A Framework for the Generation from UML/MARTE Models of IP/XACT HW Platform Descriptions for Multi-Level Performance Estimation
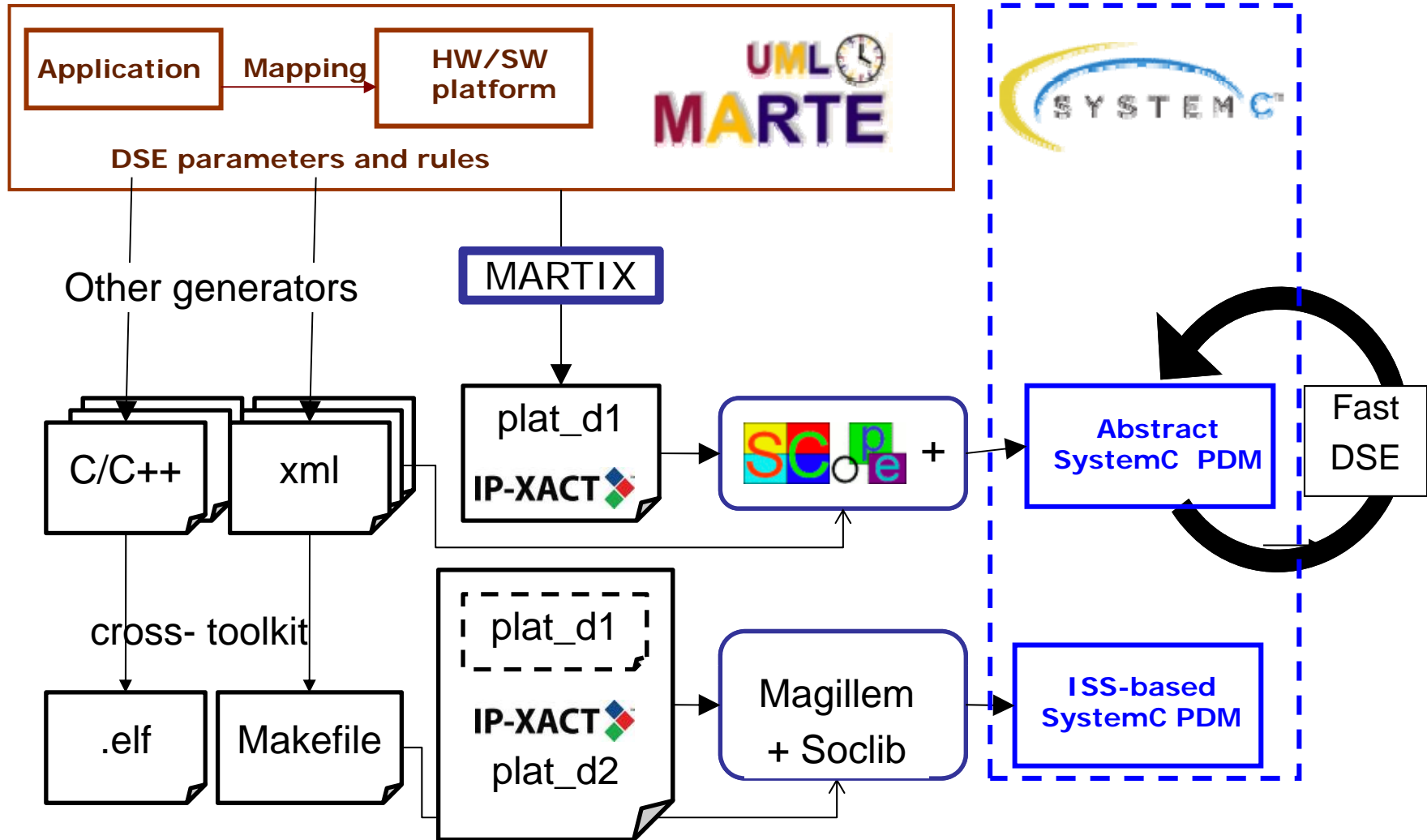
F. Herrera
E. Villar
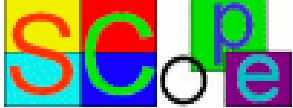
Outline

- Motivation
- Objectives
- **Mart**e to **I**P/**X**ACT Generator (Mart**I**X)
- Example
- Conclusions

Motivation:
UML/MARTE related flow in

SCoPE

- www.teisa.unican.es/scope

- Performance Estimation of MPSoC with NoC
  - Native Source Simulation
- Main Features
  - Output:
    - Performance Figures: Time, Power, CPU usage, Temperature,...
  - FAST:
    - Time estimation speed-up = 5 vs Virtualization / 100 vs ISS
    - Power estimation speed-up = NA vs Virtualization / 500 vs ISS
  - Input:
    - Application
    - HW/SW architecture, MPSoC with NoC
    - Output Metrics
    - **IP/XACT description of HW Platform**

IP-XACT

.xml File

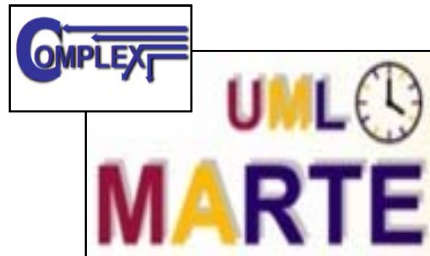**<spirit:design ... >   VLNV**

<spirit:componentInstances>

<spirit:componentInstance ... >
  <spirit:instanceName ... >
  <spirit:componentRef ... >
  <spirit:vendorExtensions ... >

<spirit:interconnections>

<spirit:interconnection ... >
  <spirit:activeInterface ... >
  <spirit:activeInterface ... >

.xml file

<spirit:component>

VLNV
 <spirit:BusInterfaces>
  <spirit:busInterface>
   <spirit:name>
   <spirit:busType>
   <spirit:abstractionType>
   <spirit:slave>
   <spirit:portMaps>

<spirit:vendorExtensions>
  <context:instanceClass>
   **<context:isInternalComponent>**
...

- Functional Information: vendorExtensions
  – SPRINT (SCIPIV) context labels
    • isProcessorComponent
    • isBusComponent
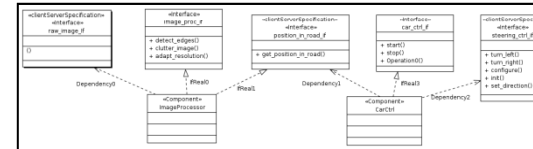   • isInternalComponent

Objectives

- Standard Format (in SW)
- Graphical, User Friendly
- Portable (Capture Tools)
- Embedded (MARTE)

- Extract Hw Platform
- Automatic
- Integration in a DSE framework
- Portable (Generation Environments**)**

- Standard Format (in HW)
- Traceability
- Potential Scalability
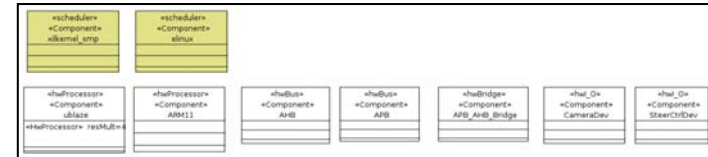
COMPLEX UML/MARTE Model
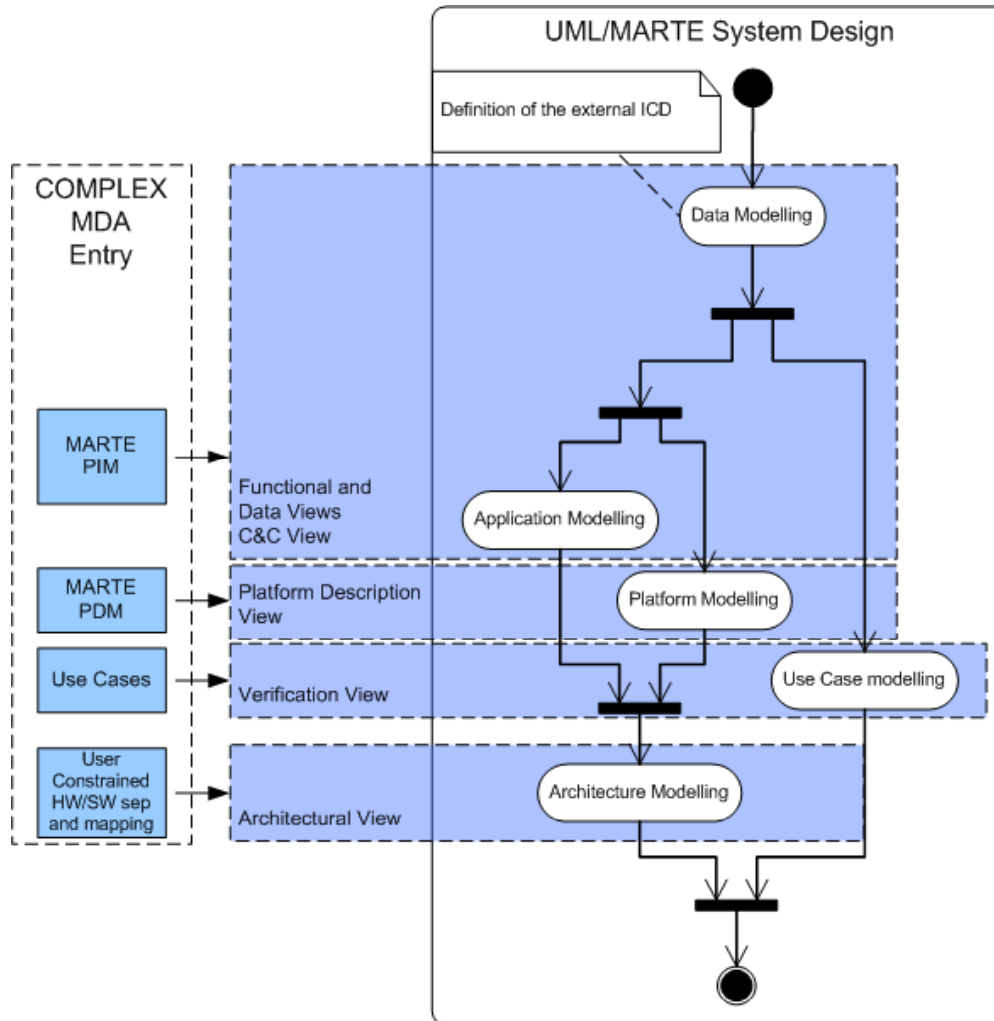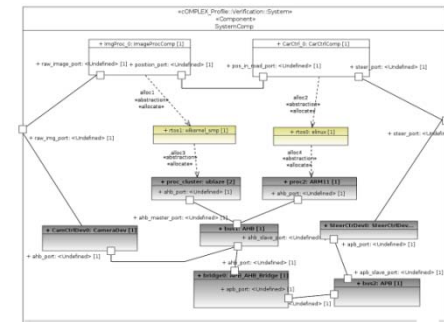

- Data view
- Functional view

- C&C view

- Platform view

- Architectural view

UML/MARTE Model: Platform View

- Declaration of System Components

SW Components

«scheduler»
«Component»
xilkernel_smp

«scheduler»
«Component»
elinux

«hwProcessor»
«Component»
ublaze

«HwProcessor» resMult=4

«hwProcessor»
«Component»
ARM11

«hwBus»
«Component»
AHB

«hwBus»
«Component»
APB

«hwBridge»
«Component»
APB_AHB_Bridge

«hwI_O»
«Component»
CameraDev

«hwI_O»
«Component»
SteerCtrlDev
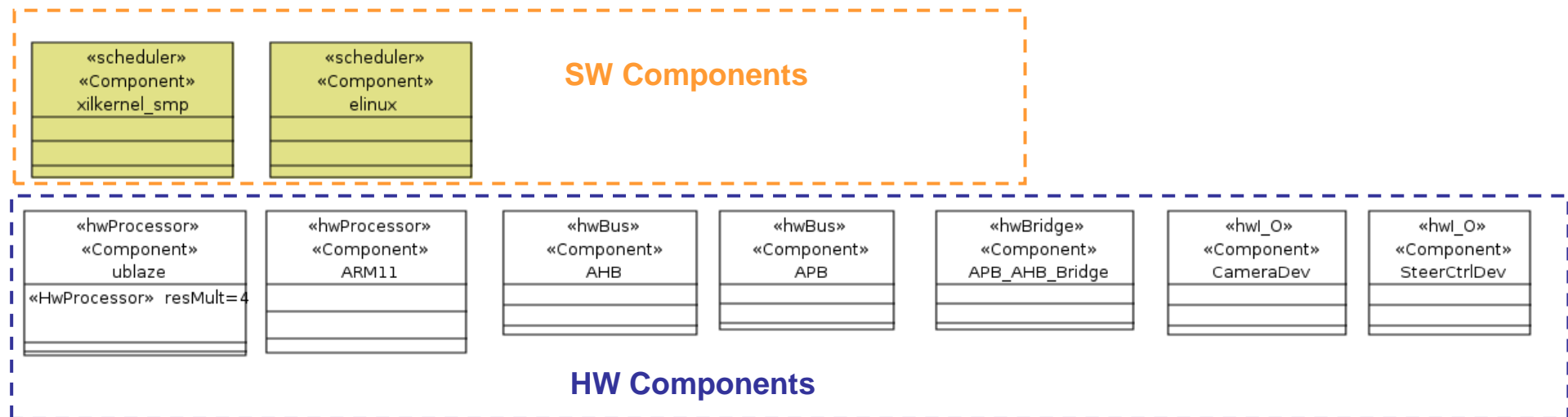
HW Components

UML/MARTE Model: Architectural View
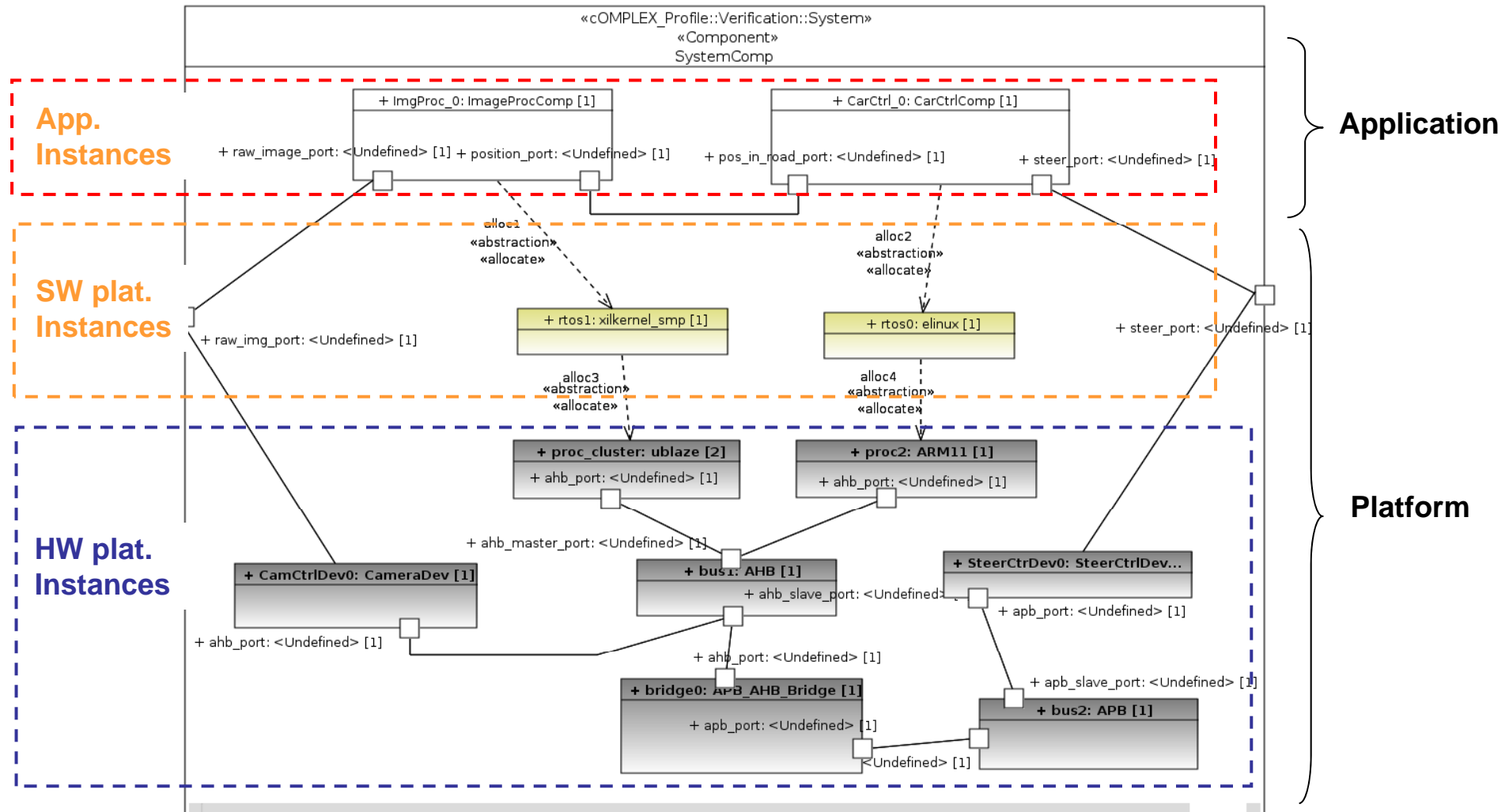
Fundamental Mapping Rules

Example

- Platform View
  - HW Components

- Architectural View
  - HW Architecture
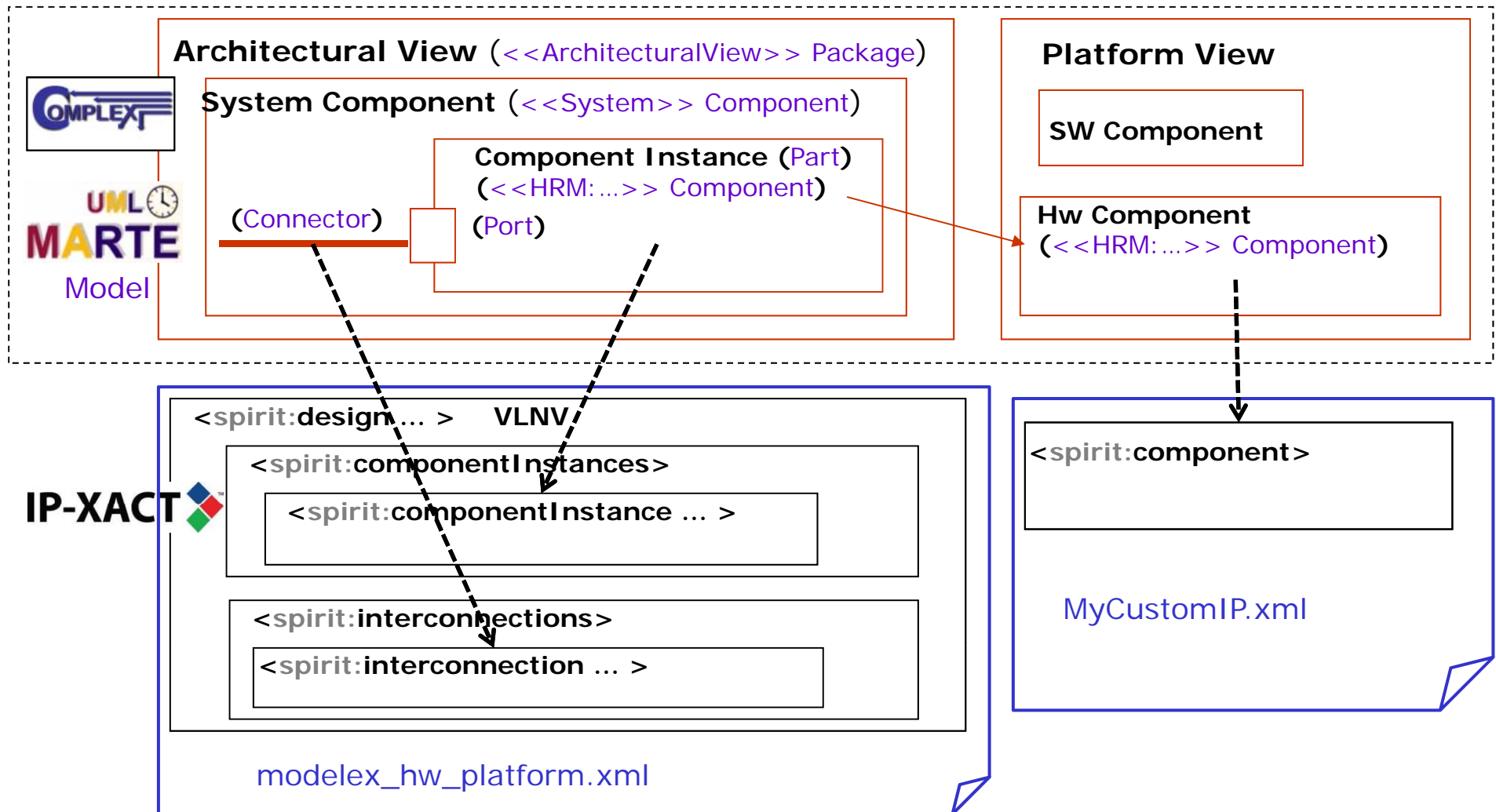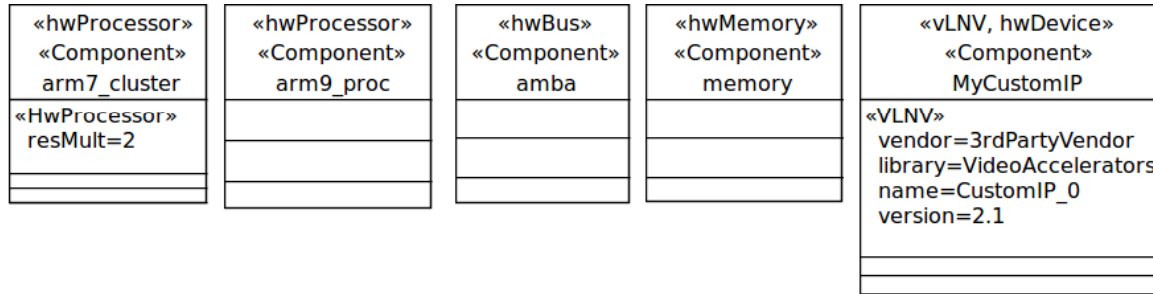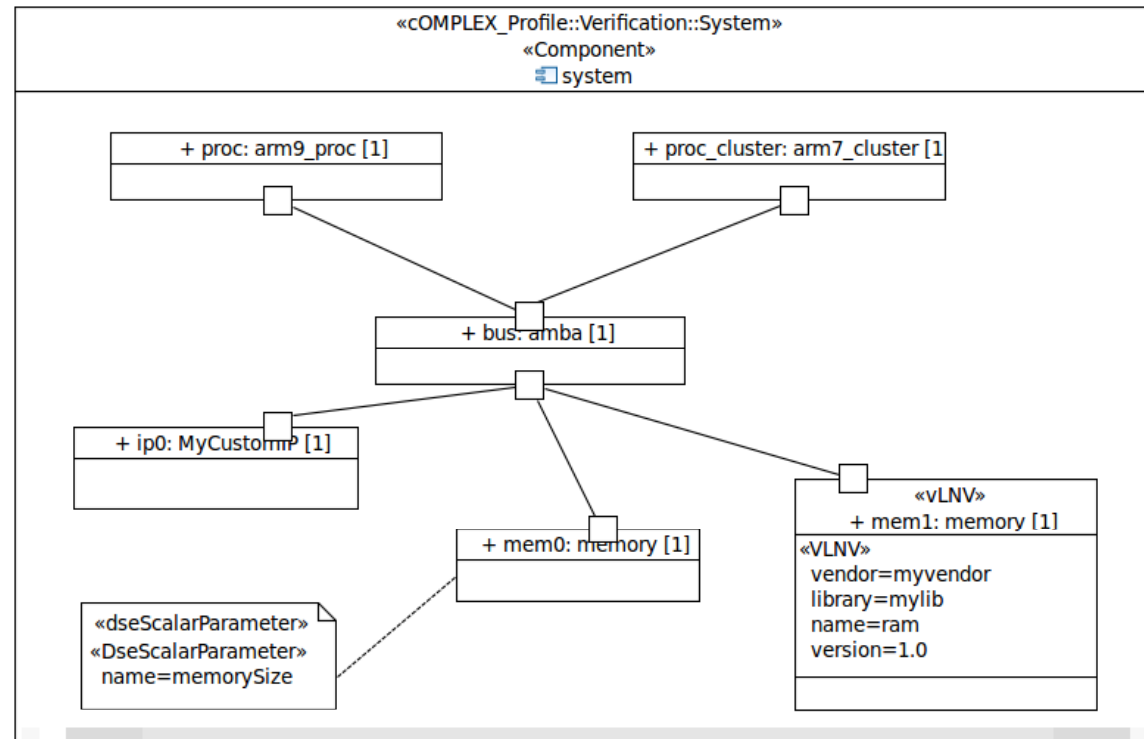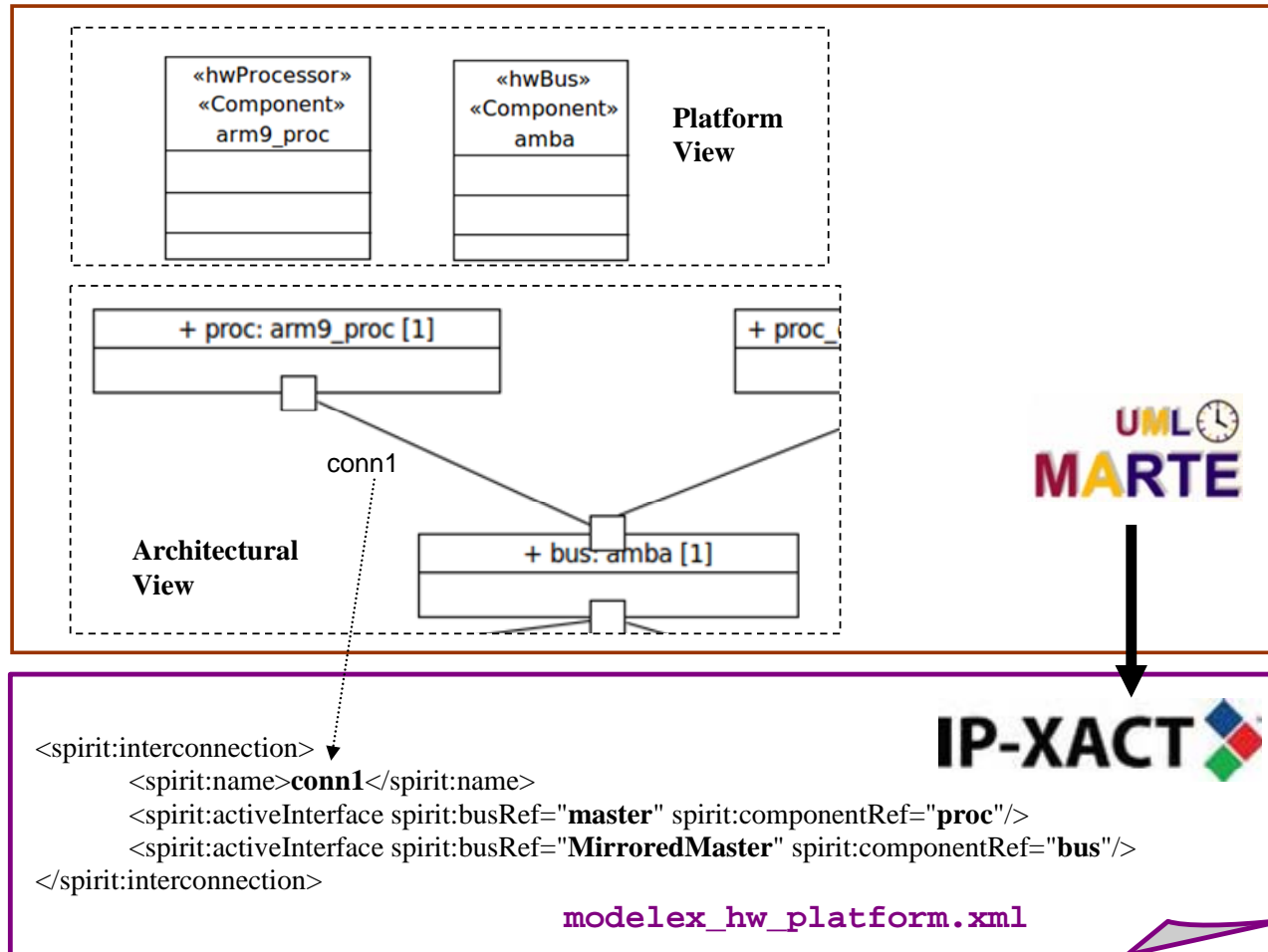
Inference of Interconnection

- Inference of Master/Slave Role

Inference of Multiple Instances

- Compact Model
- Inference from MARTE resMult stereotype
- Infere multiple instances and their corresponding interconnections


```
<spirit:componentInstance>
    <spirit:instanceName> proc_cluster_1 </spirit:instanceName>
  …
</spirit:componentInstance>

<spirit:componentInstance>
    <spirit:instanceName> proc_cluster_2 </spirit:instanceName>
  …
</spirit:componentInstance>

<spirit:interconnection>
    <spirit:name>conn0_1</spirit:name>
    <spirit:activeInterface spirit:busRef="master" spirit:componentRef="proc_cluster_1"/>
…
<spirit:interconnection>

<spirit:interconnection>
    <spirit:name>conn0_2</spirit:name>
    <spirit:activeInterface spirit:busRef="master" spirit:componentRef="proc_cluster_2"/>
…
<spirit:interconnection>                    modelex_hw_platform.xml
```
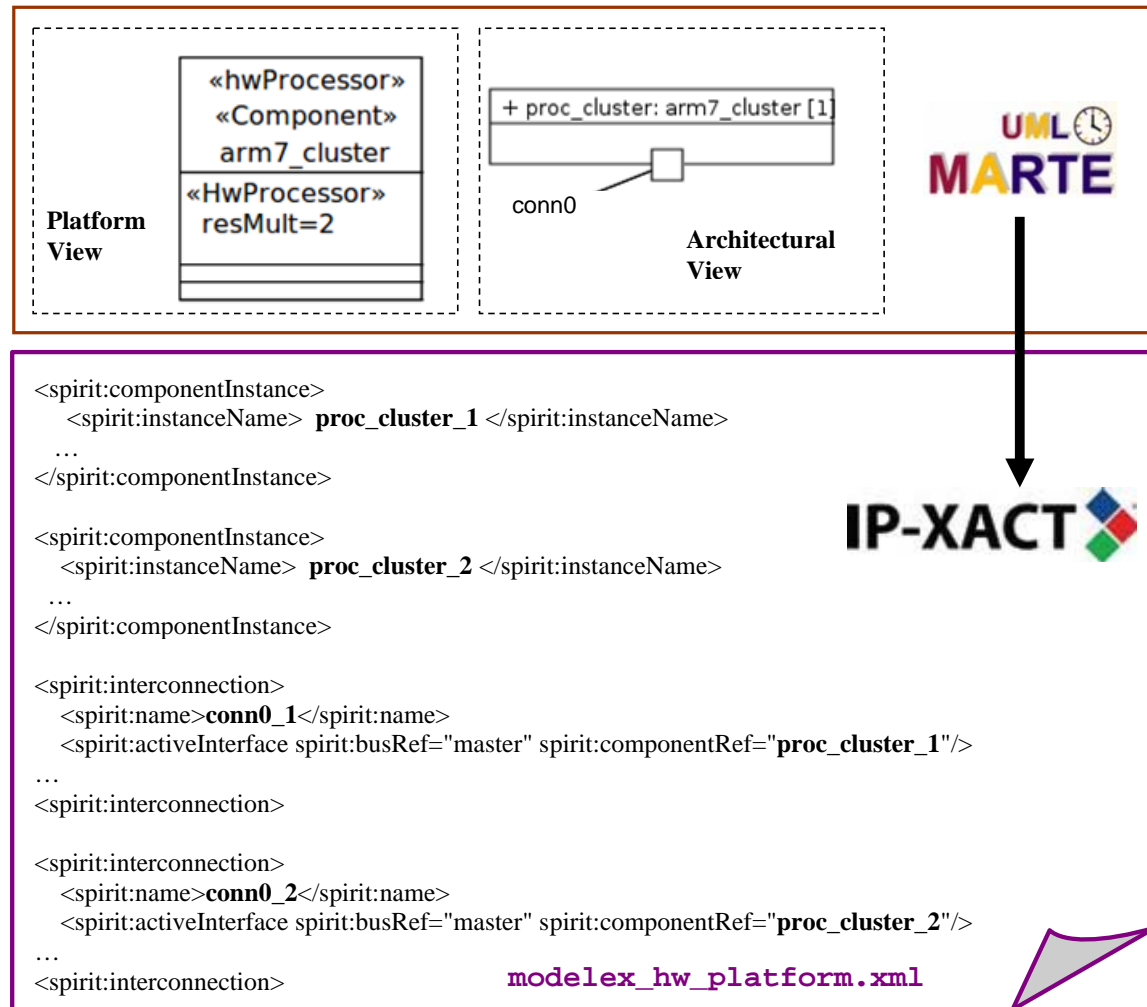
Inference of Configurable Descriptions

- Inference from COMPLEX stereotypes for DSE
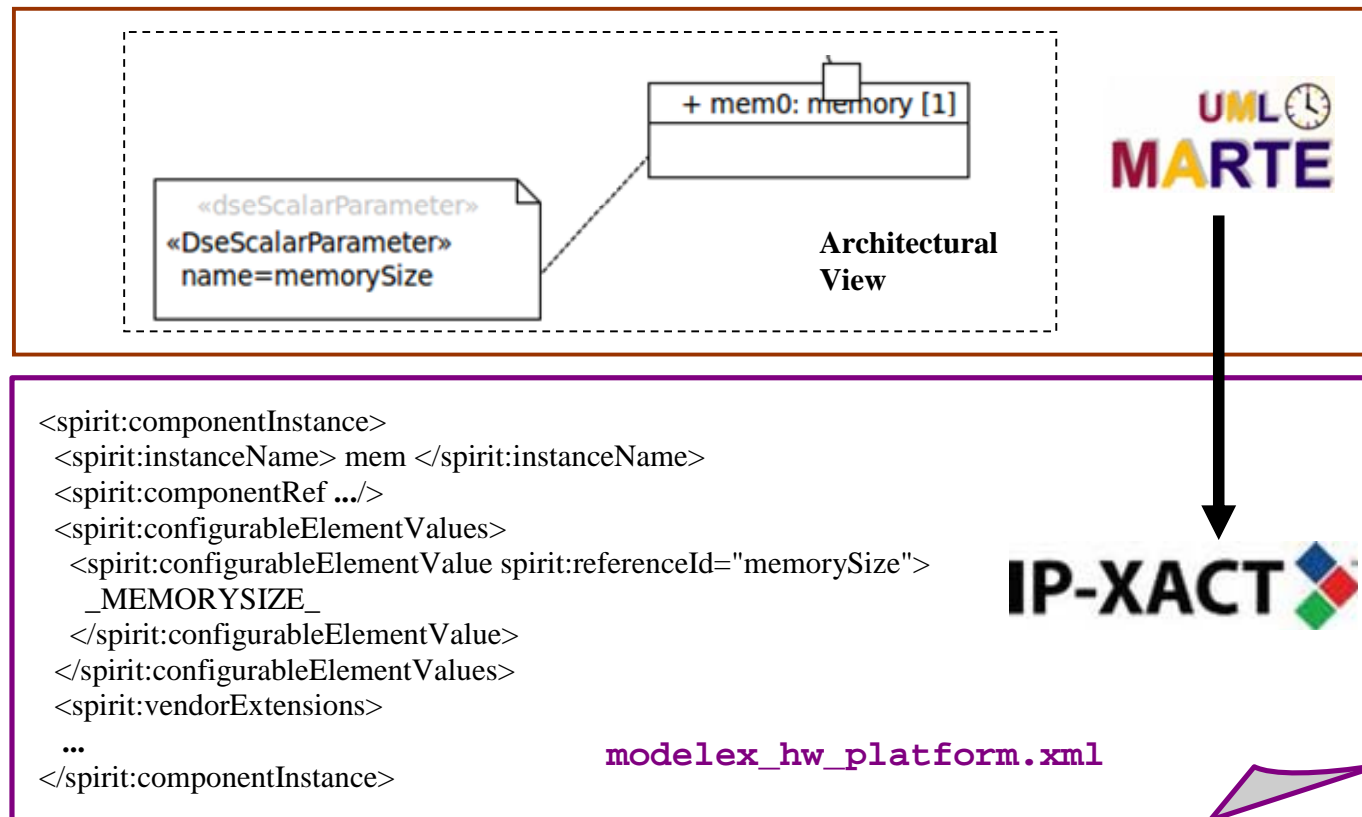
Multi-Level support: Inference of SCoPE component instances

- Default inference of instances of SCoPE components
  - SCoPE VLNV
  - Context Labels (within vendorExtensions)

UML/MARTE	Infered IPXACT entry	
Component Type of Part	**SCoPE VLNV**	**Context label**
HwProcessor	proc	isProcessorComponent
HwBus	bus	isNetworkComponent
HwMemory, HwRAM, HwROM	mem	isInternalComponent
Other components	part name	isInternalComponent

Multi-Level support: Inference of SCoPE component instances
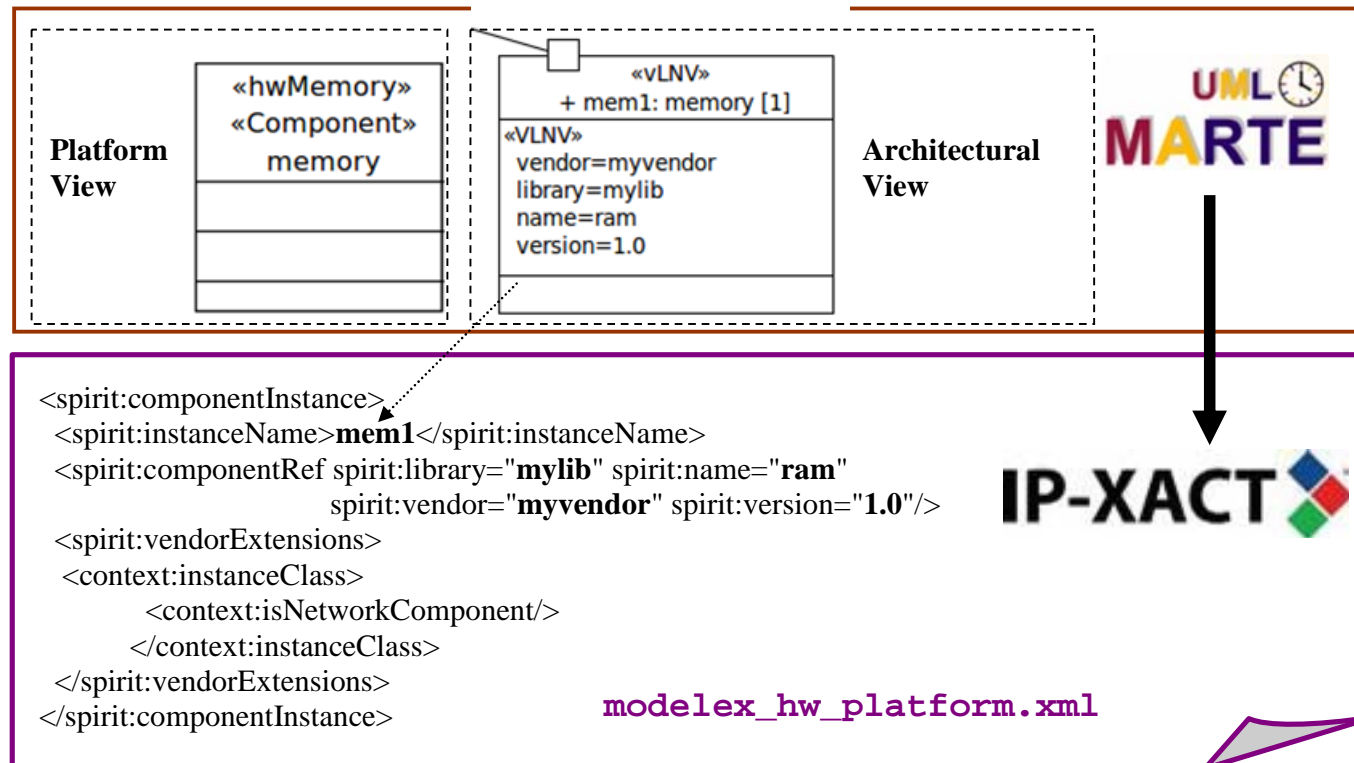
- e.g., SCoPE bus component inference
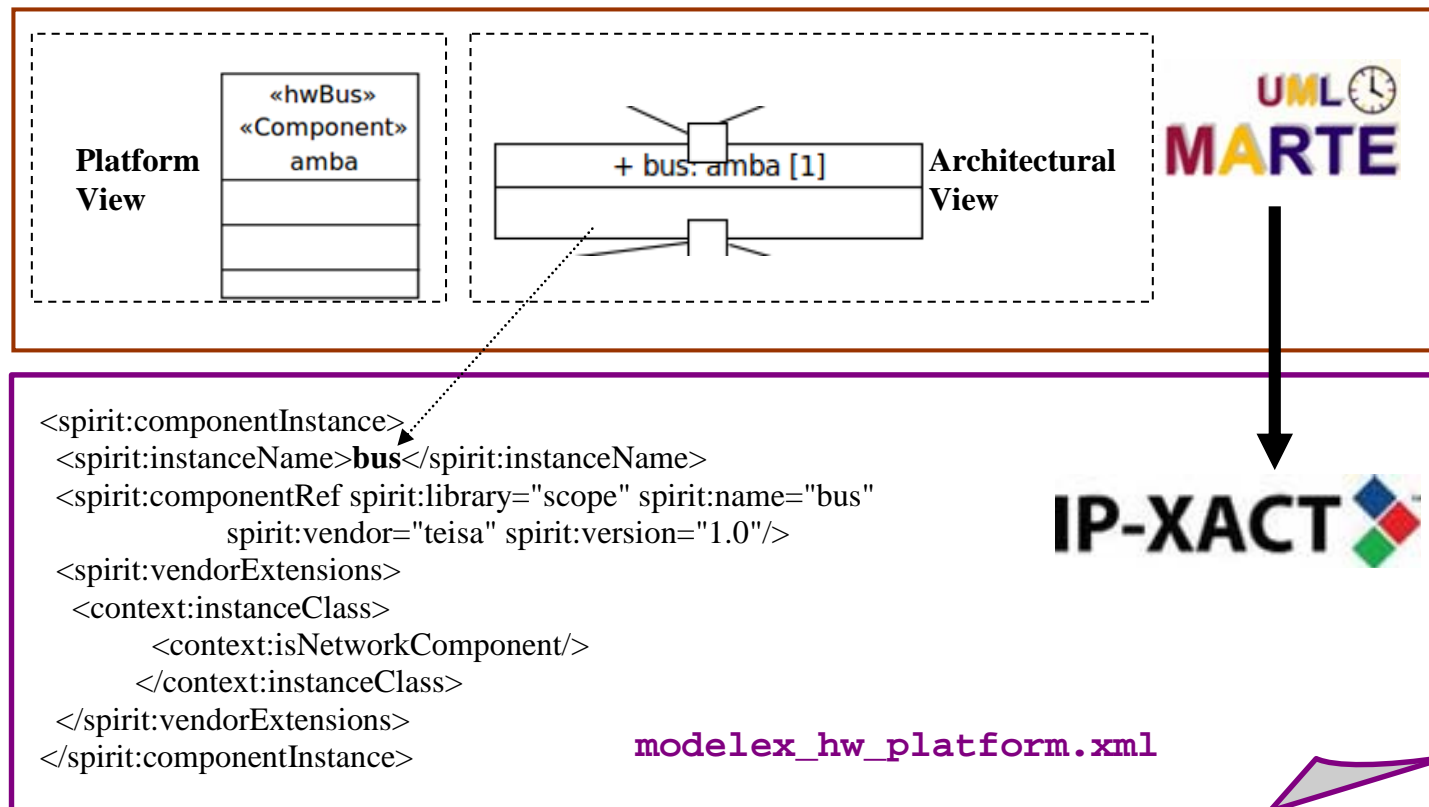
Multi-Level support: Inference of IP/XACT component instances

- Support third party components with IP/XACT wrapper
- IP/XACT stereotype (VLNV identifier) overides default inference of SCoPE VLNV


```
<spirit:componentInstance>
  <spirit:instanceName>mem1</spirit:instanceName>
  <spirit:componentRef spirit:library="mylib" spirit:name="ram"
                       spirit:vendor="myvendor" spirit:version="1.0"/>
  <spirit:vendorExtensions>
   <context:instanceClass>
        <context:isNetworkComponent/>
       </context:instanceClass>
  </spirit:vendorExtensions>
</spirit:componentInstance>
```

modelex_hw_platform.xml

Standard and Portable Implementation

- Template based Implementation
  - **MTL standard** language
  - **Portable Generator**
    - Several transformation engines (AcceleoMTL, Xpand, Jet)

```
[module martix('http://www.eclipse.org/uml2/3.0.0/UML')/]

[template public matrix(m : Model)]
…
[/template]
```

Other Features

- XML Comments for tracing generation
- Checks of error conditions
    - dump errors as XML comments
    - dump errprs to COMPLEX console

Implementation

- Integrated Framework

- **Modelling**
  - PapyrusMDT
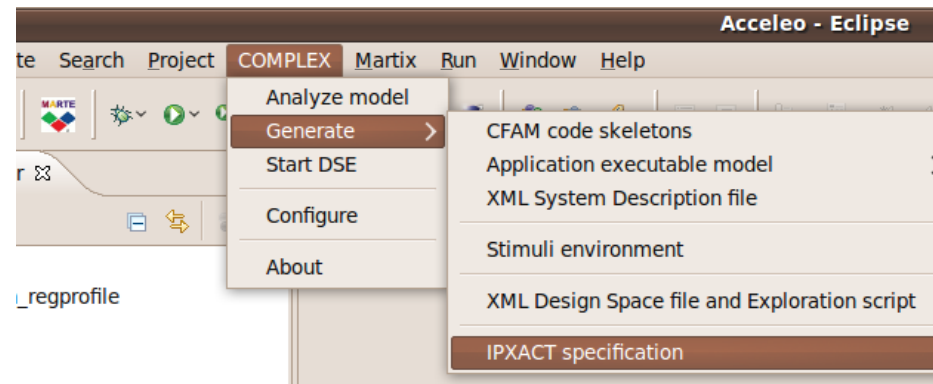   - MARTE Profile
    - COMPLEX Profile

- **Generation**
  - AcceleoMTL

Example

- Integration in COMPLEX framework

- Standalone Plugin

Conclusions

- Automatic Generation from (COMPLEX) UML/MARTE models of IP/XACT descriptions
  - Multi-Level: for fast DSE, for Performance Validation, for Implementation

- Other Features
  - Portable to different generation environments (supporting MTL)
  - Integrated in (COMPLEX) Eclipse DSE Environment

- Future
  - Compatibility with Maguillem tools

Thanks

- For your attention

- More Information:
  - Authors: {fherrera, evillar}@teisa.unican.es

  - UC/GIM: www.teisa.unican.es/gim

  - Complex: http://complex.offis.de