ENSTA Bretagne

Lab-STICC

JEAN-CHRISTOPHE LE LANN

ENSEIGNANT-CHERCHEUR

EMBEDDED SYSTEMS, MODEL-DRIVEN ENGINEERING

modaë
TECHNOLOGIES

From System-level models to heterogeneous embedded systems

Jean-Christophe Le Lann

Joel Champeau

Papa Issa Diallo

ENSTA-Bretagne / Labsticc

Pierre-Laurent Lagalaye

Modaë Technologies

Overview

- Introduction
- Experimental toolchain
- Models of computation
- Result and discussion
- Conclusions

Introduction

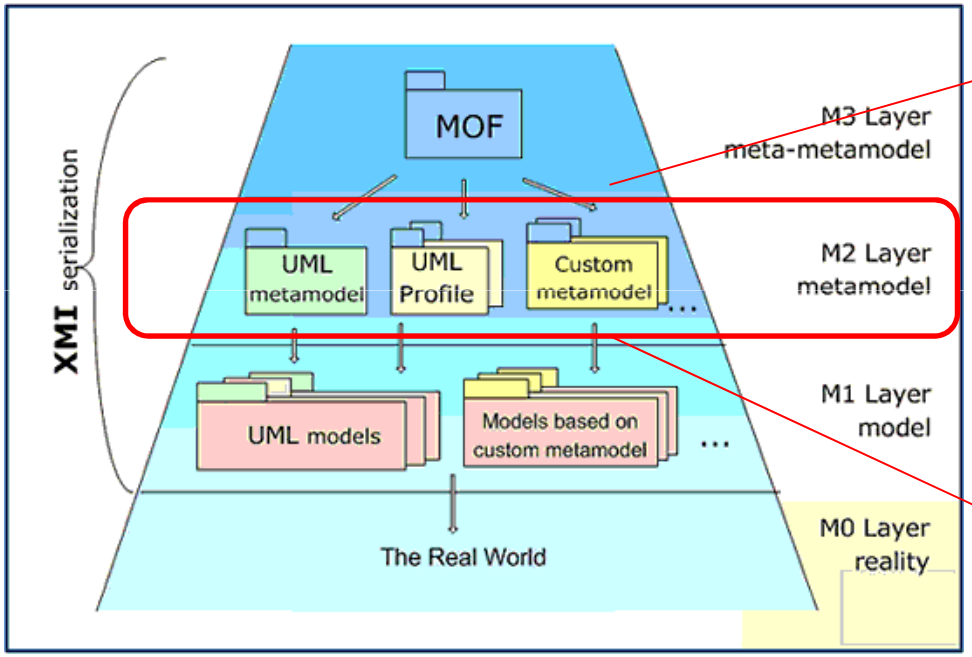
- History
  - 1999 : Thomson Multimedia / Technicolor
    - **System-level specification** for SoC design + **Mopcom ANR**
    - Video compression system
    - Multicore + SIMD ~15M gates
  - Needs :
    - ease of **algorithm capture** : data+ high-level control flow
    - Simulation : untimed, functional, data movments, events
    - Synthesis/compilation on heterogenous platforms
    - Allowing quick iterations in the design flow
  - 2009 : startup Modaë Technologies
    - **Interpretated languages** as input + DSL

New needs

- **System-engineering** practical aspects
  - IBM Rhapsody + UML 2.0 at the front
- **Software engineering** for embedded systems
  - Not only algorithmic, nor event-driven
  - Importance of object-oriented
- Need for **openness**
  - insurance of independence wrt tool providers
  - Quite different from classical ESL business-model
  - Facilitate toolchains development

Metamodeling

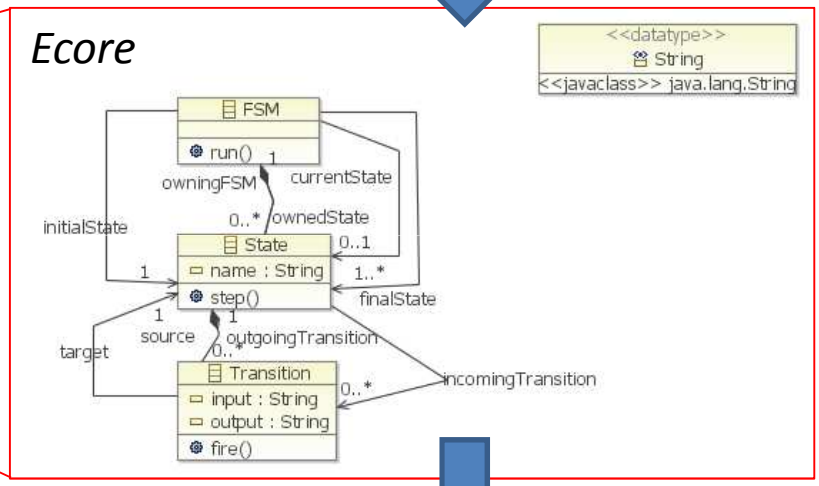
for tool development



MDD : model-driven development

Eclipse EMF support
Sodius MDWorkbench, …

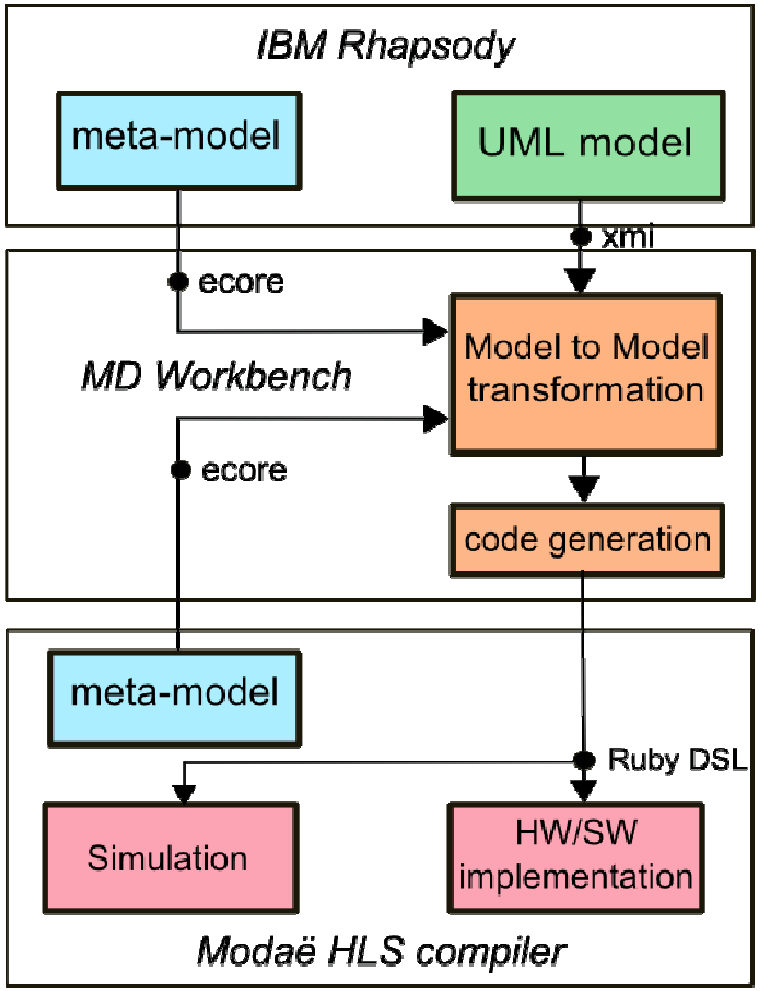
Business knowledge



Model edition

Code generation, …

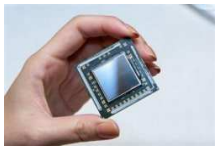
Experimental toolchain



Modeling in UML 2.0

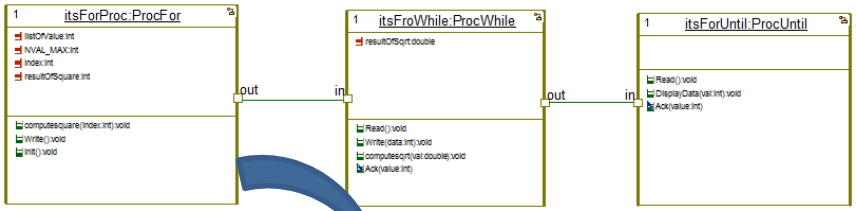
Transformation scripts in MDWorkbench

« Backend »
System-level synthesis
Modaë SLS

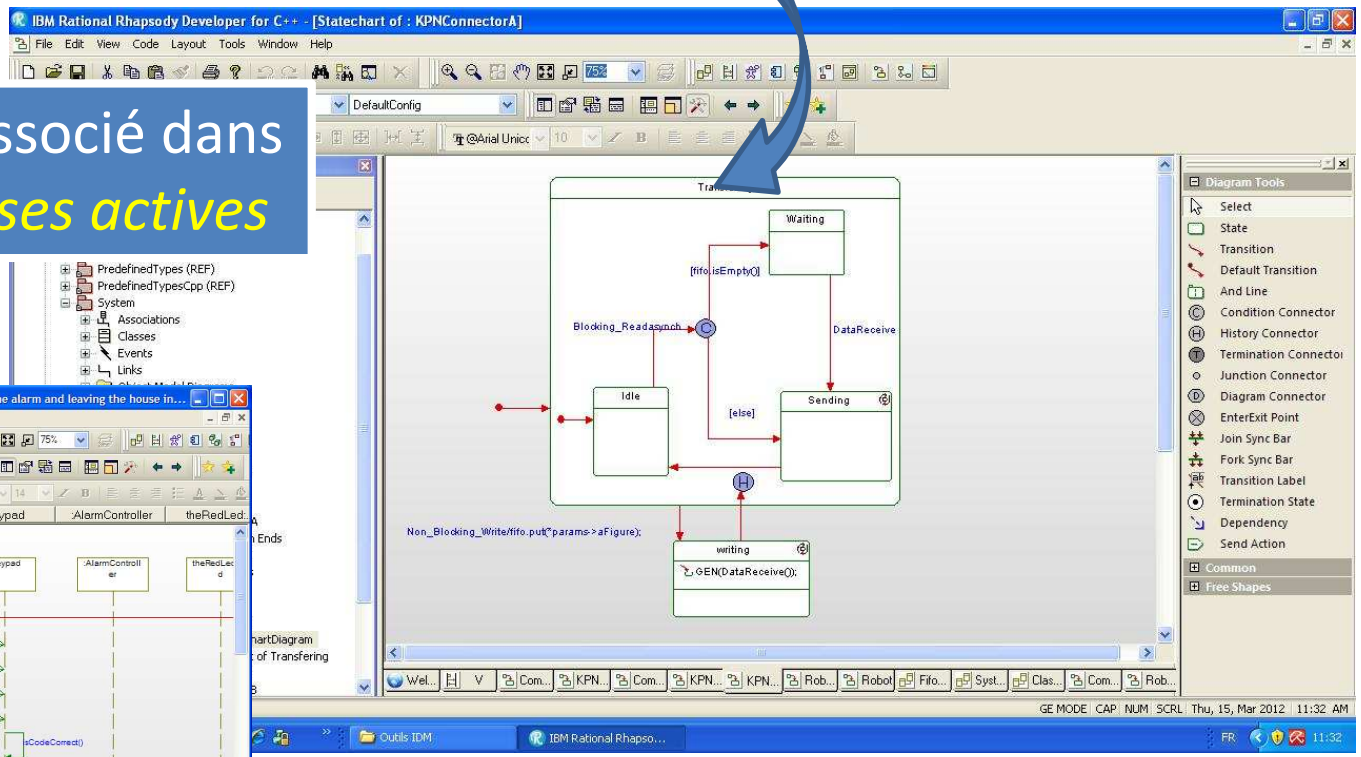


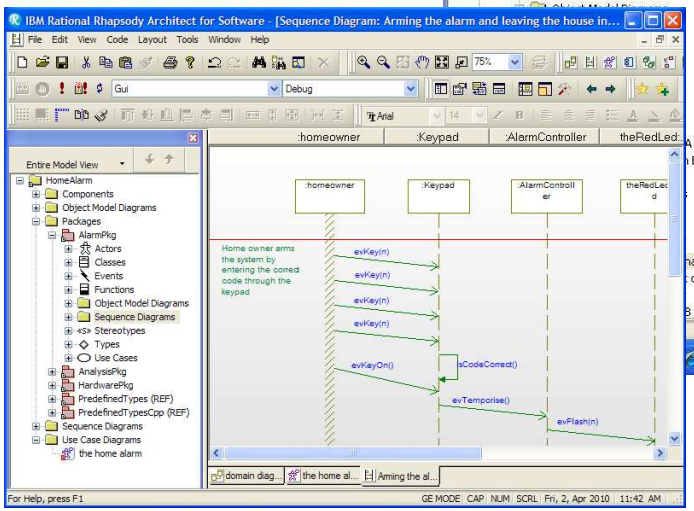
IBM Rhapsody UML 2.0

Diagramme de **composants** (éventuellement composite)

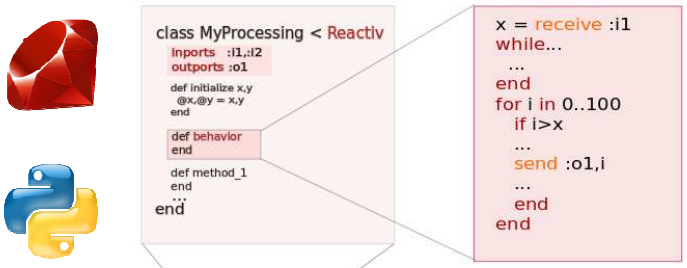


Statecharts associé dans le cas de *classes actives*

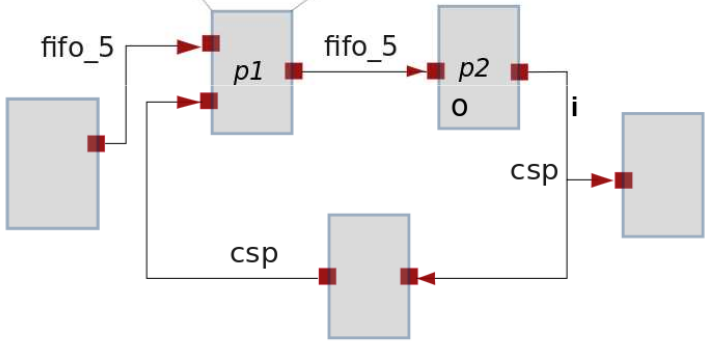




System modeling with *Modaë*



Ruby/Python algorithms, object-oriented



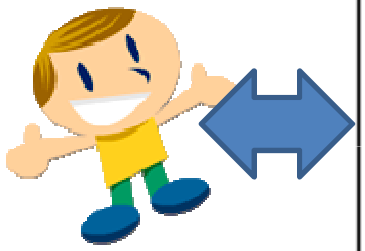
Addition of an internal DSL

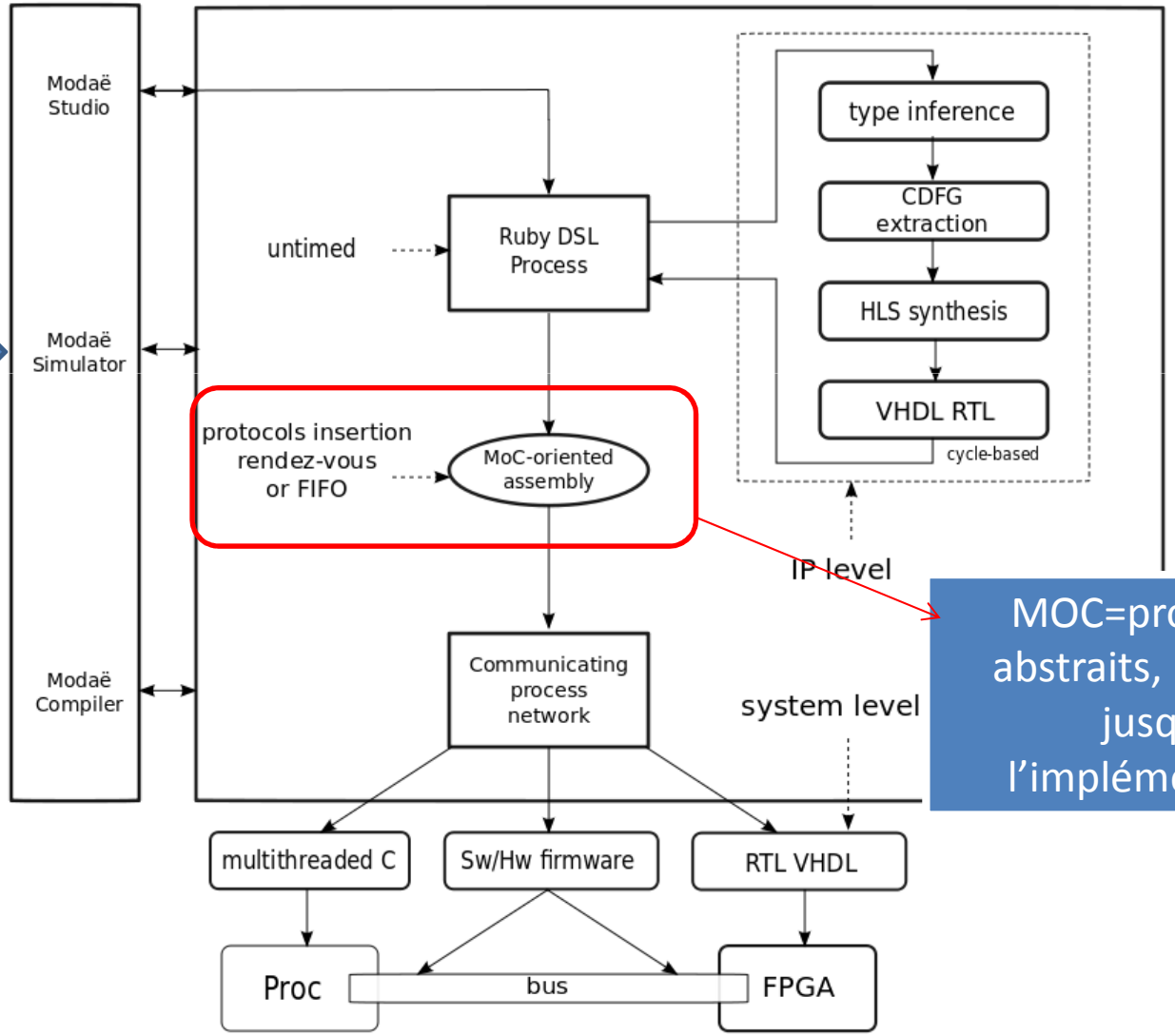
...graphical

```
Network.new('example') do
    p1=MyProcessing.new('p1')
    p2=....
    ...

    connect :fifo_5, p1.o => p2.i
    ...

end
```

...textual

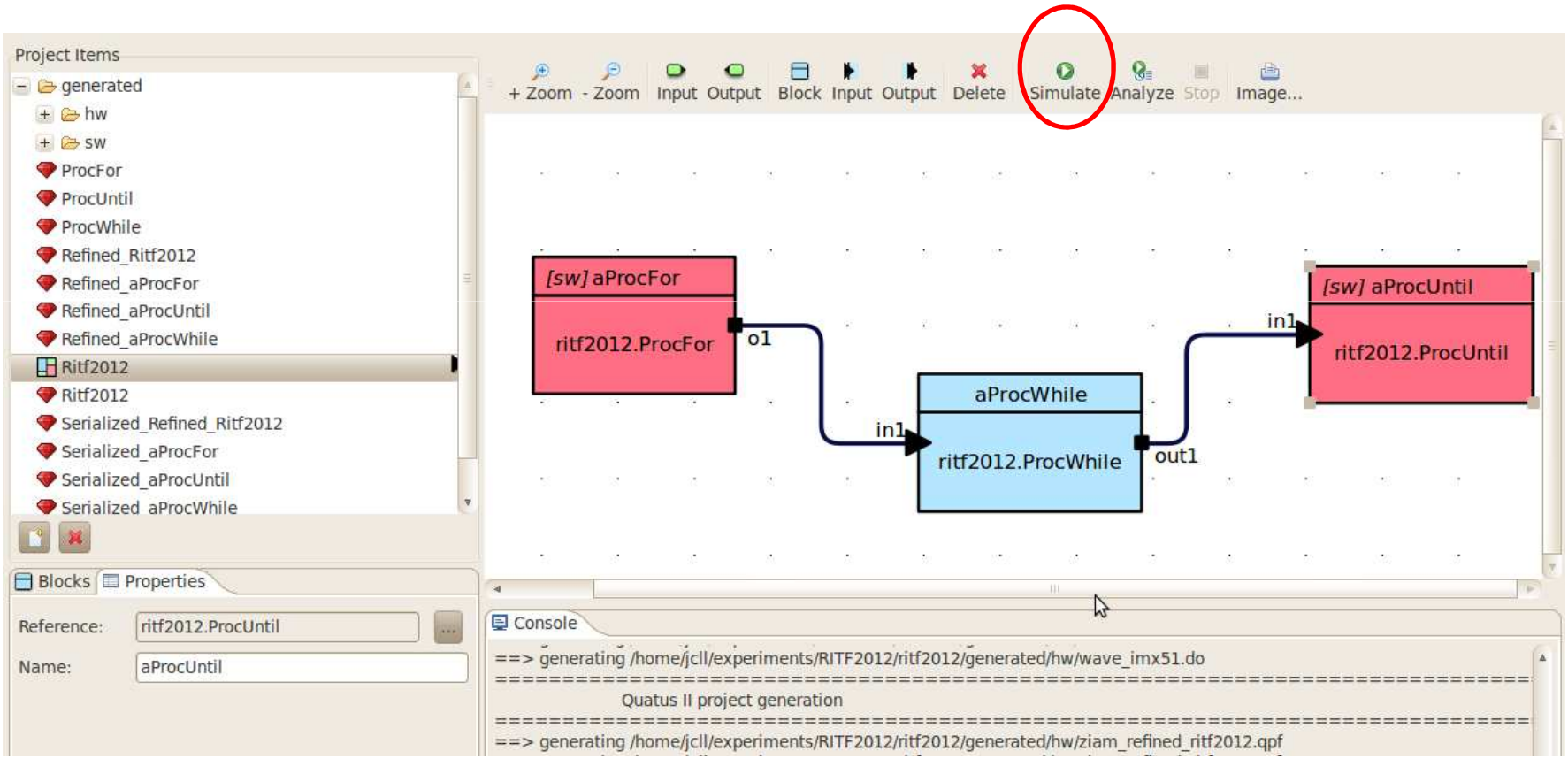
System modeling with *Modaë*



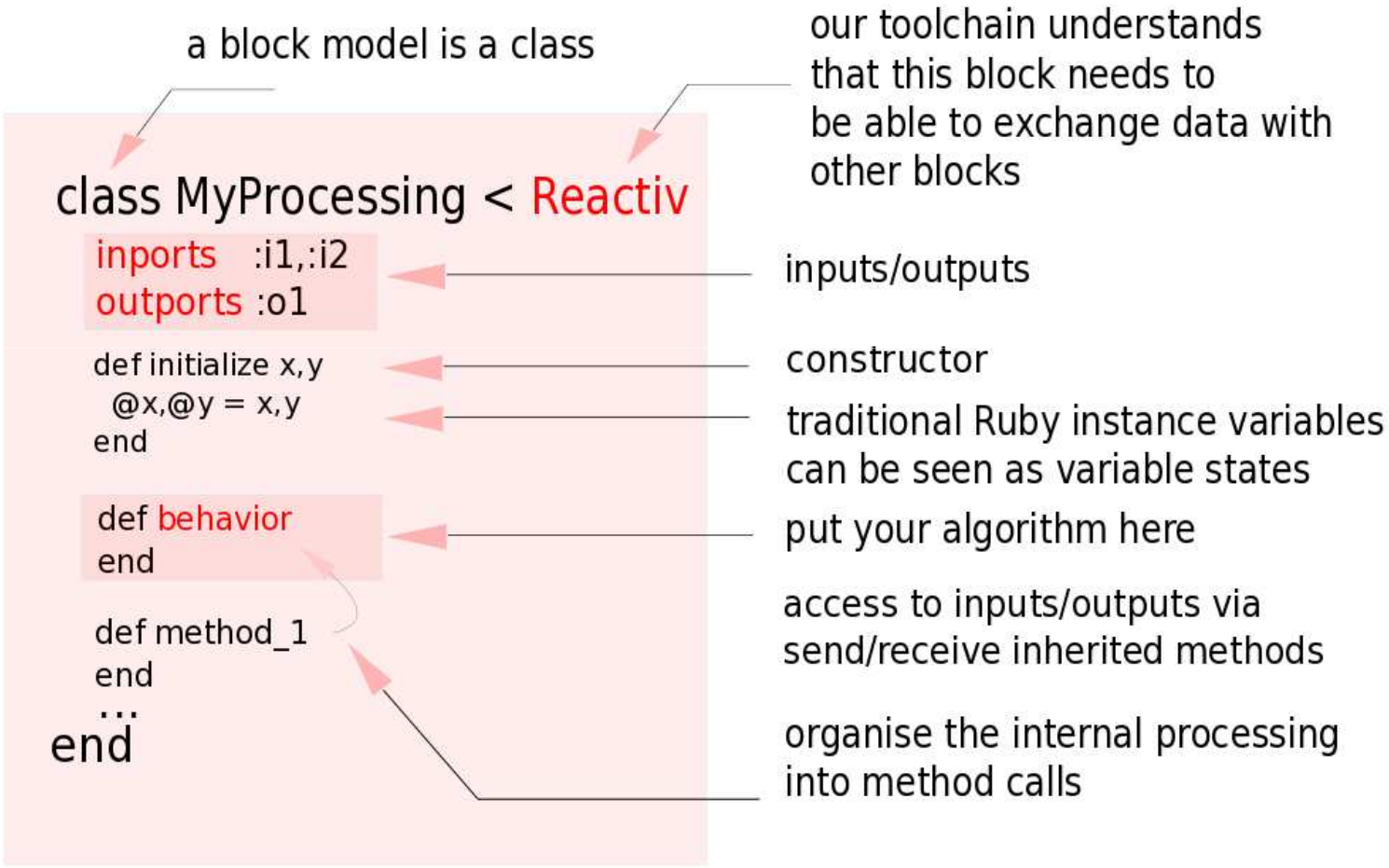


MOC=protocoles abstraits, maitrisés jusqu'à l'implémentation

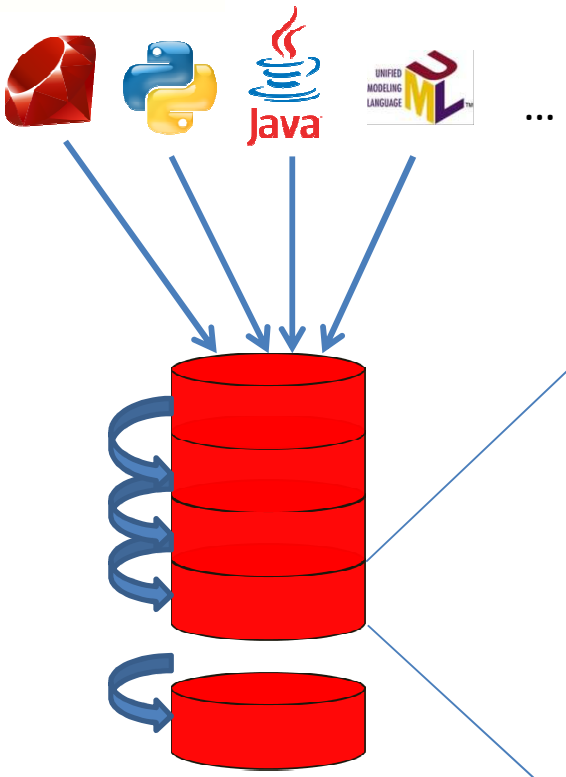
System modeling with *Modaë*



Behavioral blocks in *Modaë*



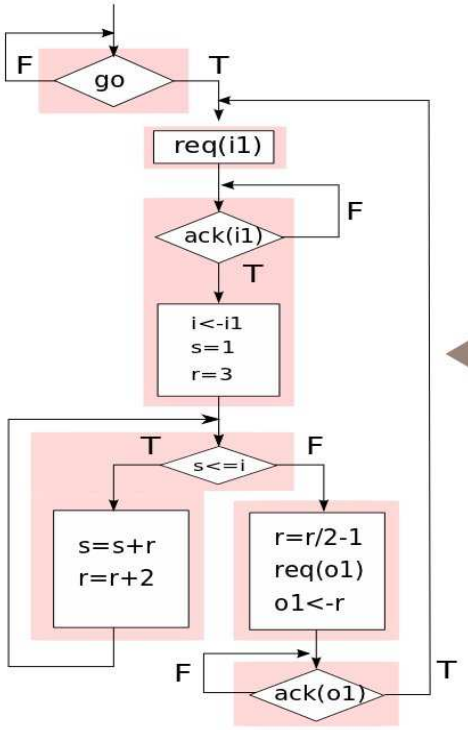
Modaë SLS (2/2)

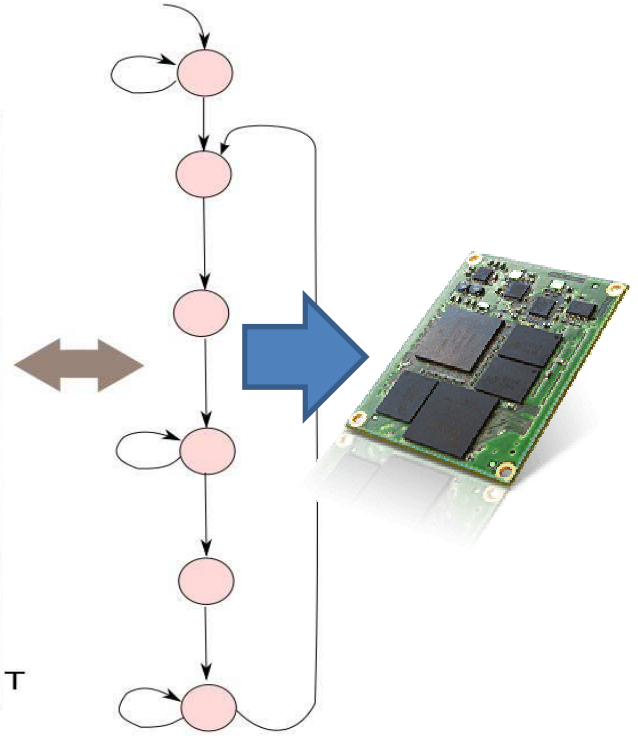


Internal representation

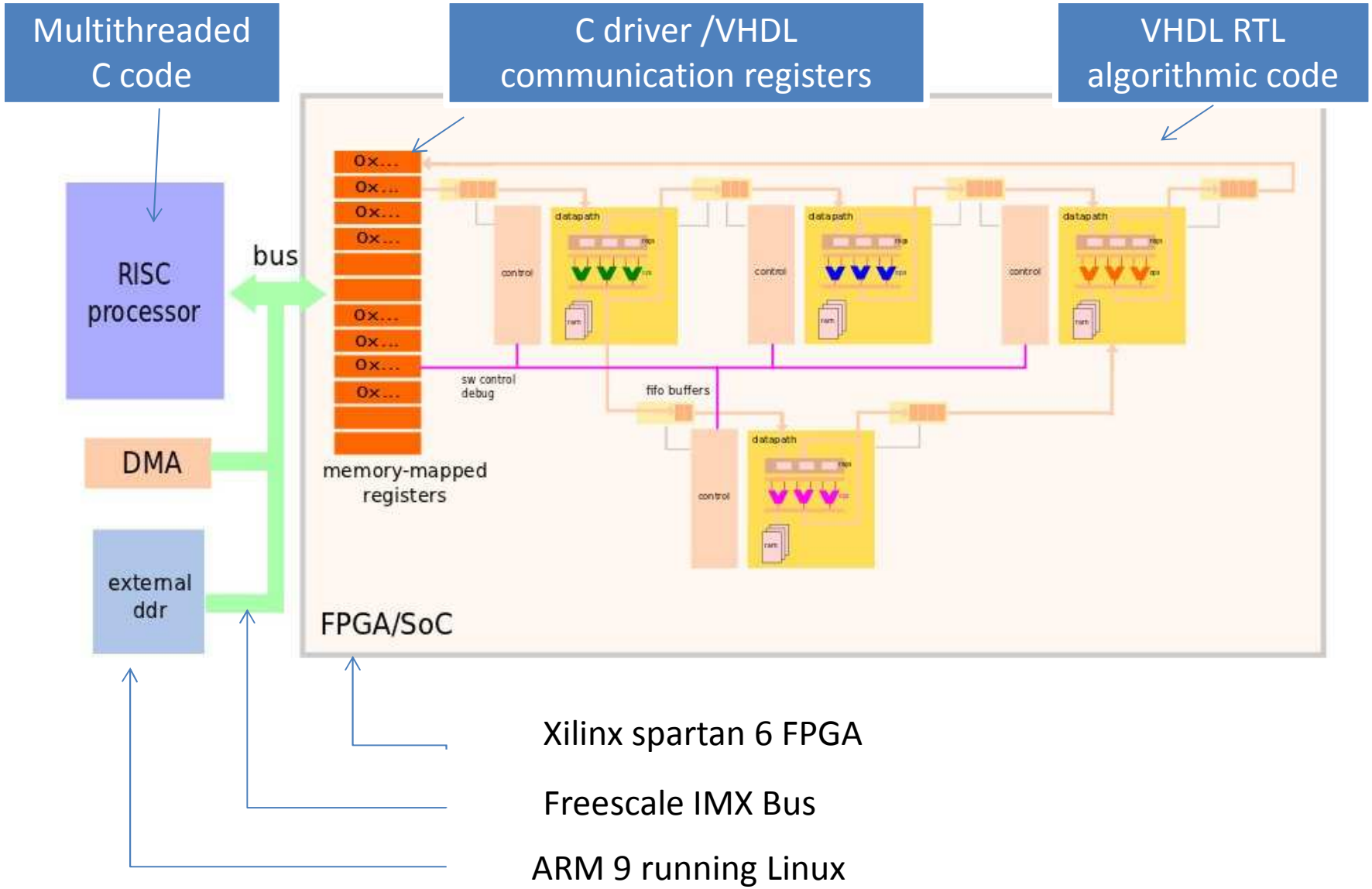
CDFG : Control-data flow graph

Synthesizable HDL code

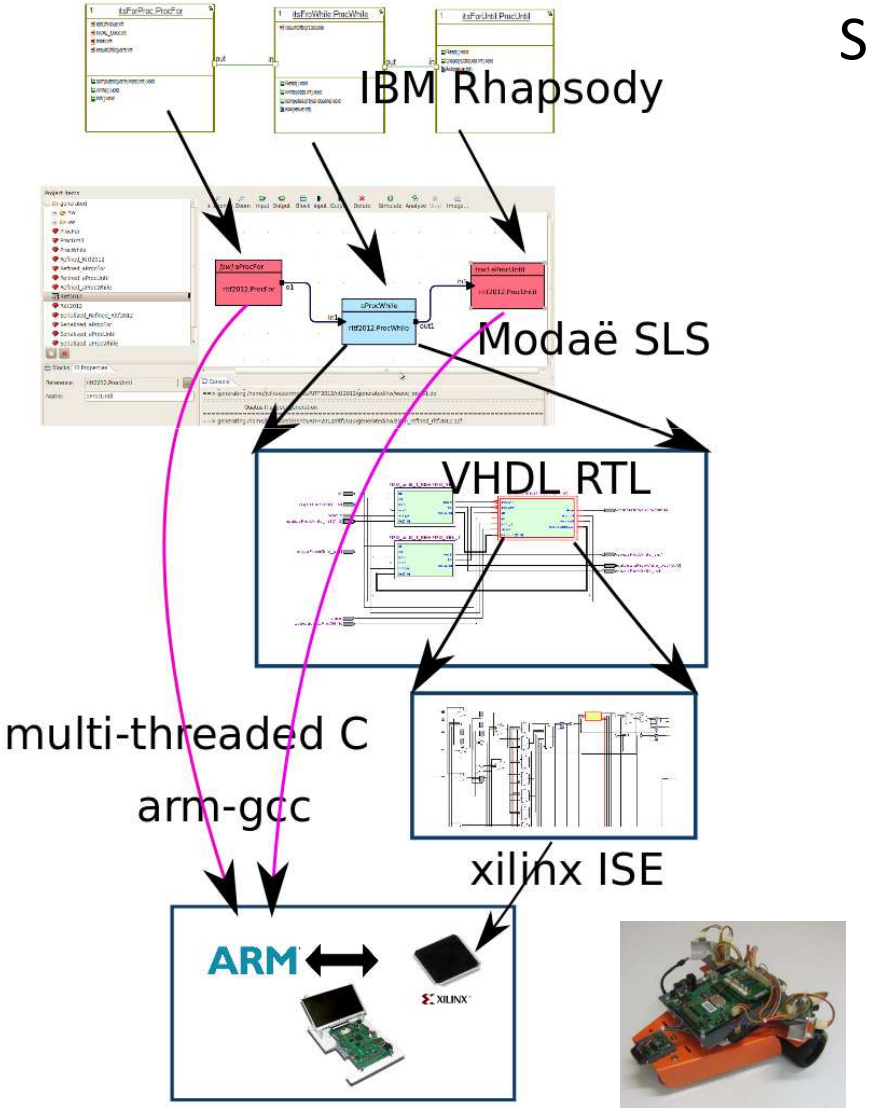




Architecture template



Preliminary results



Simple UML 2.0 model – simple action language

HW/SW mapping annotated

Software synthesis + HLS

VHDL RTL synthesis

Porting on platform

Conclusions

- Ceremonial system-level processes vs agile processes and languages
  - Possible interactions
  - Complementary
- MDD : several technologies to developp a system
  - Endogeneous vs Exogeneous battle ?
- Example :
  - Is UML easier / more expressive then Ruby +DSL ?

QUESTIONS ?

Jean-christophe.le_lann@ensta-bretagne.fr