Multi-Level Modeling and Simulation of Cognitive Radio Equipments

Stéphane LECOMTE[1,2], Christophe MOY[1] et Pierre LERAY[1]

[1]Supélec/IETR, SCEE Team, Rennes, France
[2]Technicolor, Research & Innovation, Rennes, France
stephane.lecomte@supelec.fr

Outline

- Context & challenges
- Problem & Solution
- Objectives
- Proposed MDA methodology for co-design
- Extension for reconfigurable systems
- Uses case: Cognitive Radio system
- Conclusion

Context & challenges

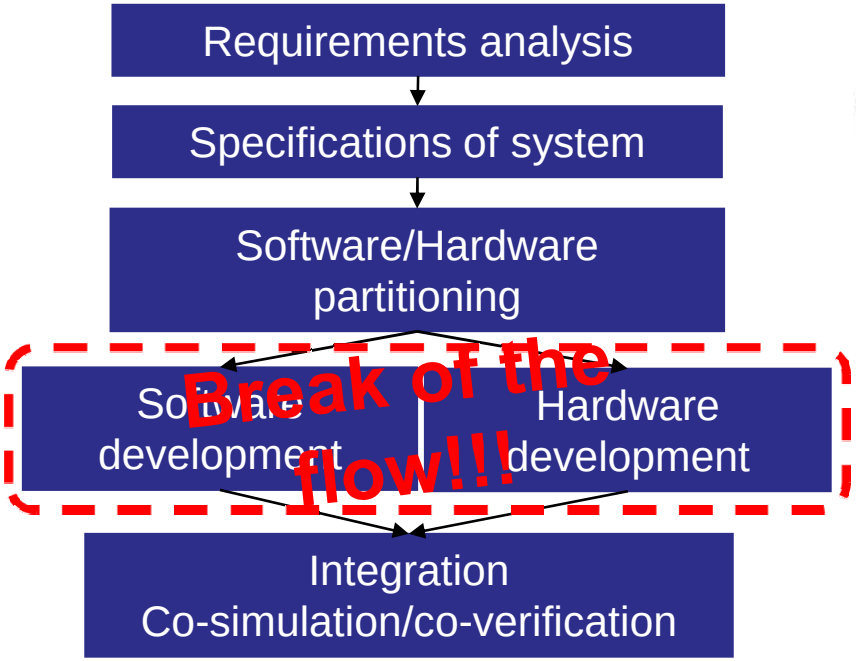
Design of radio communications systems

- **Complexity of systems ↗**
- **Heterogeneous systems**
- **Embedded systems → real-time constraints**
  - Memory resource limitation,
  - Computing resource limitation,
  - Power consumption,
  - Etc.
- **Technology capacity ↗** (Moore's law)
  - System can be integrated in single chip
- **Time-to-Market ↘**

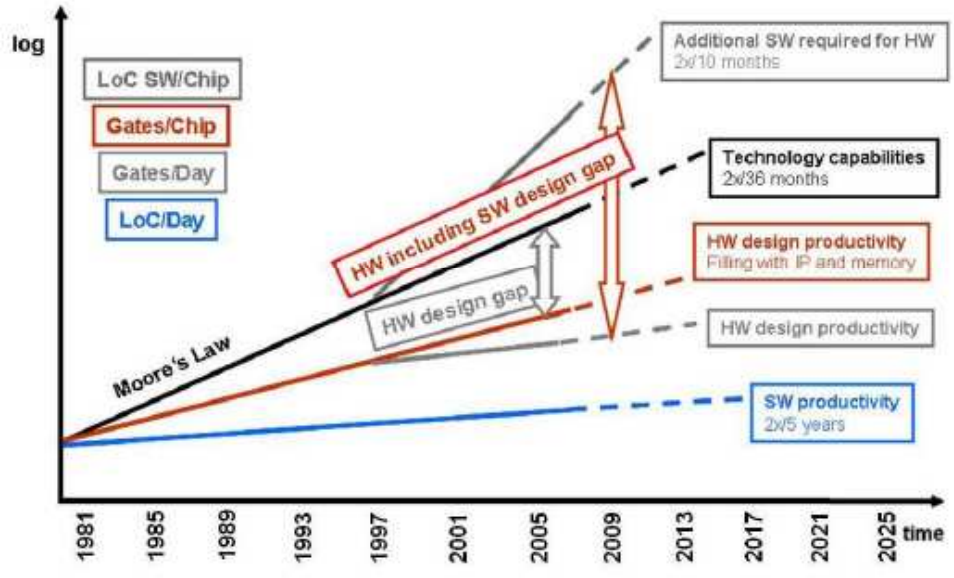
Reconfigurability is a key enabler → FPGA

Problem & Solution

- **Co-design flow**
  - HW/SW partitioning



- **Co-design methodology**
  - do not progress as quickly as the technology



Source: ITRS report 2009

➔ Use a high level approach in a new co-design methodology, integrating reconfigurable capabilities

Objectives

- **Formalizing a new development process based on high level models for Cognitive Radio co-design (extension to SoC/SoPC)**
  - Covers Electronic System Level (ESL) domain
  - Use **UML** models
  - Use **MARTE** profile from OMG, extension of UML
  - Use **Model Driven Architecture** (MDA) approach
  - Automatic **code generation**
  - Generation of documentation
  - Top-down iterative process
- **This presentation**

Focus on reconfigurable systems

- Integration of dynamic and partial reconfiguration of FPGA technology (reconfigurable hardware for SoPC)

MDA methodology for co-design

Three levels of modeling

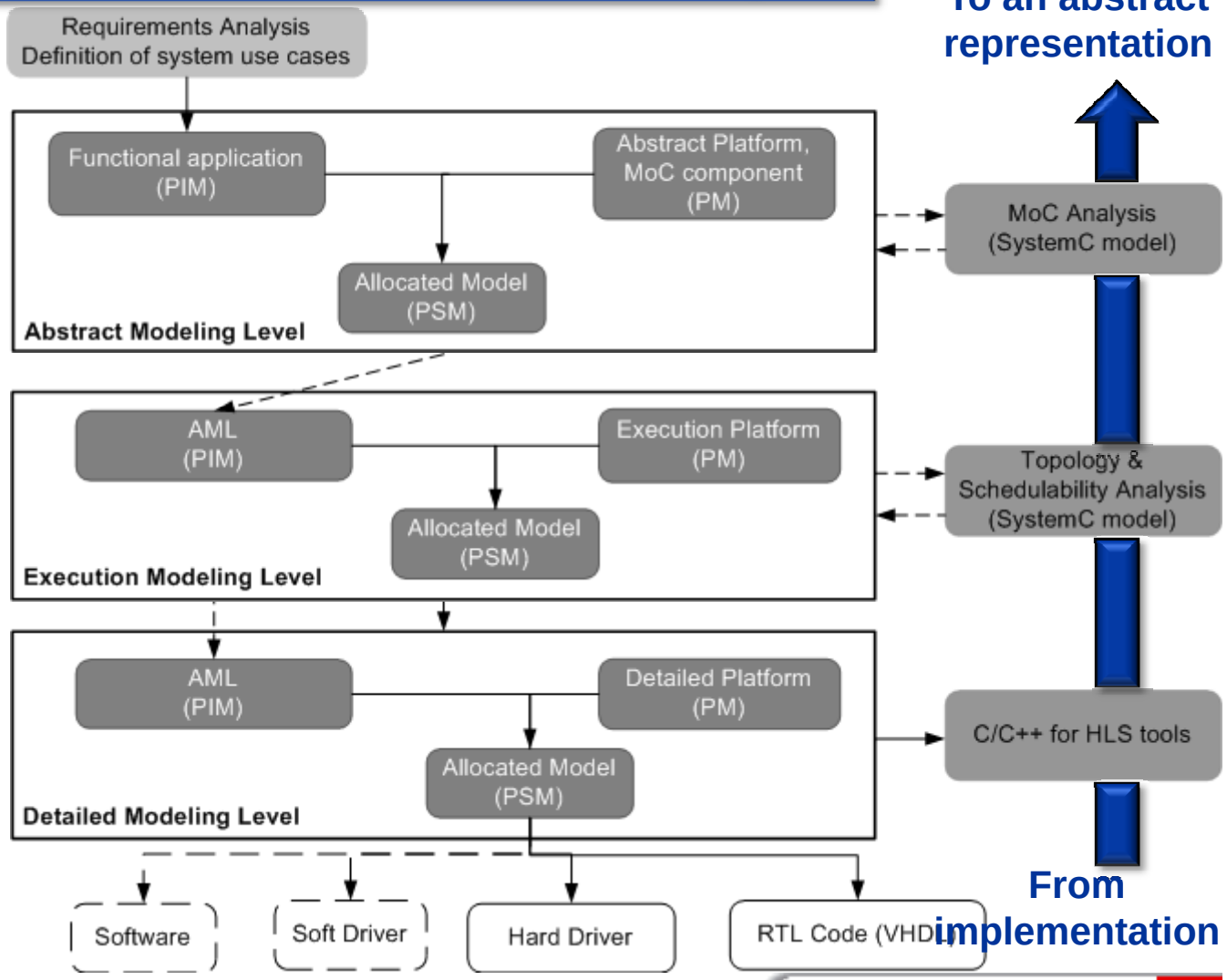
From specification

- Modeling of high level of abstraction
- Validation of functional architecture and behavior

- Modeling the topology of hardware platform
- Dedicated to architecture exploration

- Detailed modeling hardware platform
- Enable to VHDL code generation

To implementation



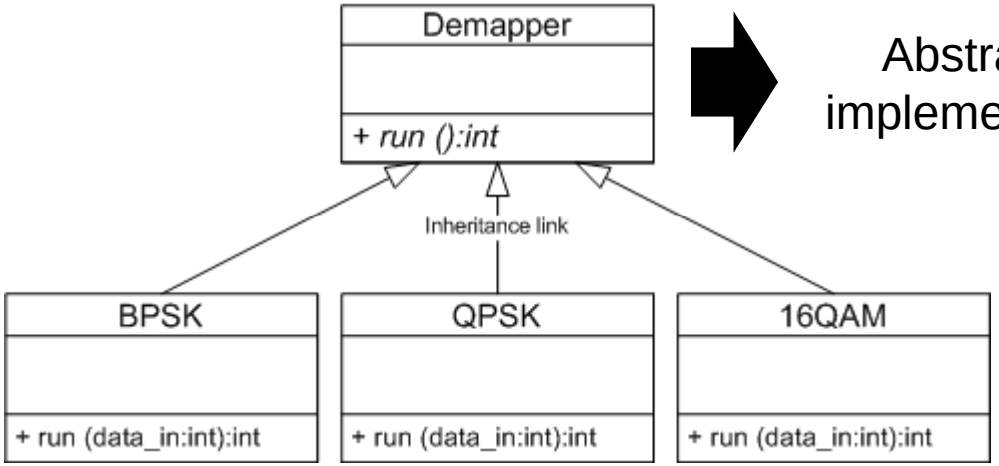
Outline

- Context & challenges
- Problem & Solution
- Objectives of works
- Proposed MDA methodology for co-design
- **Extension for reconfigurable systems**
- Uses case : Cognitive Radio system
- Conclusion

How to model reconfiguration?
In functional model (PIM)

How to express reconfigurability in UML?

→ Polymorphism (object-oriented programming)



Abstract class without implementation of methods

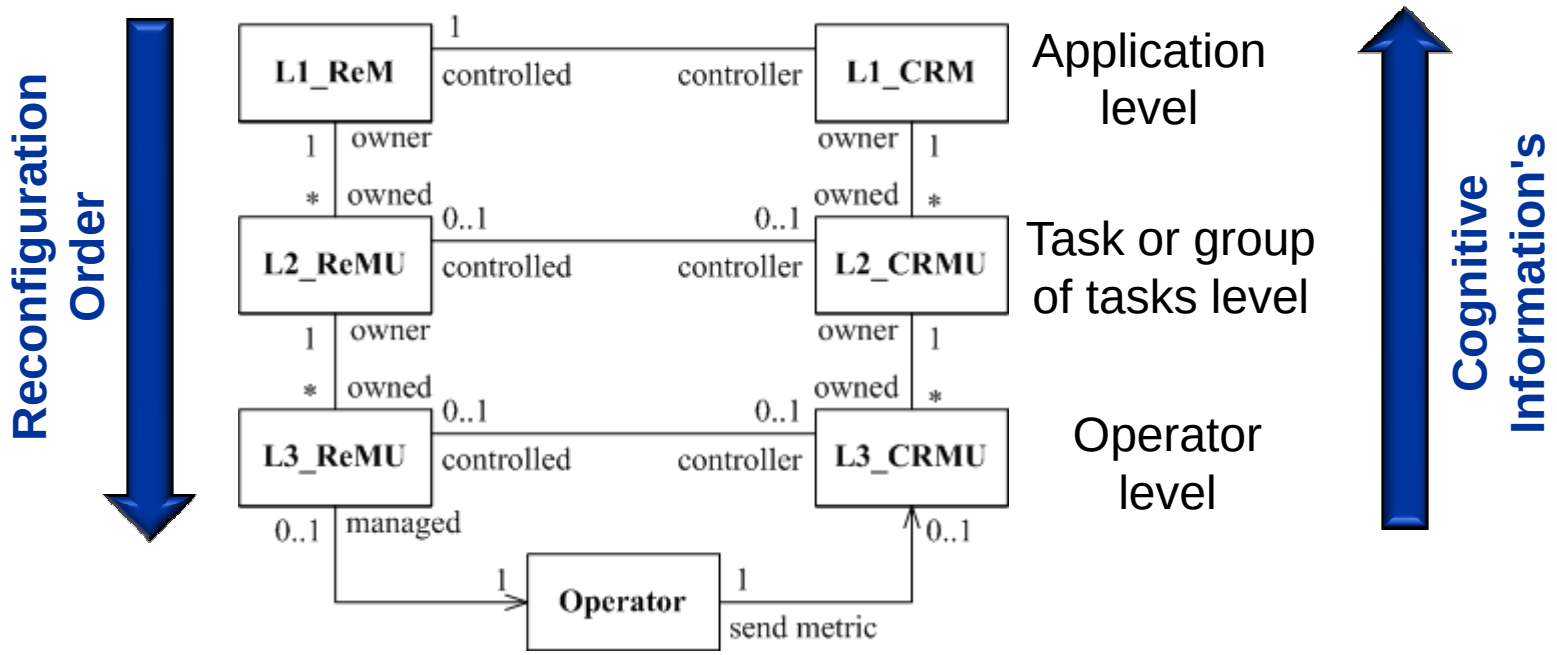
Actives classes with different implementations of the method run
Only one is instantiated at a moment

→ But **insufficient** to model reconfigurable application

→ How to choose the good implementation according to the context?

Need to add a dedicated architecture to manage reconfiguration and cognitive cycle

– HDCRAM[1] (Hierarchical and Distributed Cognitive Radio Architecture Management)



[1] Loïg GODARD, Christophe MOY, Jacques PALICOT , **"An Executable Meta-Model of a Hierarchical and Distributed Architecture Management for the Design of Cognitive Radio Equipments" ,** *Annals of Telecommunications, Special issue on Cognitive Radio,* Springer Editions, vol. 64, number 7-8, pp. 463-482, Aug. 2009.

How to model reconfiguration?
In platform model (PM)

- **No impact on the highest and middle levels of modeling (AML and EML)**
  - → The type of the computing unit is not specified

- **Impact on the lowest level of modeling: DML**

  Especially for the dynamic and partial reconfiguration of FPGA technology

  - → Partial reconfiguration of FPGA requires some logistics addings
    - Processor
    - HwICAP
    - ICAP Controller
  - → External memory of FPGA to store configurations bitstream

- **Ideal: the design flow automatically inserts these elements as soon as a PR operator is integrated in the design**

How to simulate the reconfiguration?

- **Model the reconfiguration, what for?**
  - Observe/anticipate its impact on the system
    - Reconfiguration time duration, list of files, exchanges, etc.

  ➔ Evaluate reconfiguration overhead

- **How to simulate the model?**
  - Native UML language can not be simulated

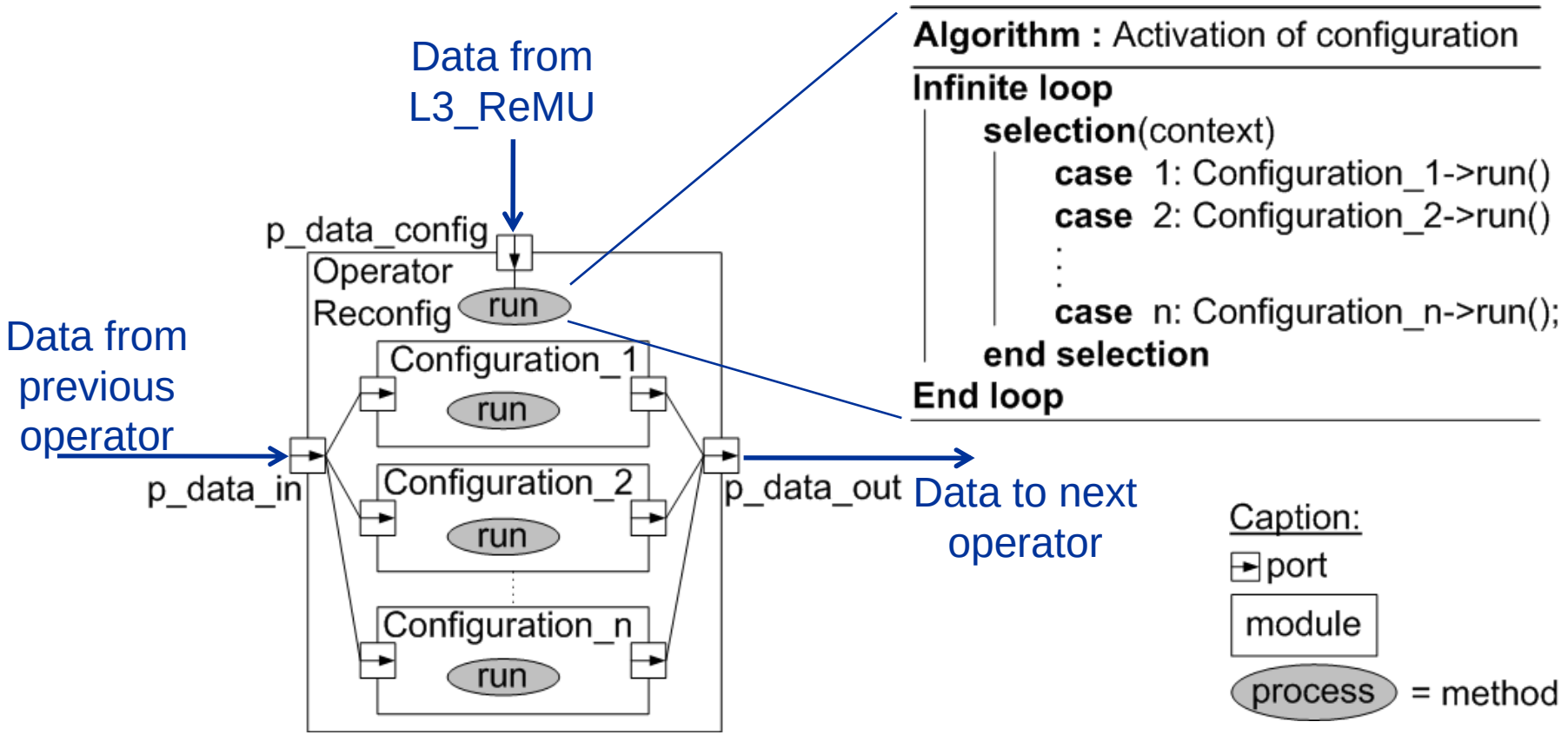
  ➔ Solution: Translate model to SystemC environment

- **SystemC model**
  - Based on C++
  - Modeling both SW and HW at different levels of abstraction

  →But a module (object in C++) can not be destroyed during the simulation step

How to simulate the reconfiguration?

- **A generic module for reconfigurable operator**



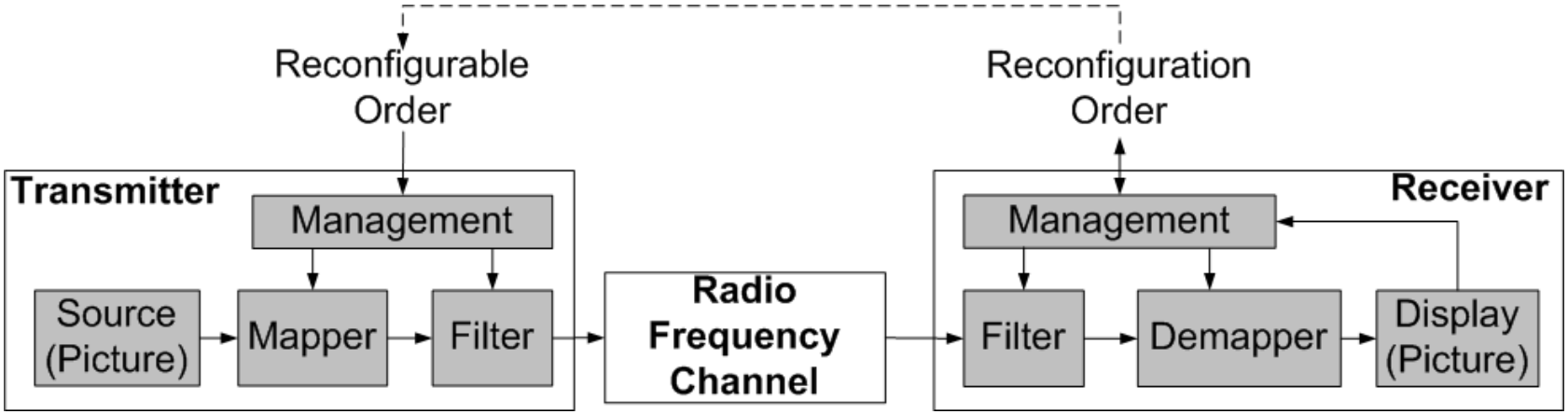
→ Only one configuration is active at a moment according to the context
→ Information of context can be done by the L3_ReMU associated module

Outline

- Context & challenges
- Problem & Solution
- Objectives of works
- Proposed MDA methodology for co-design
- Extension for reconfigurable systems
- **Uses case: Cognitive Radio system**
- Conclusion

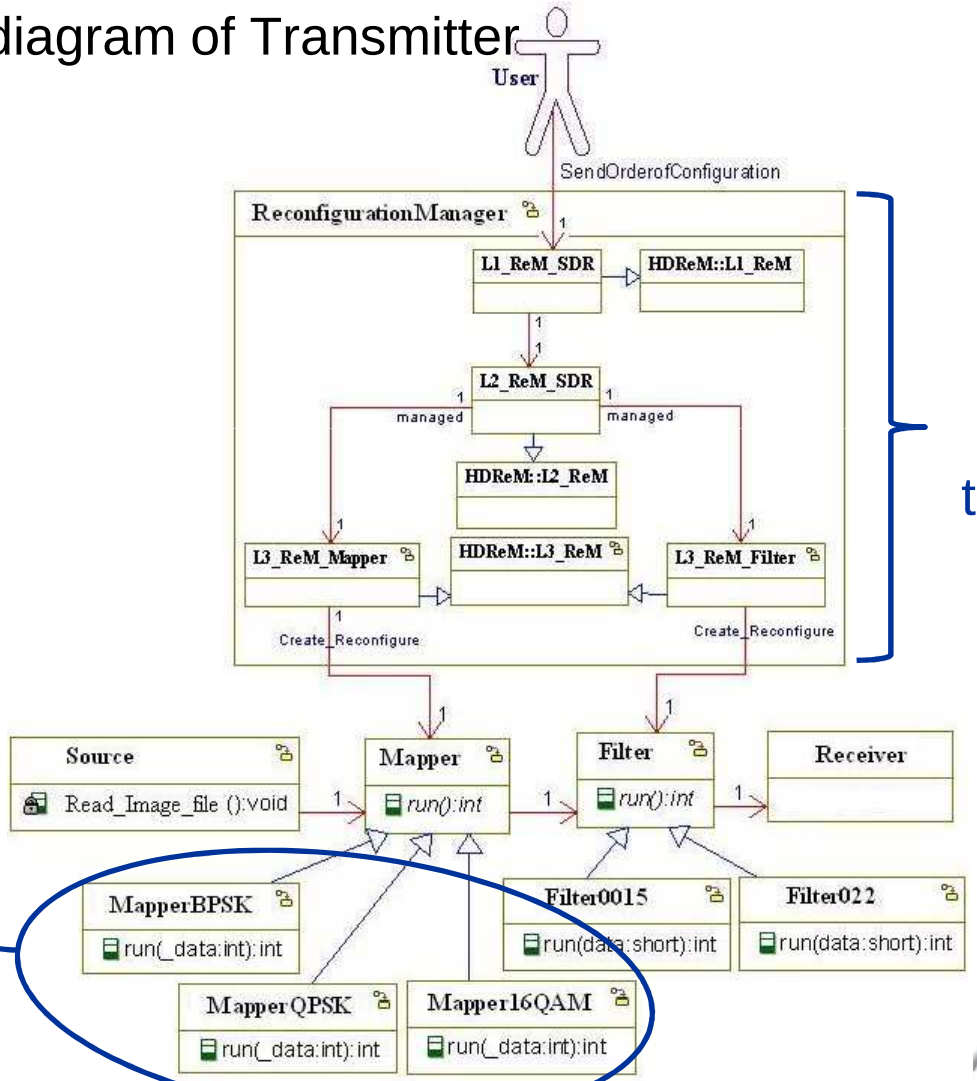
A simple Cognitive Radio system

- **CR proof-of-concept scenario**
  - auto-adaptation in function of SNR
- **Simplified functional architecture of the system**
  - Mapper/Demapper: 3 configurations
    - BPSK, QPSK or 16QAM
  - Filter: 2 sets of coefficients
    - Roll-off 0,22 or 0,015



- **The hardware target for implementation is**
  - a ML506 board from Xilinx

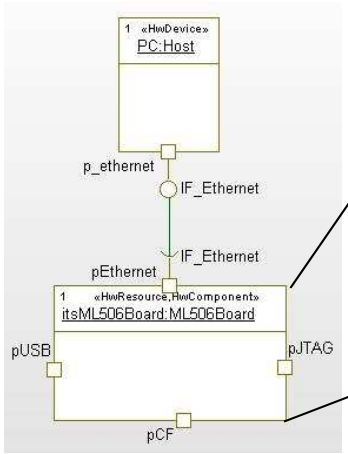
- **Functional model view at highest level of modeling**
  - Class diagram of Transmitter

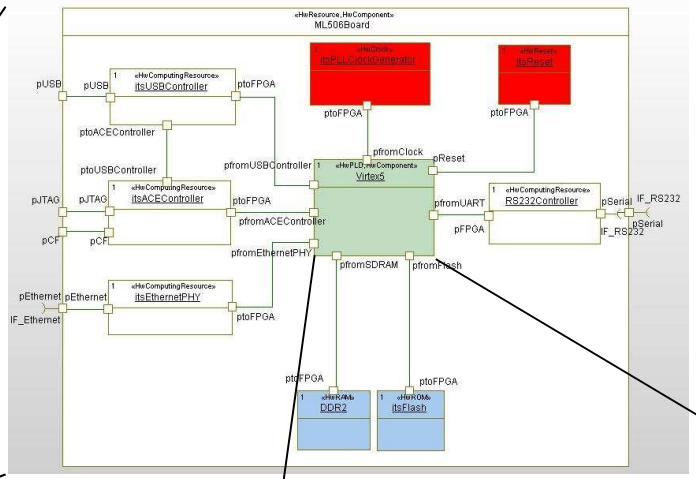


The 3 configurations of Mapper function

Instance of HDCRAM for the Transmitter

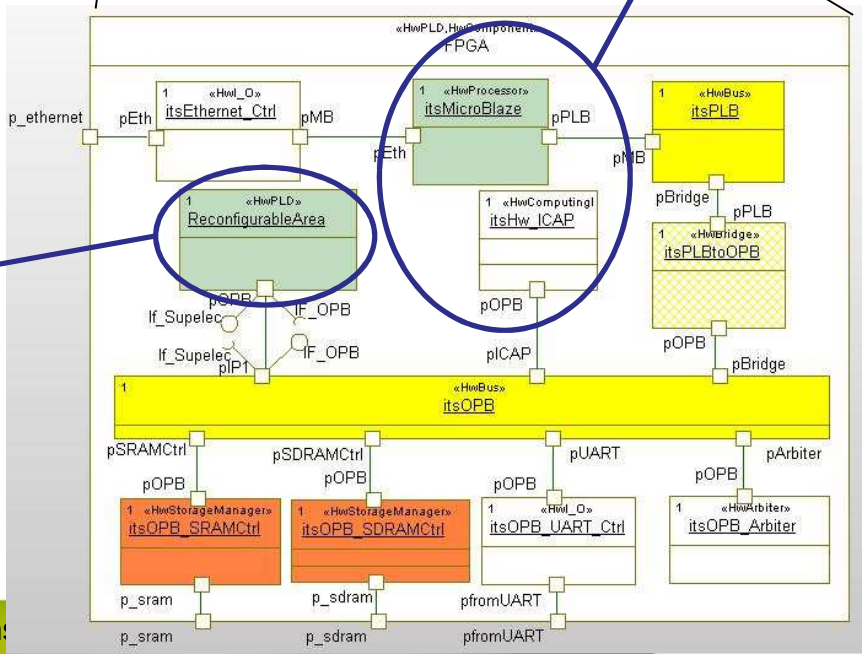
Detailed Modeling Level (DML)/PM





Used to manage the partial reconfiguration

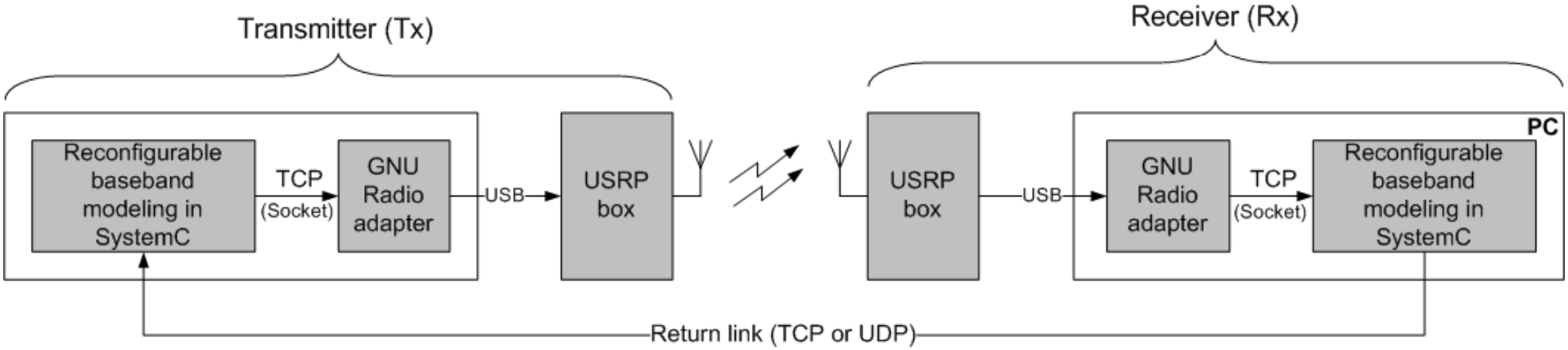
Identify this PLD resource to a reconfigurable resource with a specific **tag**



Demonstration of real wireless transmission

"A SystemC Radio-in-the-Loop Modeling for Cognitive Radio Equipments"

- Transmitter and Receiver baseband **modeling and simulation** in SystemC
- Two USRP boards for wireless transmission
- GNU radio adapter between SystemC model and USRP board



➔ See you at **Technology Showcase** session on Wednesday 17h30 - 19h30

Conclusion

- **Sum-Up**
  - New challenges in system co-design
  - Reconfigurability is a key enabler
  - Current co-design methodology not adapted for these new challenges

  ➔ high level design and modeling

- **Perspectives**
  - SystemC modeling and simulation at the lowest level (very near from implementation code and constraints)
  - UML to SystemC automatic code generation
  - Automatic HDL code for reconfigurable system

MOPCOM SoC/SoPC project*

- **Funding**
  - French National Research Agency
  - Brittany and Pays-de-Loire regions
  - The cluster « Media & Networks »
- **Partners**
  - Thales Airborne Systems
  - Technicolor
  - Sodius
  - ENSIETA
  - Lab-STICC from University of South Brittany
  - Supélec (SCEE team)**
  - INIRA (Triskell team)

* MOPCOM website : www.mopcom.fr

** SCEE team website: www.rennes.supelec.fr/ren/rd/scee/

Thanks !!!

Questions & Discussions

Stéphane LECOMTE[1,2], Christophe MOY[1] et Pierre LERAY[1]

[1]Supélec/IETR, SCEE Team, Rennes, France
[2]Technicolor, Research & Innovation, Rennes, France
stephane.lecomte@supelec.fr