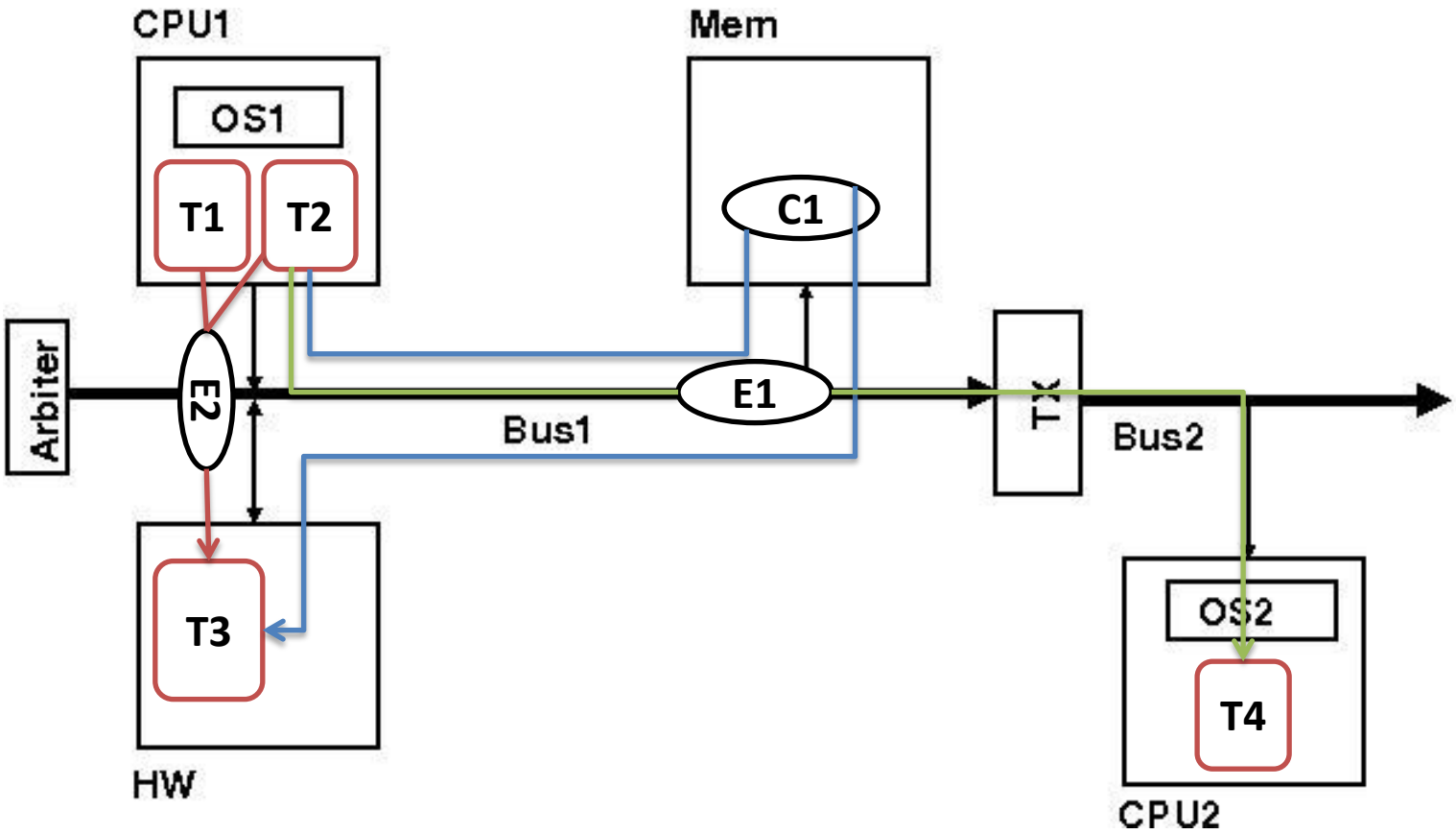
**ME** (Memory element)

**IF** (Interface Element )

**CE** (Communication element)

# Mapping

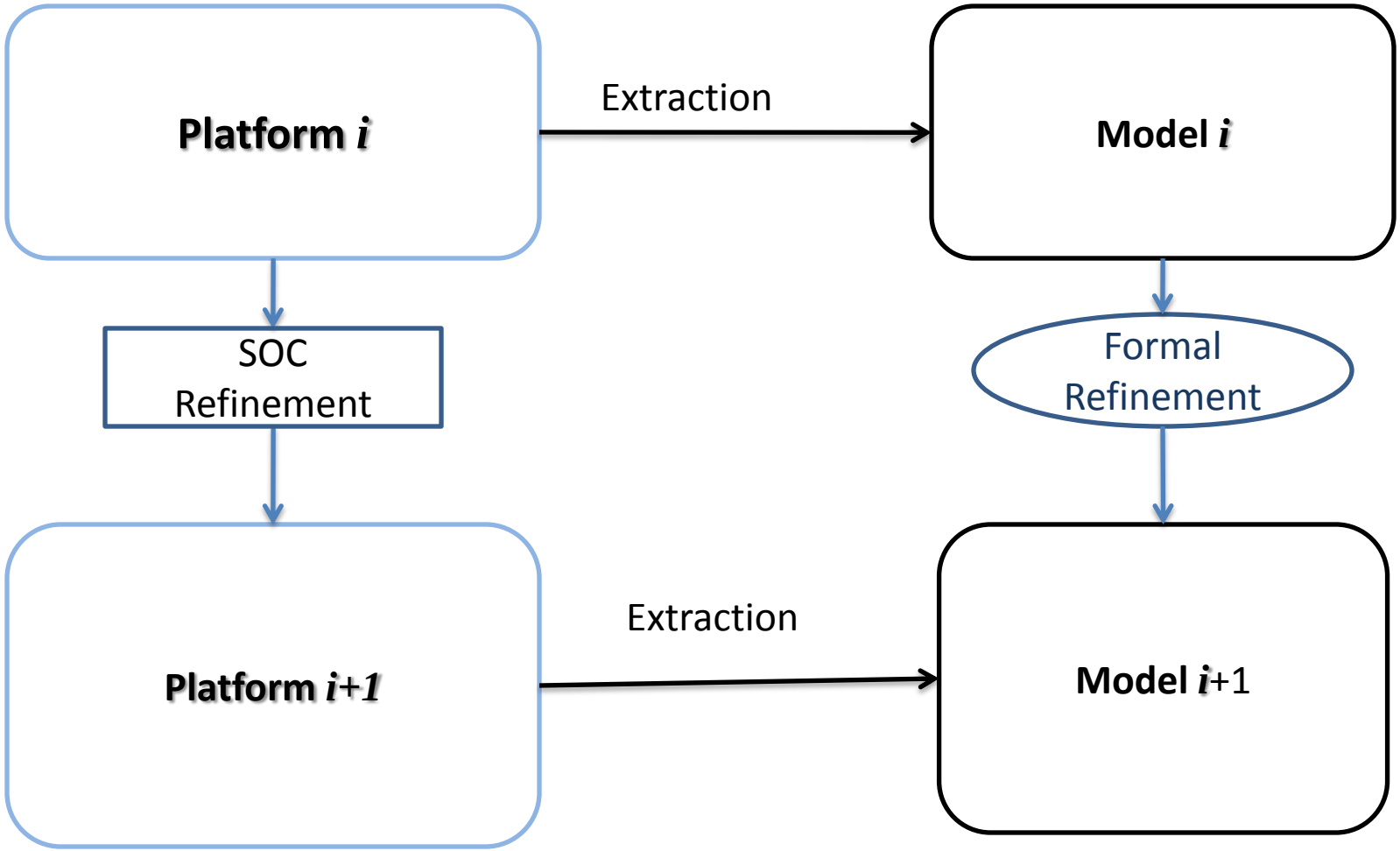
Platform = Architecture ⊕ Application



# SOCs Refinement

- Software Refinement.
  - [2] Sebastian Ritz and al. *High-Level Software Synthesis for the Design of Communication Systems.*1993.
- Hardware Refinement.
  - [3]F.Balarin and al. *Hardware-software co-design of embedded system : the polis approach*. 1997
- Communication Refinement.
  - [5] Paul Lieverse and al, *A trace transformation technique for communication refinement.*2001.

# Formalization



# Formal refinement

- Formal Refinement ⇒ Property preservation.
- Formalisms and tools:
  - B method.
    [6] J.R. Abrial. *The B-book*, 1996.
  - Z language.
    [7] G. Smith. *The Object-Z Specification Language*. 2000.
  - LTS refinement (Simulation, Bisimulation, …).
    [8] M.B. Josephs, *A stat-based approach to communicating processes*.1988.
  - Process Algebra Refinement (Trace, Failure, readness,…).
    [9] J.R van Glabbeek, *The linear time-branching time spectrum I*.
- Property preservation⇒ Refinement Semantics.

# Models

**Platform**

Collection of interconnecting Processes

**Automata Network**

**Property Guarantee**

[8] M.B. Josephs, *A state-based approach to communicating processes.* 1988.

# Case Study

**CHANNEL C1**, BRBW, 1,Task1, Task2

```
TASK Task1
    |
    |       WHILE (1)
    |         | EXEC
    |         | WRITE C1
    |       ENDWHILE
ENDTASK

TASK Task2
    |
    |       WHILE(1)
    |         | READ C1
    |         | EXEC
    |       ENDWHILE
ENDTASK
```

# Application Model Construction

**CHANNEL C1**, BRBW, 1,Task1, Task2

TASK **Task1**

WHILE (1)

**EXEC**

**WRITE** C1

ENDWHILE

ENDTASK

TASK **Task2**

WHILE(1)

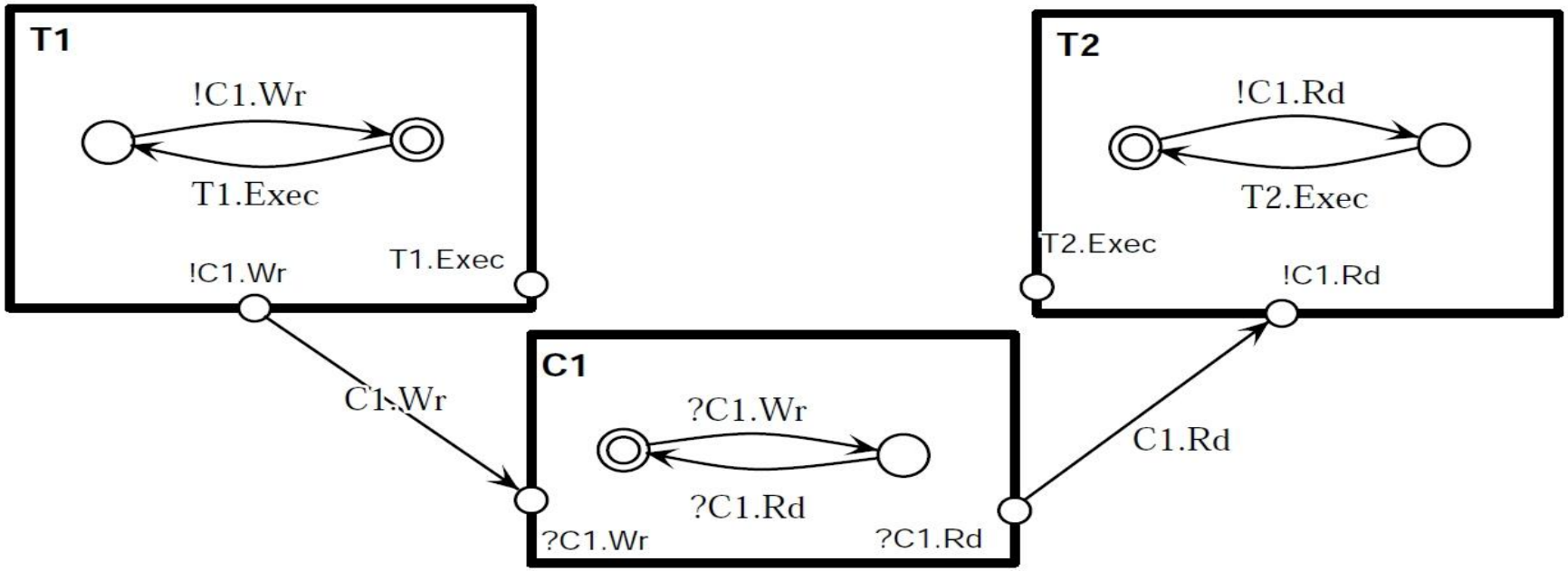
**READ** C1

**EXEC**

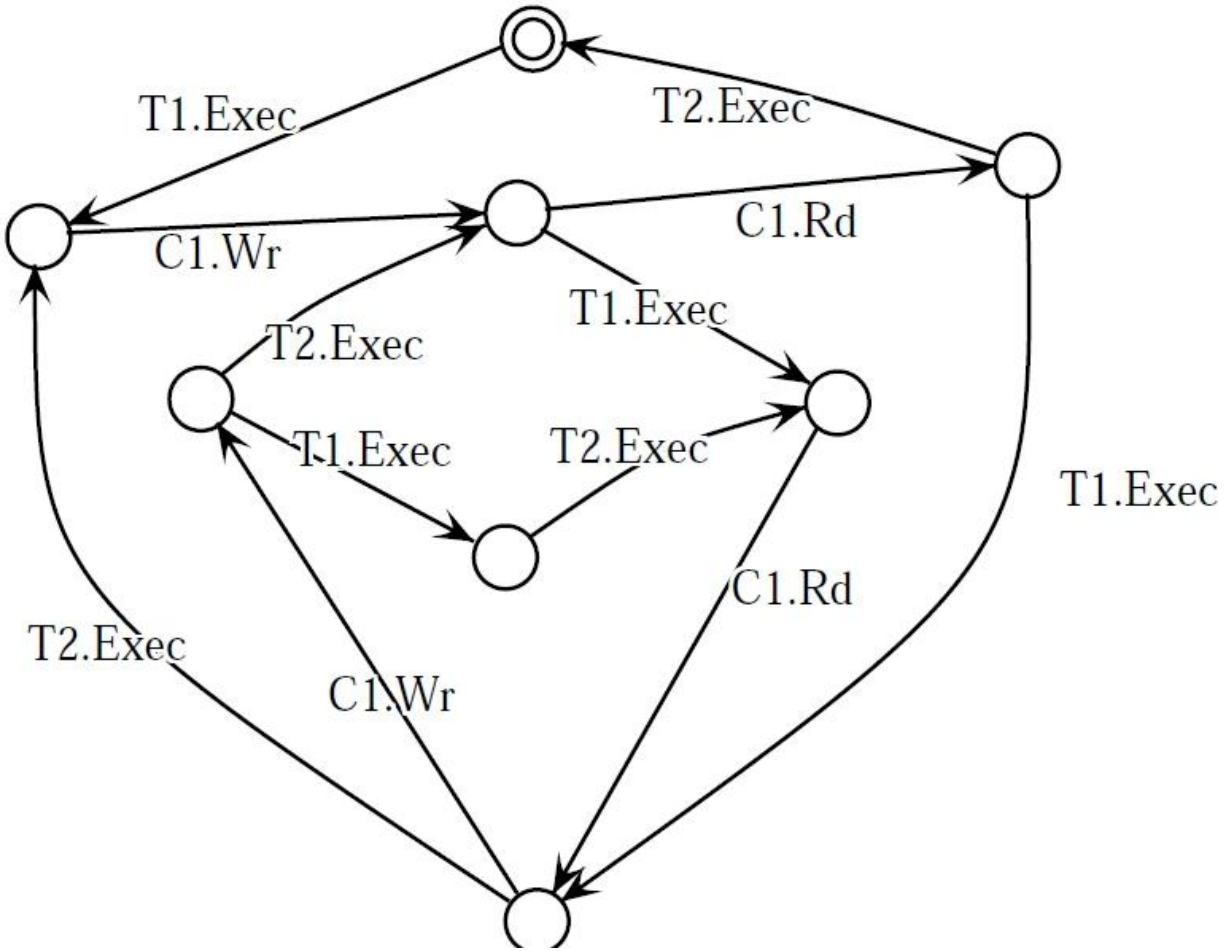
ENDWHILE

ENDTASK

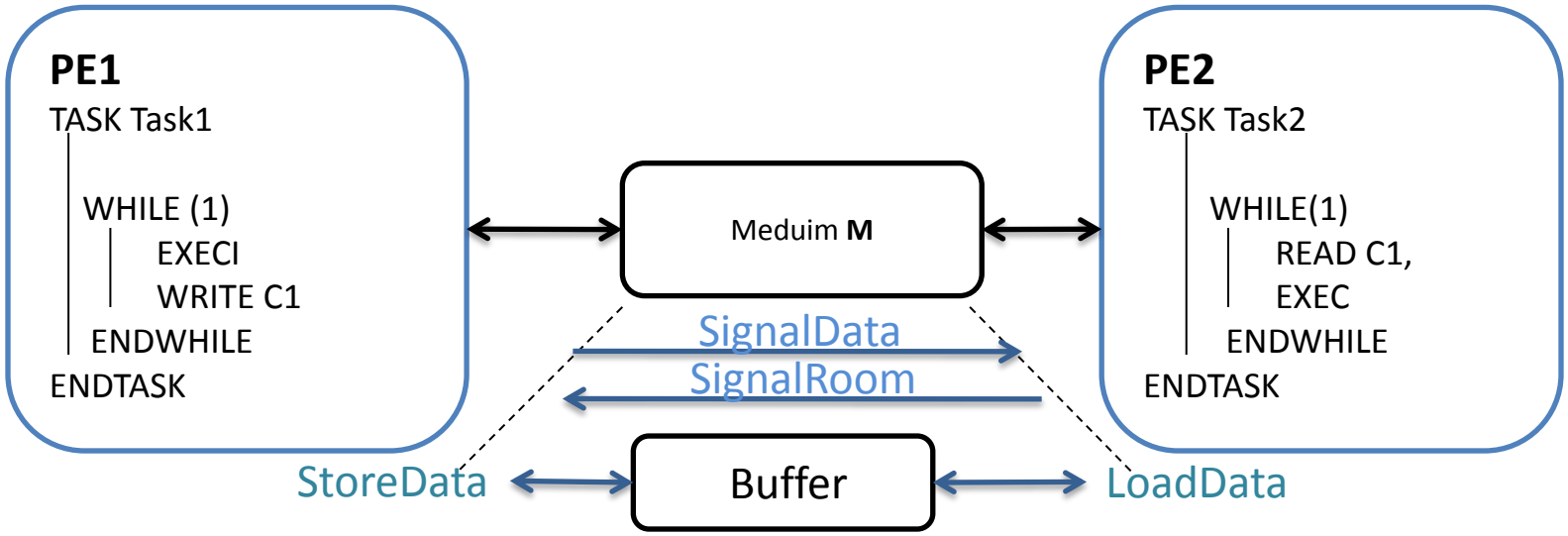
# Application Case Study Construction



# Global Application Model



# Platform2 Model

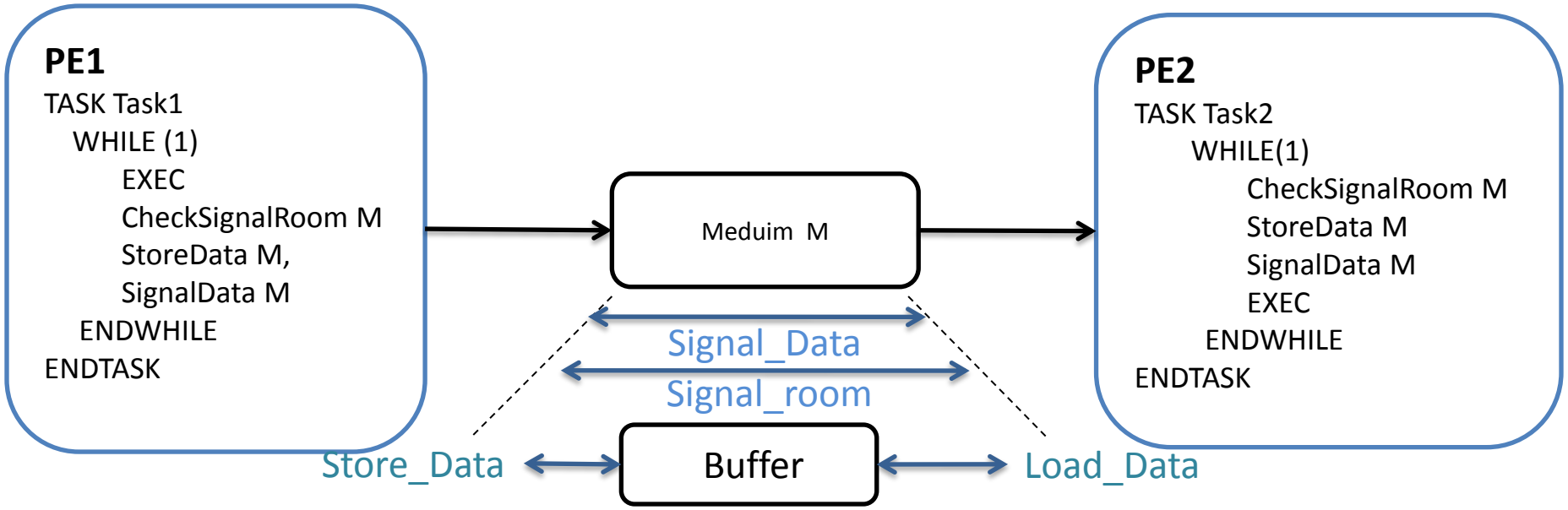



- **Write** ≙ CheckSignalRoom; StoreData; SignalData.
- **Read** ≙ CheckSignalData; LoadData; SignalRoom.
- Transformation:

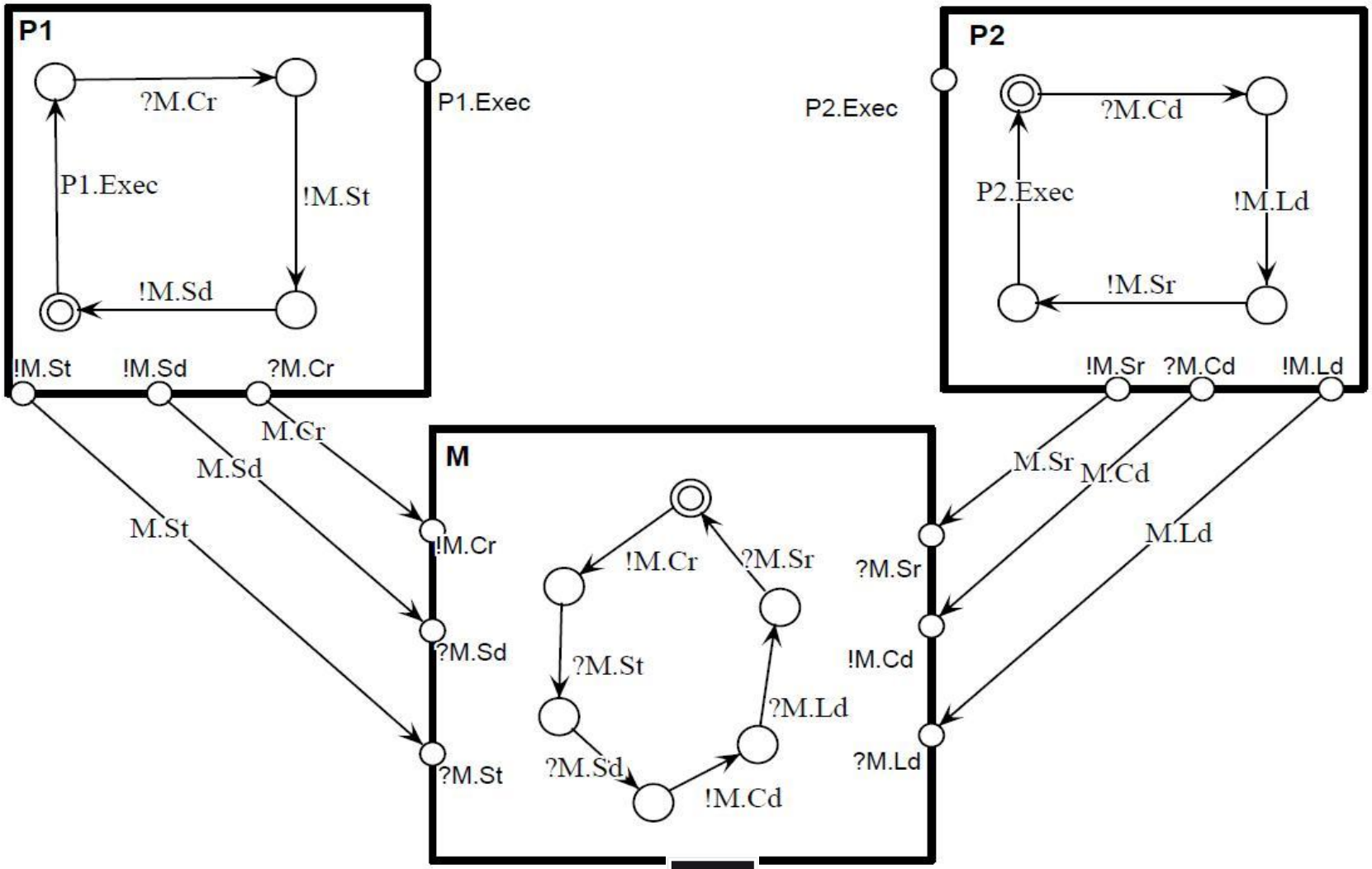
**Read; Exec** ≙ CheckSignalData; LoadData; SignalRoom

? Exec ? Exec ? Exec

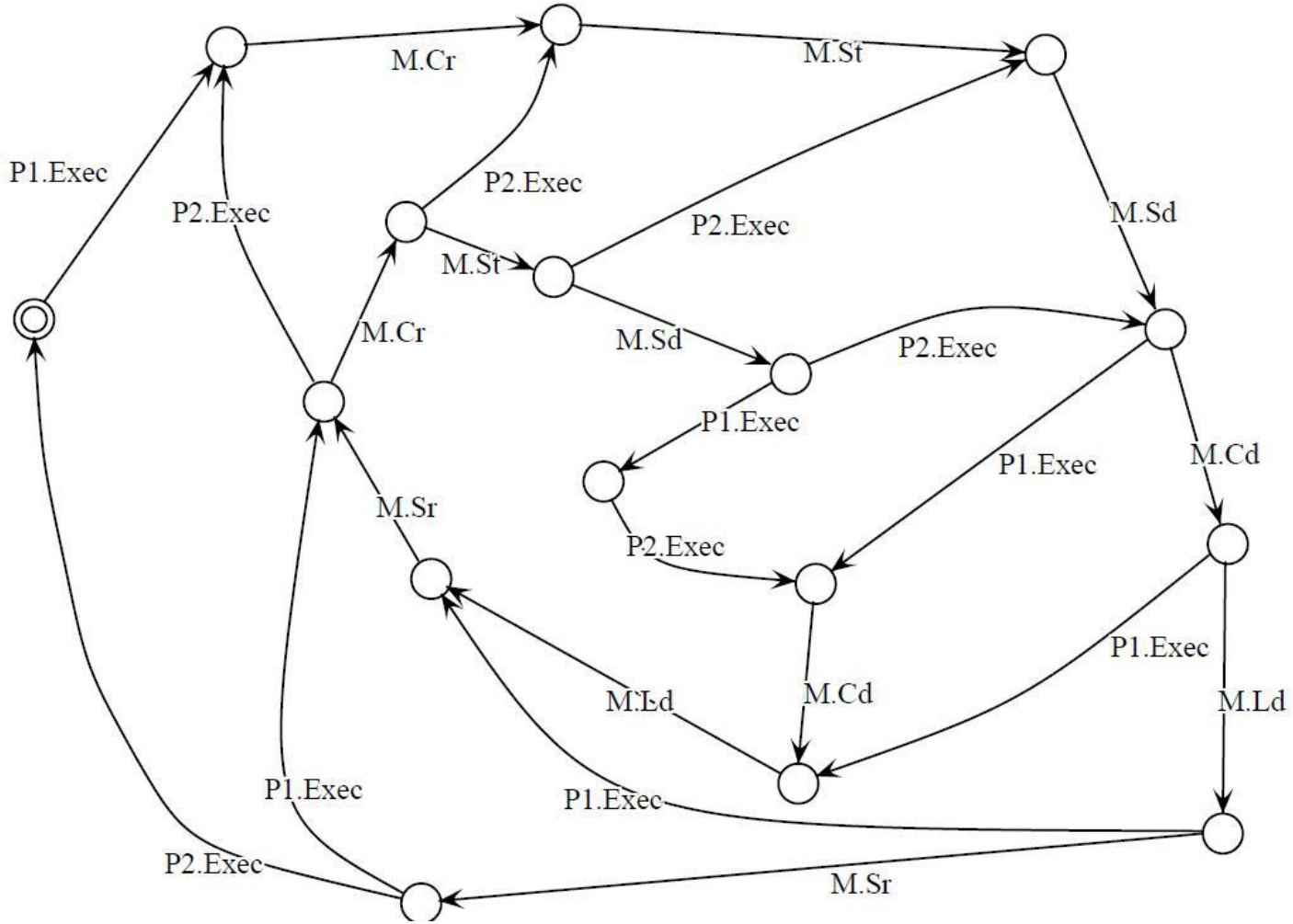
# Platform Model Construction



# Platform Model Construction

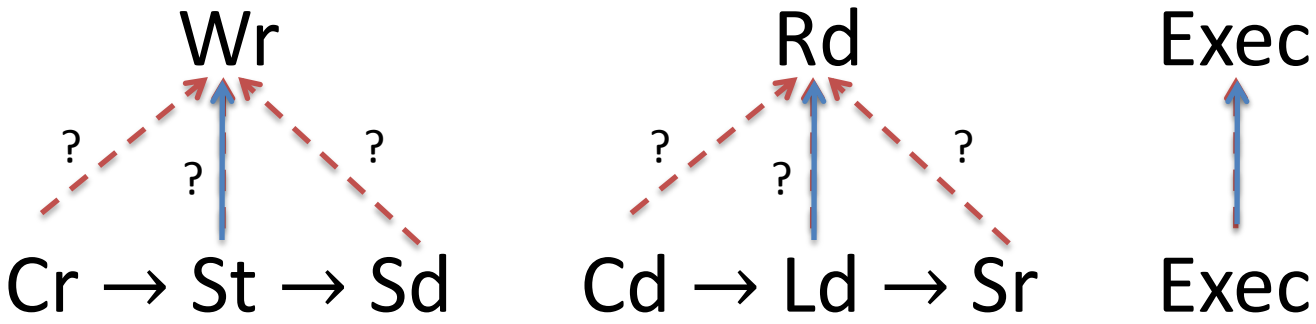


# Global Platform Model



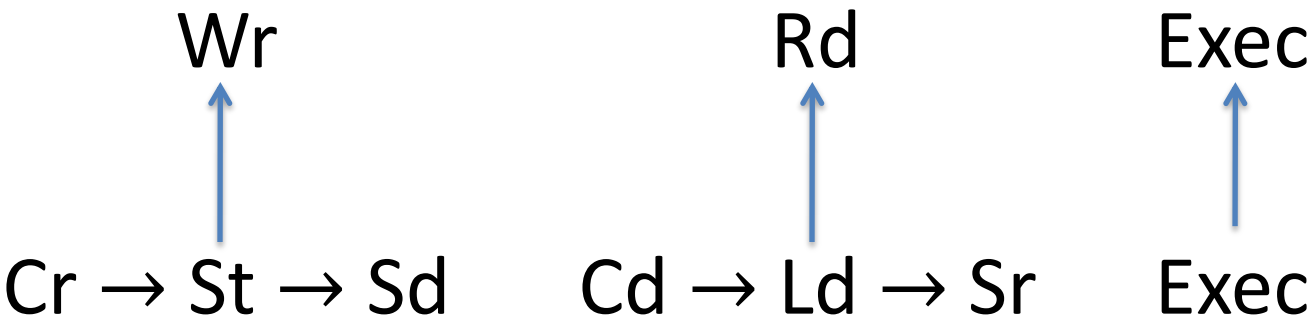
# Refinement Study

- To apply refinement[8], we need:
  - to give a correspondence between all Application actions and platform actions.
  - hide the rest  of platform actions .

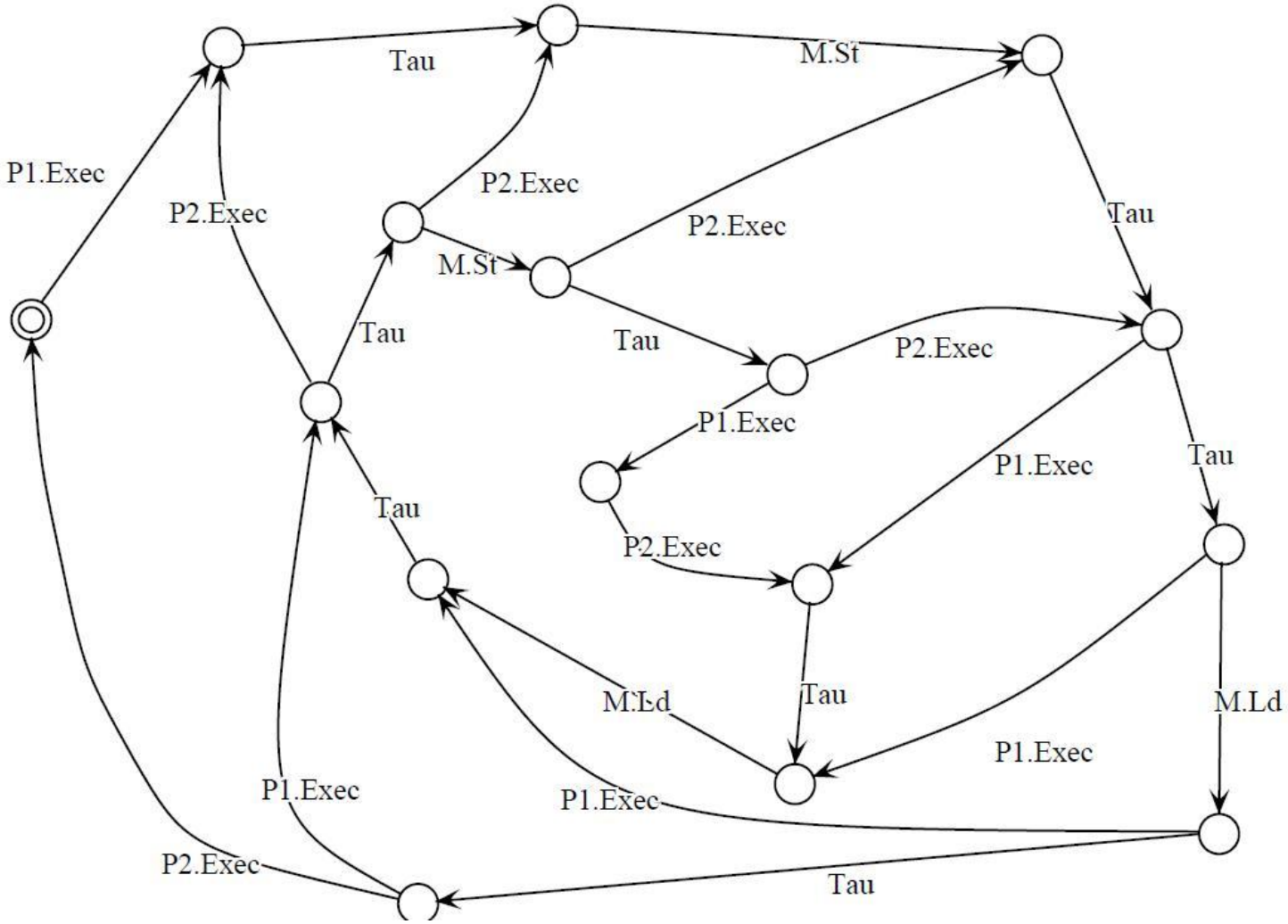


# Refinement Study

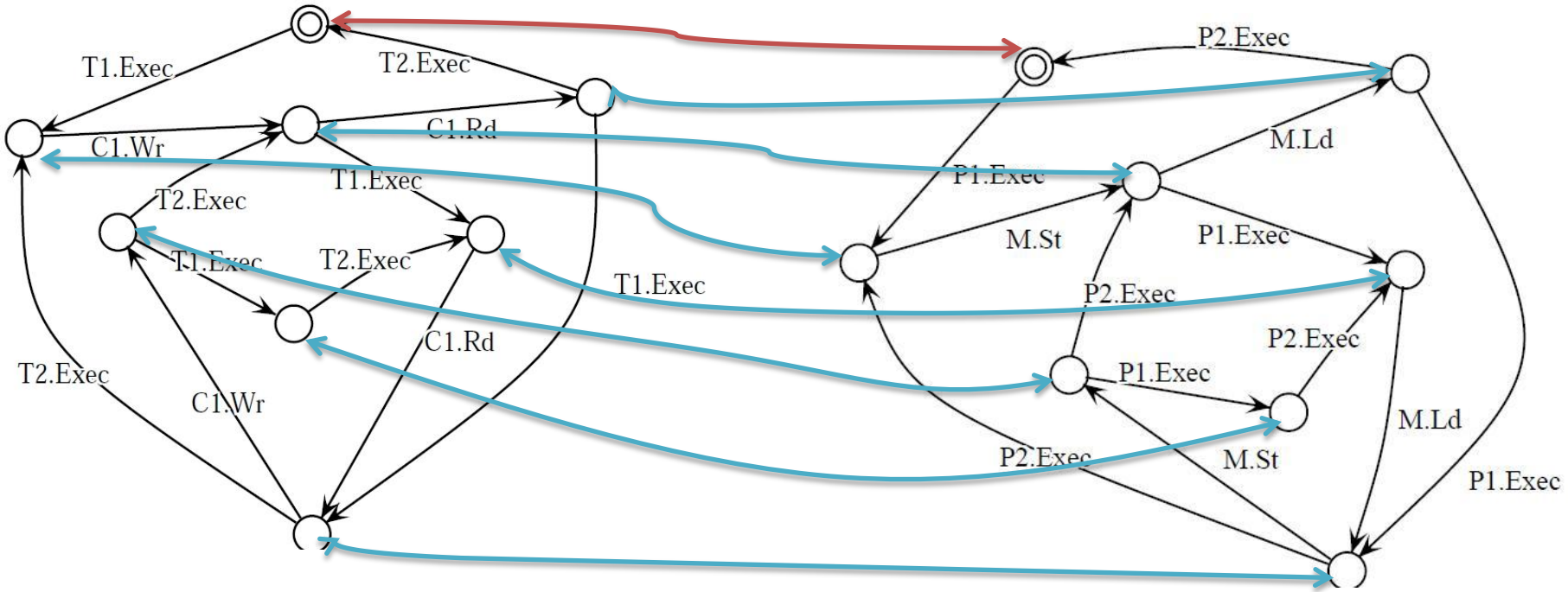
- To apply refinement[8], we need:
  - to give a correspondence between all Application actions and platform actions.
  - hide the rest  of platform actions .



# Global Platform Model (Tau)



# Downward Simulation




- **D relation such that:**
  - Every initial state of Process ***i+1*** must correspond to a initial state of Process*i*.
  - If the Processes are in corresponding states, they must be able to engage in the same events.
  - If the processes are in corresponding states and P2 can engage in e, P1 must be able to engage in e in such a way that the processes remain in corresponding states.

# Conclusion and future work

Integration of formal refinement in system-level design methodology.

- Currently
  - Formalizing the presented concept (Application, Architecture, Mapping) => Choose the accurate abstraction.
  - Study of real bus Refinement. (AMBA bus)
- Future
  - Extraction function.
  - How about the other components Refinement.

# Thank you