HiPEAC 2020
20-22 January, 2020, Bologna, Italy

CERBERO H2020 Project – Tutorial: Design Flow for Heterogeneous Embedded Computing Infrastructures

Tiziana Fanni[2], **Daniel Madroñal**[1], **Maxime Pelcat**[3], Alfonso Rodriguez[1], **Carlo Sau**[4], Claudio Rubattu [23], **Leonardo Suriano**[1]

[1]Universidad Politécnica de Madrid, [2]Università degli Studi di Sassari, [3]IETR INSA-Rennes, [4]Università degli Studi di Cagliari

Horizon 2020
European Union funding
for Research & Innovation

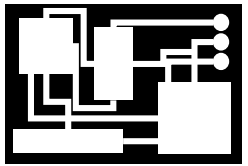
Adaptivity and Heterogeneity Requirements

- Here is a **Martian robot**

CERBERO

Adaptivity and Heterogeneity Requirements

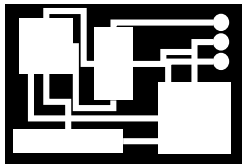
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**



CERBER

Adaptivity and Heterogeneity Requirements

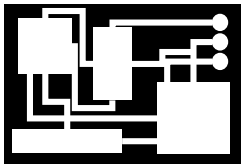
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot



CERBER

Adaptivity and Heterogeneity Requirements

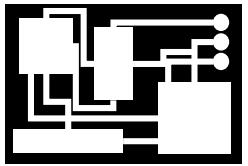
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot
  - one of its **cameras** can be occluded



CERBER

Adaptivity and Heterogeneity Requirements

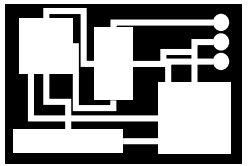
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot
  - one of its **cameras** can be occluded
  - one of its **wheels** can be blocked



CERBER

Adaptivity and Heterogeneity Requirements

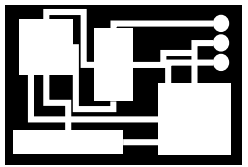
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot
  - one of its **cameras** can be occluded
  - one of its **wheels** can be blocked
  - one **arm** segment can be stuck



CERBER

Adaptivity and Heterogeneity Requirements

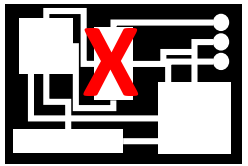
- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot
  - one of its **cameras** can be occluded
  - one of its **wheels** can be blocked
  - one **arm** segment can be stuck
  - one **motor** can overheat



CERBER

Adaptivity and Heterogeneity Requirements

- Here is a **Martian robot**
  - it has **wheels**, a **robotic arm**, many **sensors**, and a **complex control** and **computing system**
- Here is **what can happen** to this robot
  - one of its **cameras** can be occluded
  - one of its **wheels** can be blocked
  - one **arm** segment can be stuck
  - one **motor** can overheat
  - on **processing** itself, a region of hardware can be temporarily or permanently unavailable
    - in particular because of ionizing particles



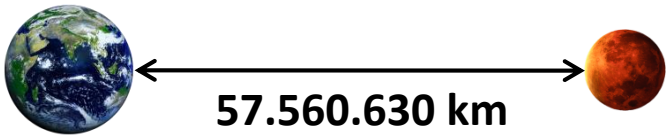
CERBERO

Adaptivity and Heterogeneity Requirements

- How do we **resume the mission** in case of **problems**?
  - or at least use **remaining robot capacities** for future operations?

Adaptivity and Heterogeneity Requirements

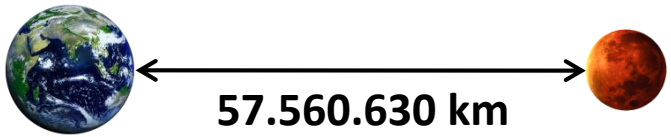
- How do we **resume the mission** in case of **problems**?
  - or at least use **remaining robot capacities** for future operations?
- By the way, **radio-controlling** is **not feasible**
  - **light** return trip to earth is **9 minutes** so…



CERBER

Adaptivity and Heterogeneity Requirements

- How do we **resume the mission** in case of **problems**?
  - or at least use **remaining robot capacities** for future operations?
- By the way, **radio-controlling** is **not feasible**
  - **light** return trip to earth is **9 minutes** so…
- Proposition
  - **Self-adaptation** support and **CERBERO** tooling
  - **HW/SW reconfiguration** for reliability and performance



Ultimate Goal: Self-Adaptation Support

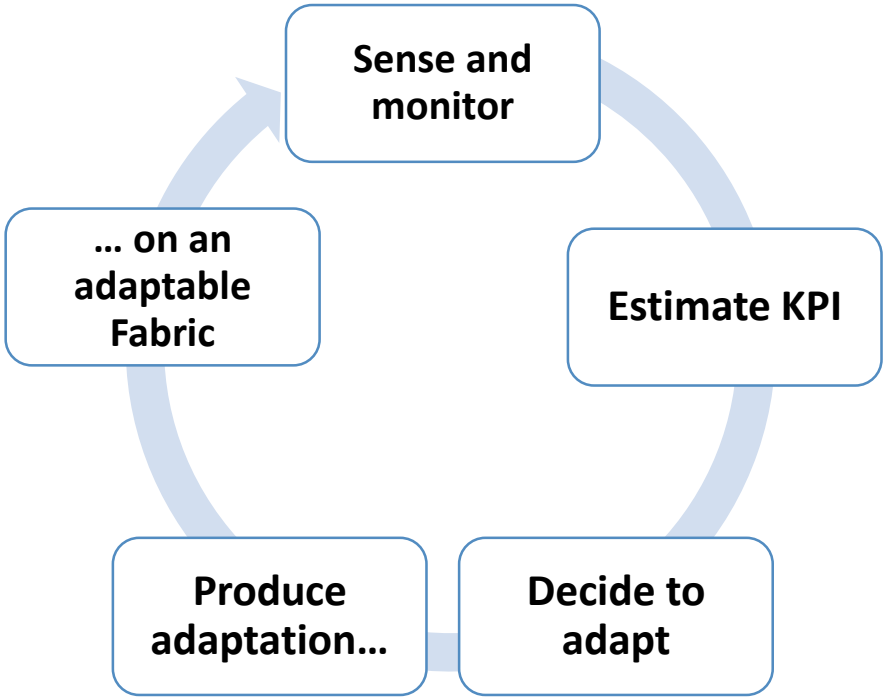
Self-adaptation: ***runtime*** action ***changing structure, functionality and/or parameters of a system***, according to environment, user or self-sensing info.

[F.D. Macías-Escrivá, et al. "*Self-adaptive systems: A survey of current approaches, research challenges and applications*" In Expert Systems with Applications, 2013]

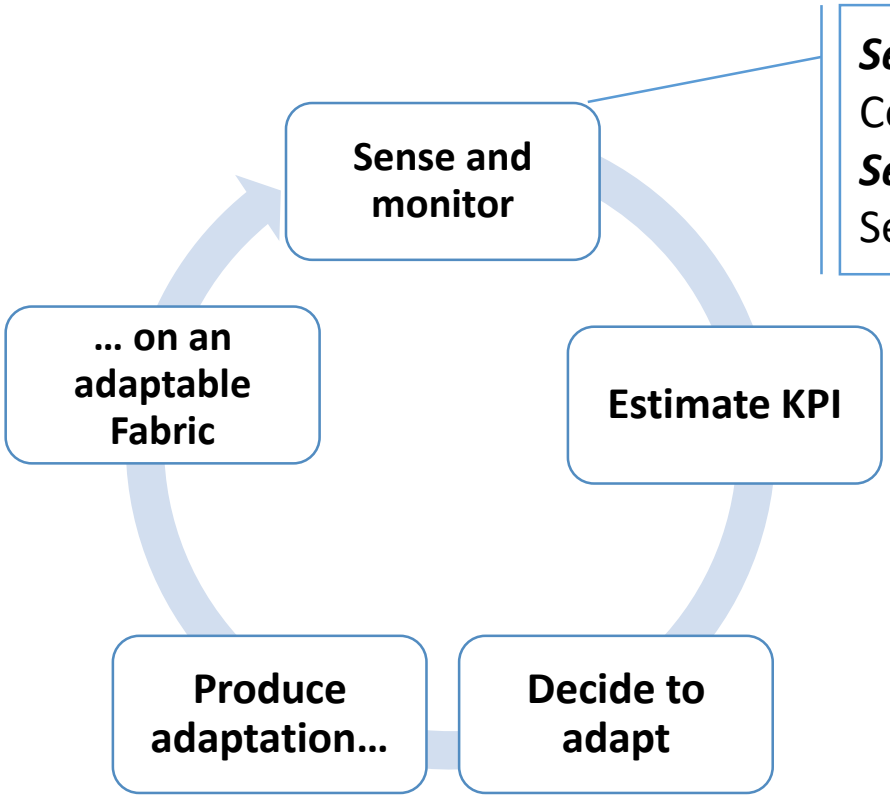
System self-adaptation: ***awareness*** + ***reconfiguration***.

- Reconfiguration → decided ***inside the CPS*** by a smart ***self-adaptation manager (hierarchy of managers)***.
- Decisions → based on run-time **Key Performance Indicators** (KPIs) measurements and estimations.

CERBERO Adaptation Loop Formalization

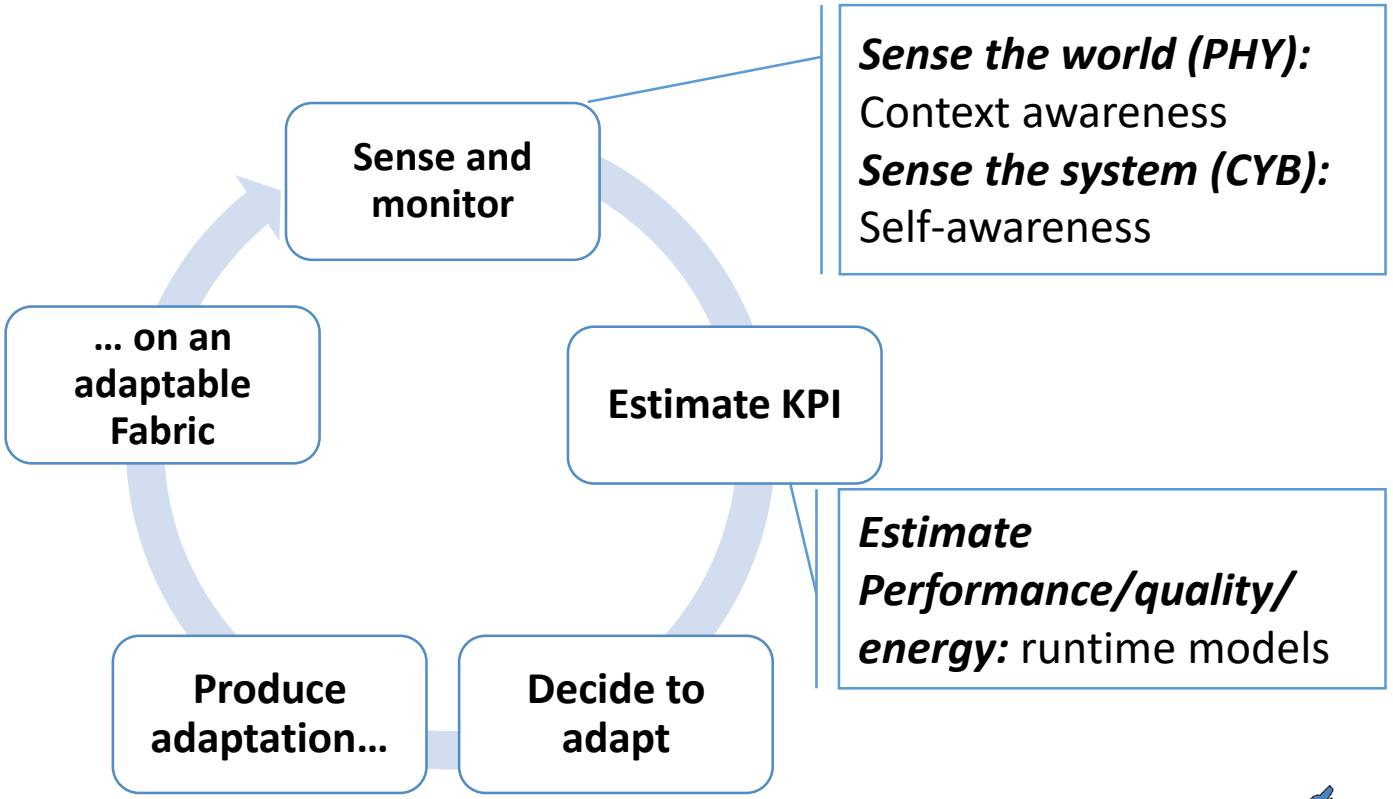


CERBERO Adaptation Loop Formalization

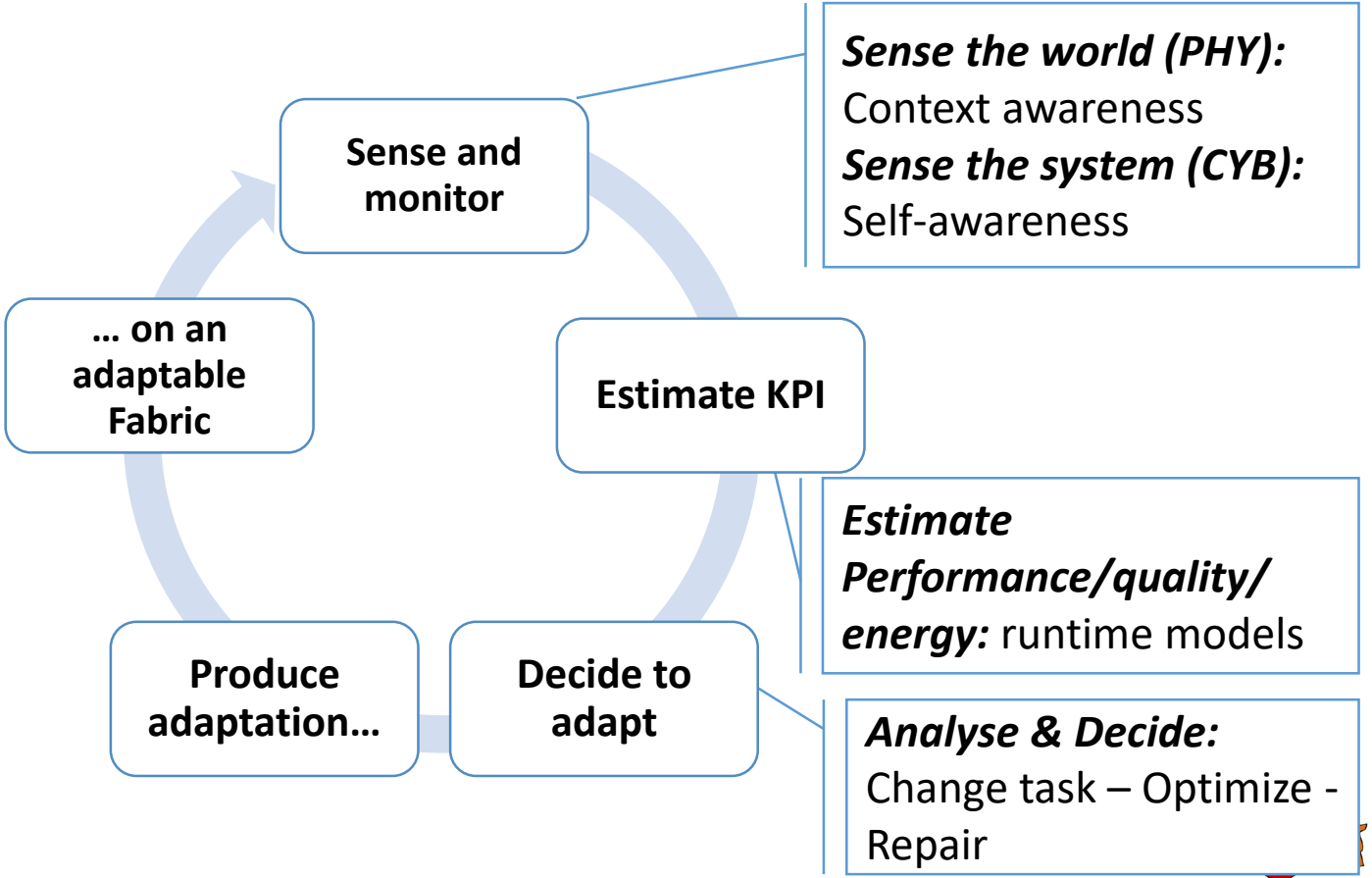


Sense the world (PHY):
Context awareness
Sense the system (CYB):
Self-awareness

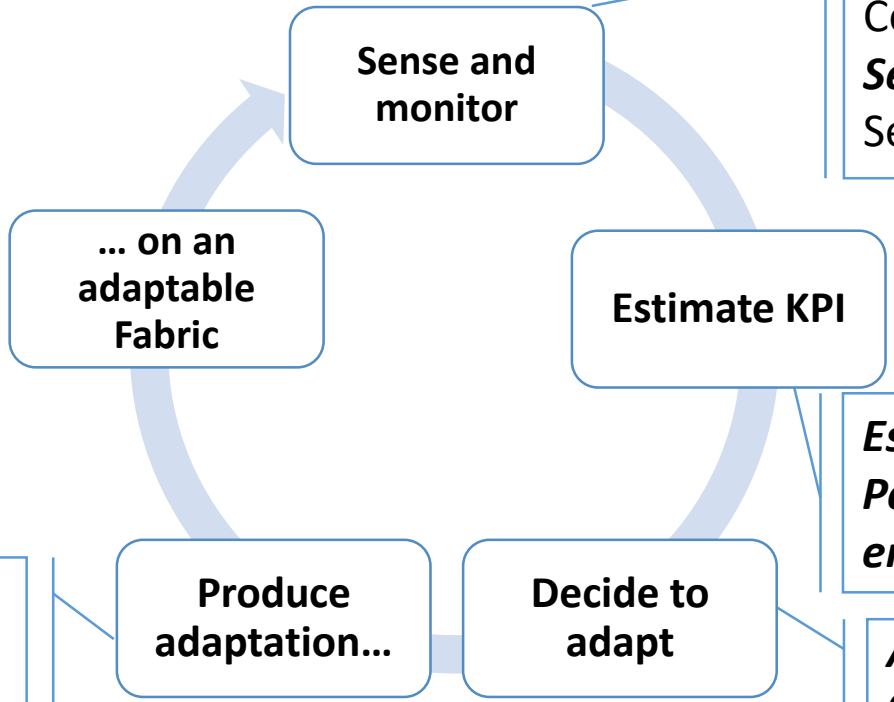
CERBERO Adaptation Loop Formalization



CERBERO Adaptation Loop Formalization



CERBERO Adaptation Loop Formalization



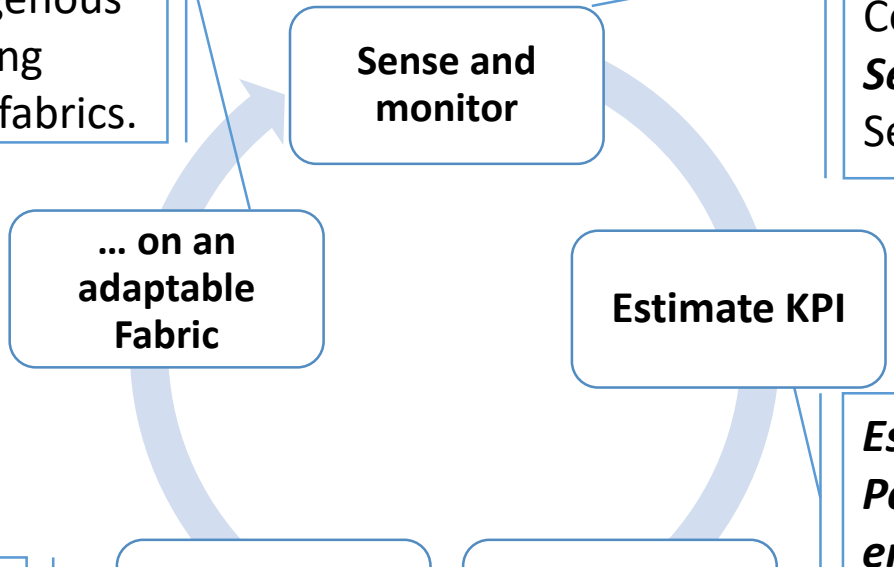
Sense the world (PHY): Context awareness
Sense the system (CYB): Self-awareness

Estimate Performance/quality/ energy: runtime models

Analyse & Decide: Change task – Optimize - Repair

Command Adaptation: Put in place the actions for the required adaptation

CERBERO Adaptation Loop Formalization



Sense the world (PHY): Context awareness
Sense the system (CYB): Self-awareness

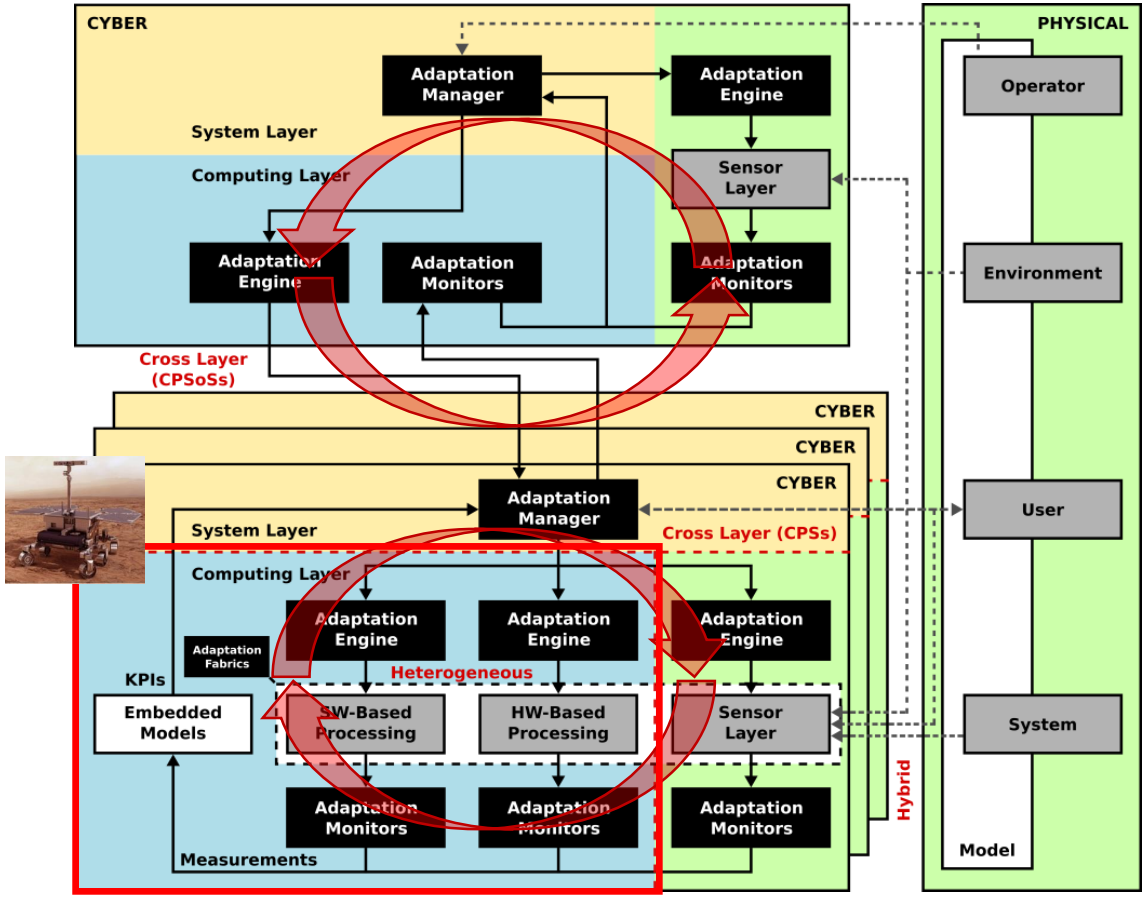
Estimate Performance/quality/ energy: runtime models

Analyse & Decide: Change task – Optimize - Repair

Command Adaptation: Put in place the actions for the required adaptation

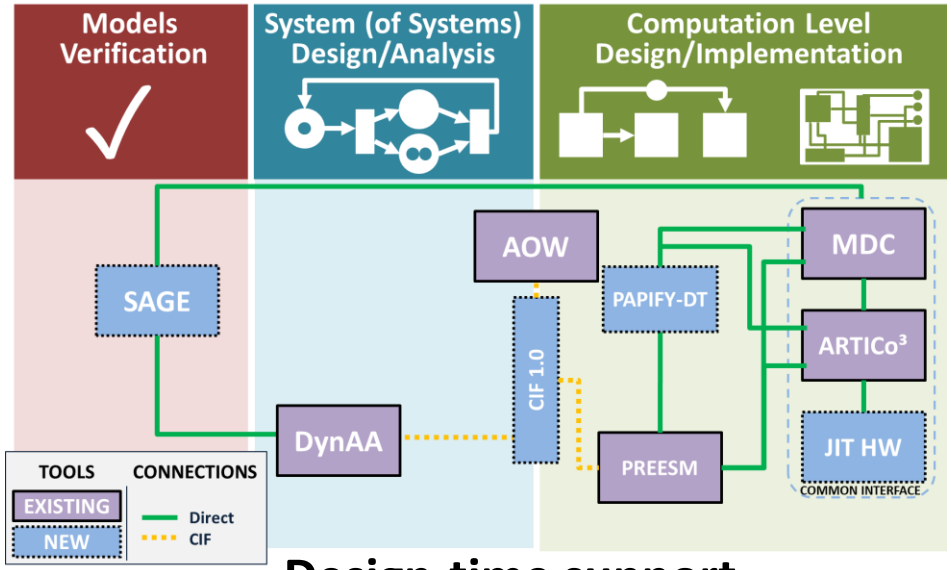
Adapt: Reconfigure the heterogenous and multi-level computing infrastructure. Multiple fabrics.

CERBERO Self-Adaptive Multi-Level Infrastructure

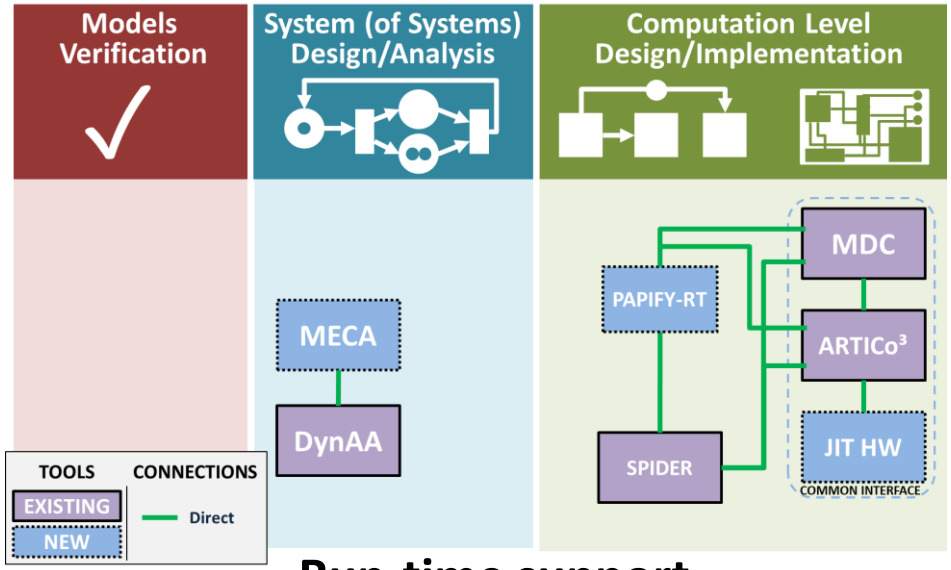


CERBERO

CERBERO Framework

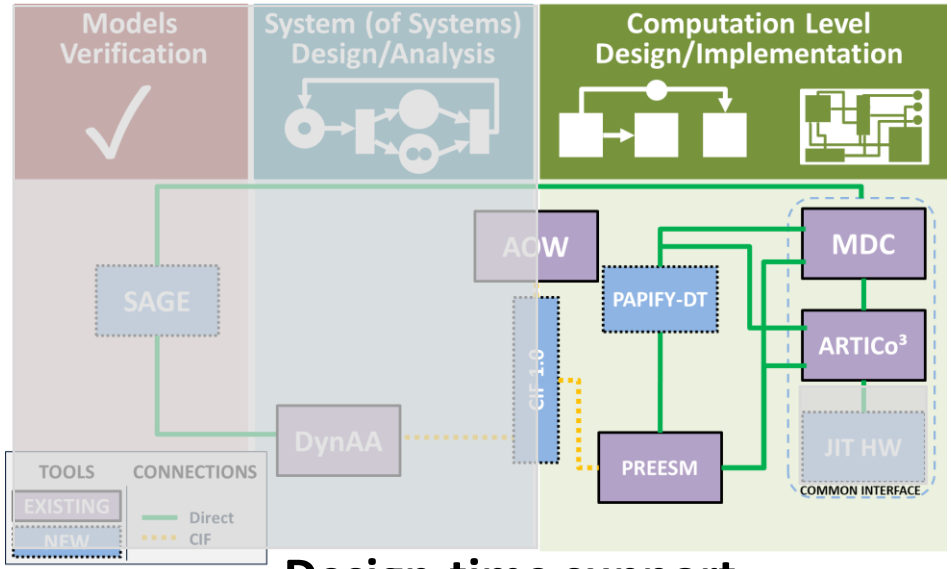


Design-time support

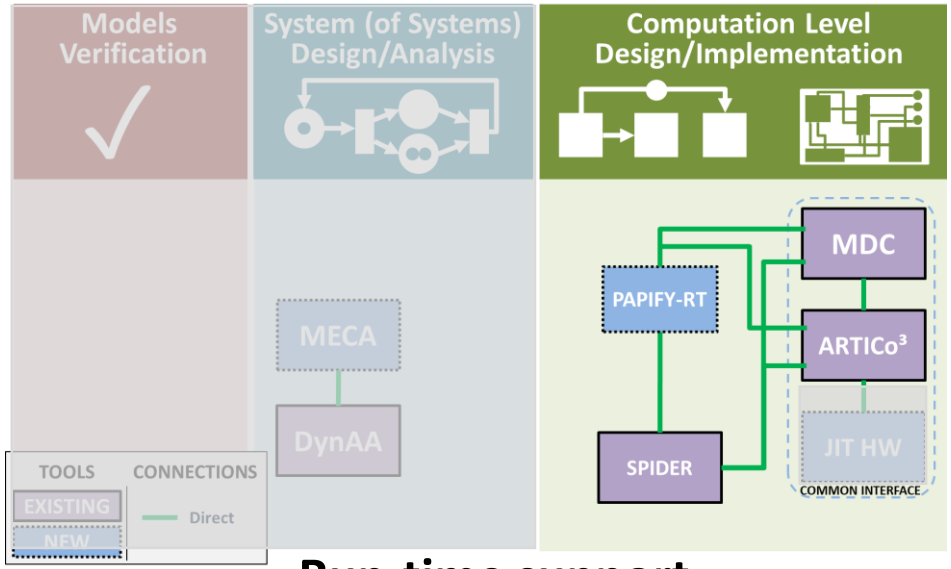


Run-time support

CERBERO Framework

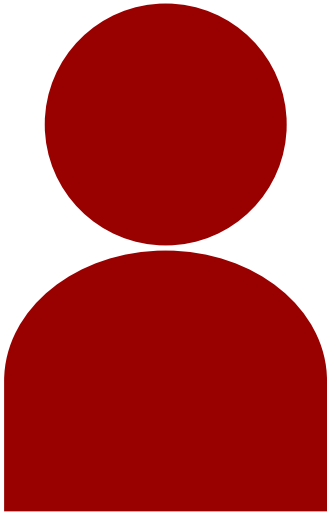


Design-time support



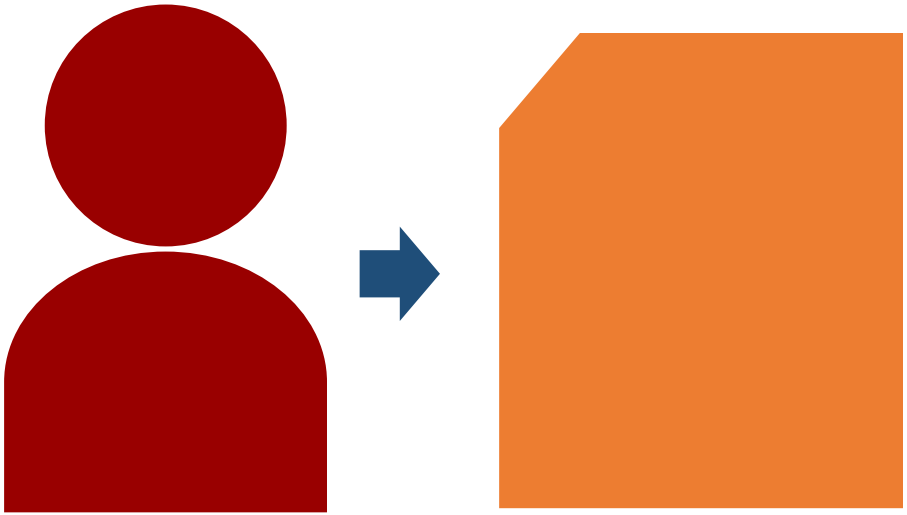
Run-time support

Adaptivity and Heterogeneity for Developers



Developer

Adaptivity and Heterogeneity for Developers

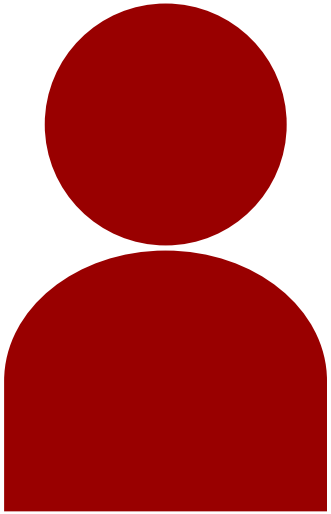


Developer

- SW skills

Application

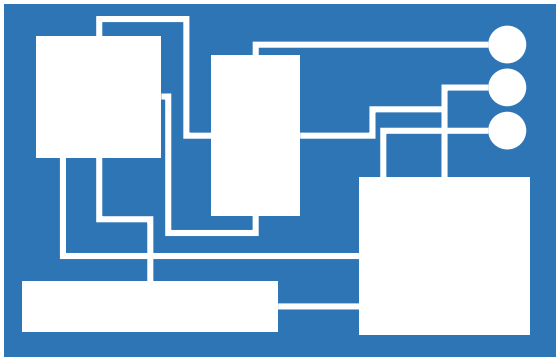
Adaptivity and Heterogeneity for Developers



Developer

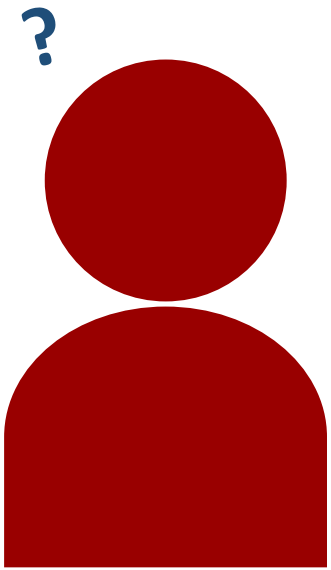
- SW skills
- HW notions

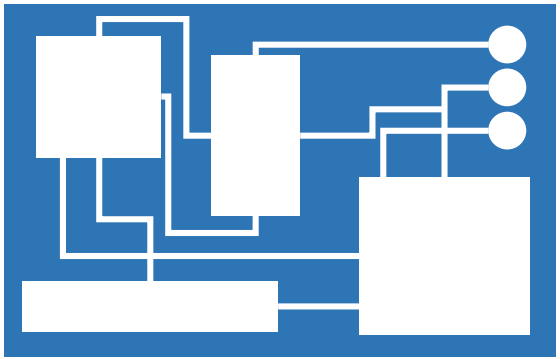
Application



Computing System

Adaptivity and Heterogeneity for Developers





Developer

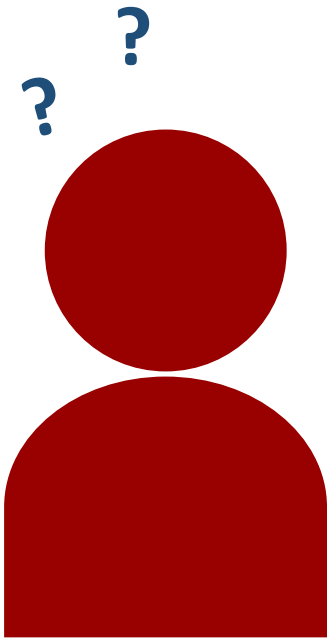
- SW skills
- HW notions

Application

Computing System

CERBERO

Adaptivity and Heterogeneity for Developers

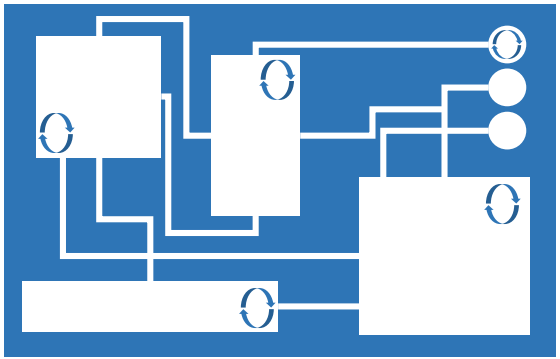


Developer

- SW skills
- HW notions

Application

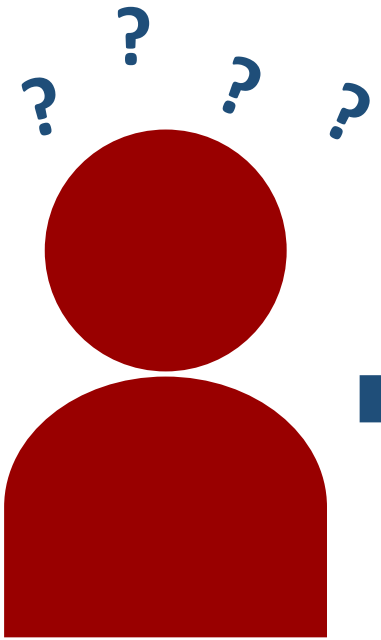
- Adaptivity



Computing System

CERBERO

Adaptivity and Heterogeneity for Developers

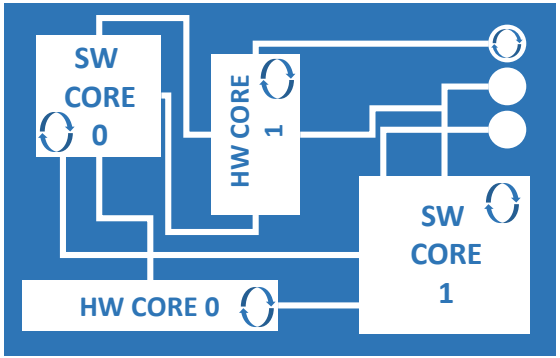


Developer

- SW skills
- HW notions

Application

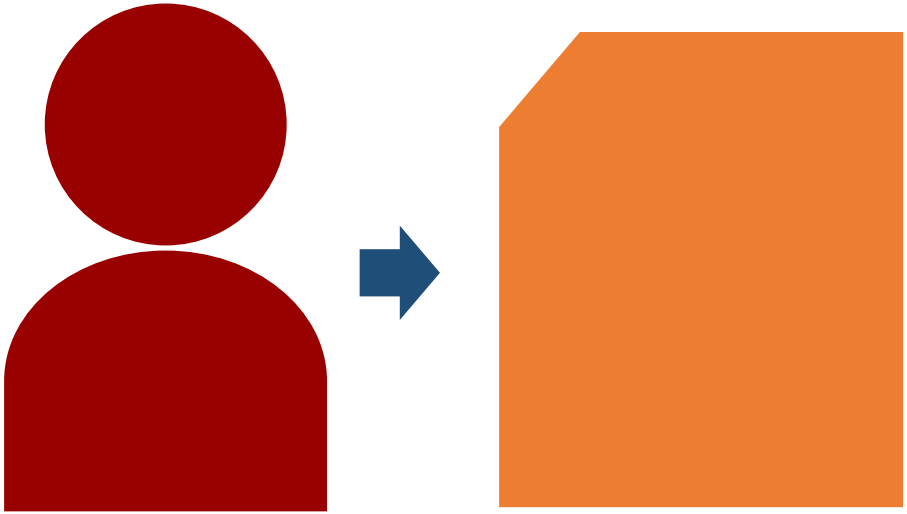
- Adaptivity
- Heterogeneity



Computing System

CERBERO

CERBERO Toolchain Solution

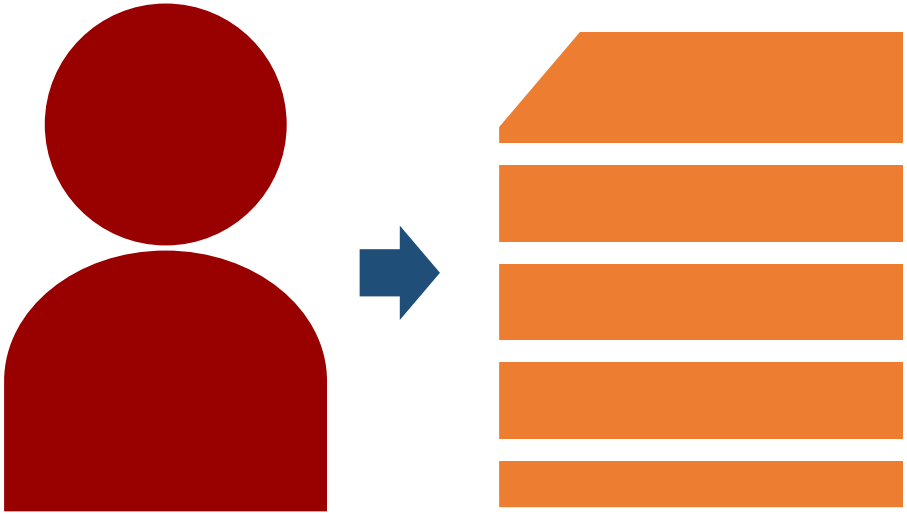


Developer

- SW skills
- HW notions

Application

CERBERO Toolchain Solution

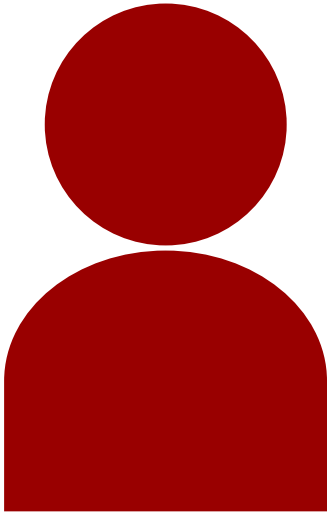


Developer

- SW skills
- HW notions

Application

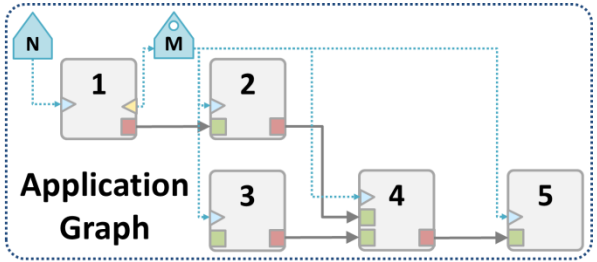
CERBERO Toolchain Solution



Developer

- SW skills
- HW notions

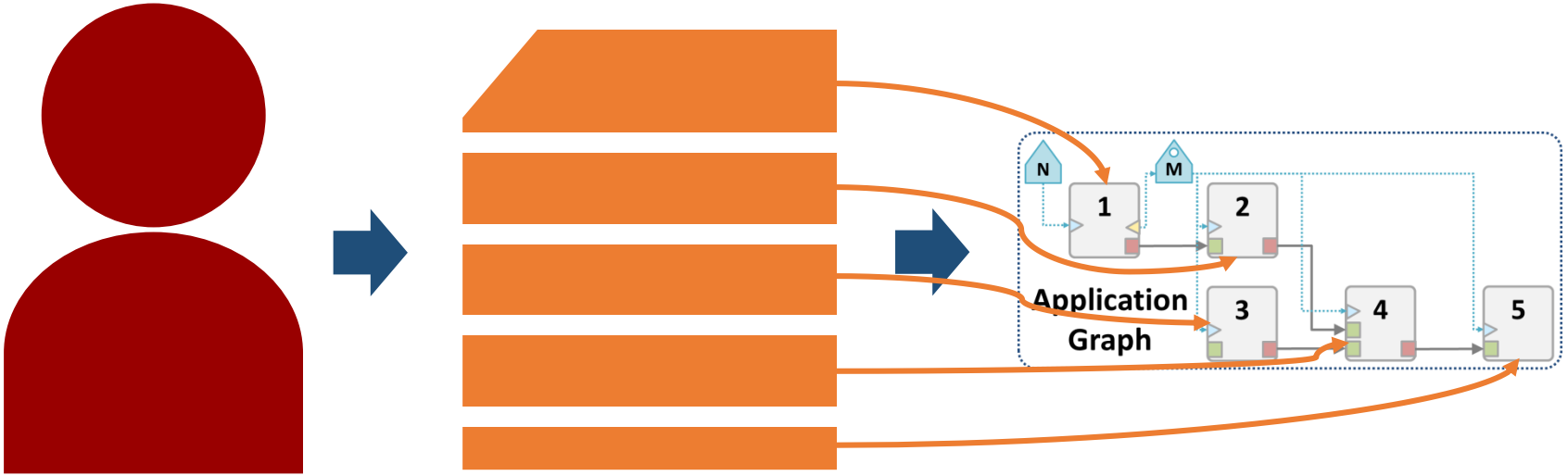
Application



Dataflow Application

CERBERO

CERBERO Toolchain Solution



Developer

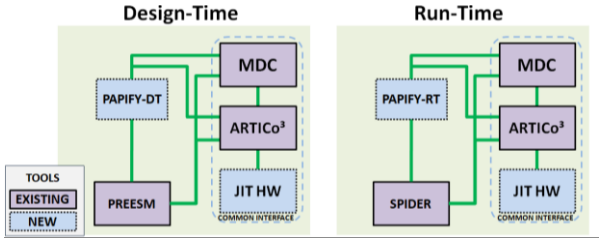
- SW skills
- HW notions

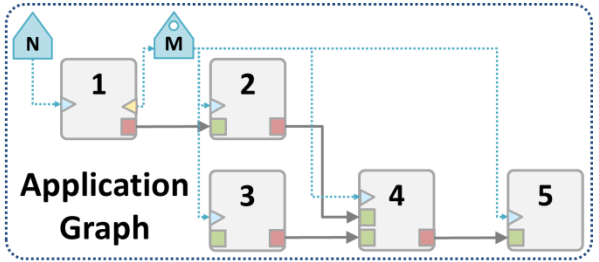
Application

Dataflow Application

CERBERO

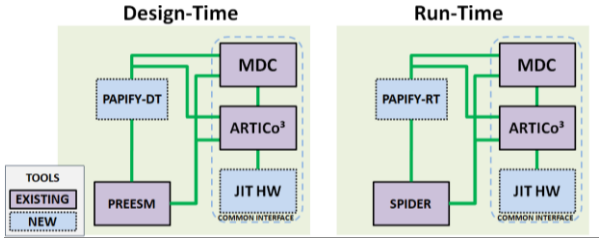
CERBERO Toolchain Solution

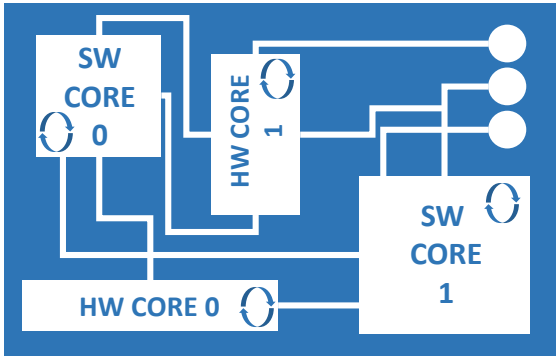




Dataflow Application

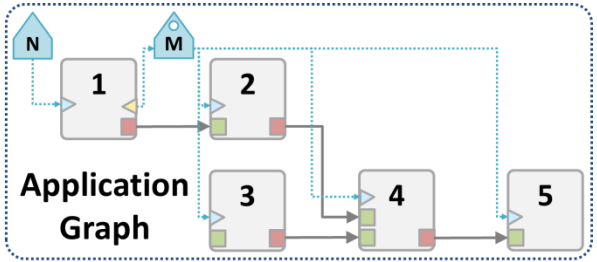
CERBERO Toolchain Solution





Computing System

- Heterogeneity
- Adaptivity

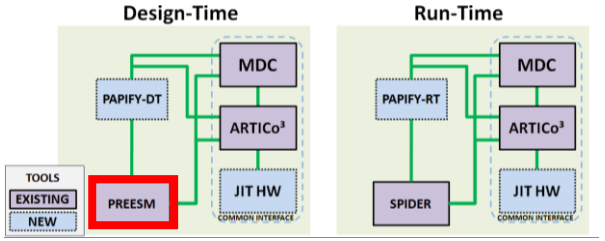


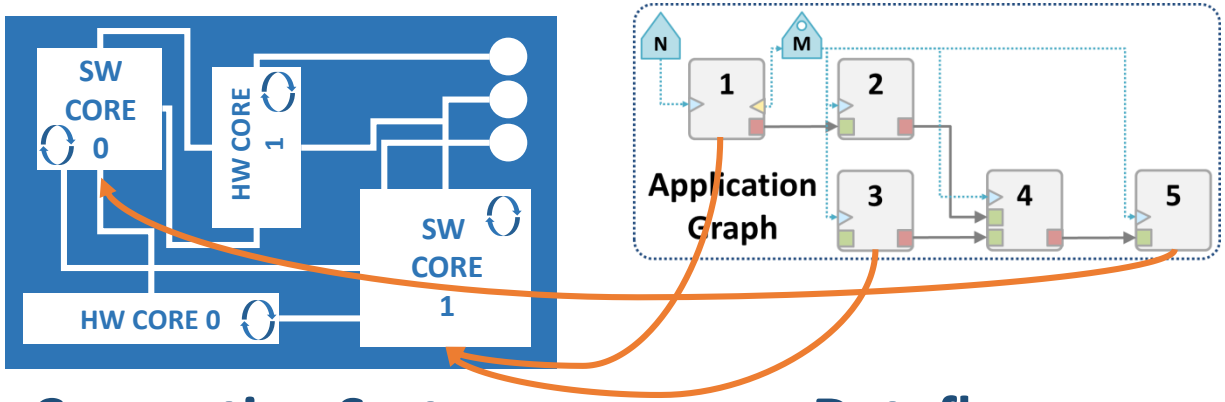
Dataflow Application

CERBERO

CERBERO Toolchain Solution

- Parallel SW Code Generation and Mapping





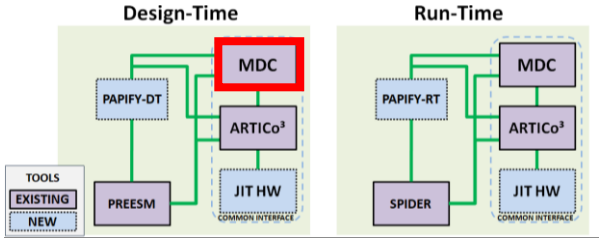
Computing System

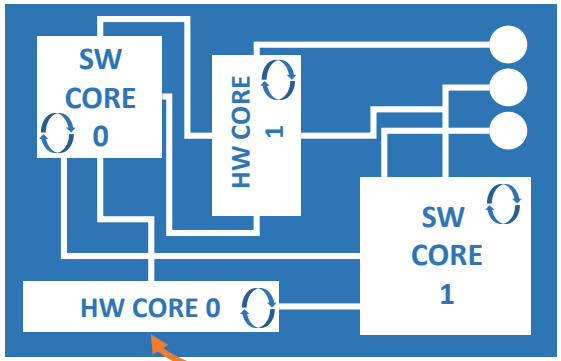
- Heterogeneity
- Adaptivity

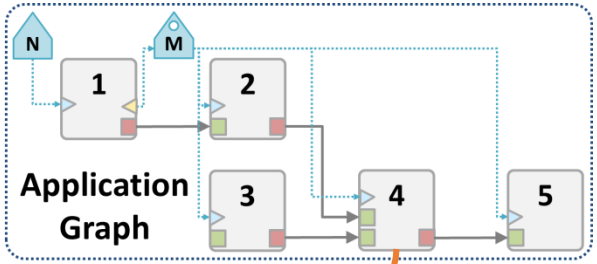
Dataflow Application

CERBERO Toolchain Solution

- Parallel SW Code Generation and Mapping
- Deployment of HW CGR Accelerator (fast reconf.)







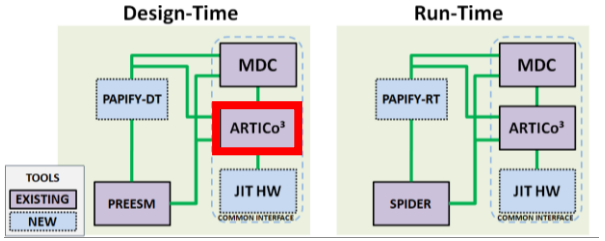
Computing System

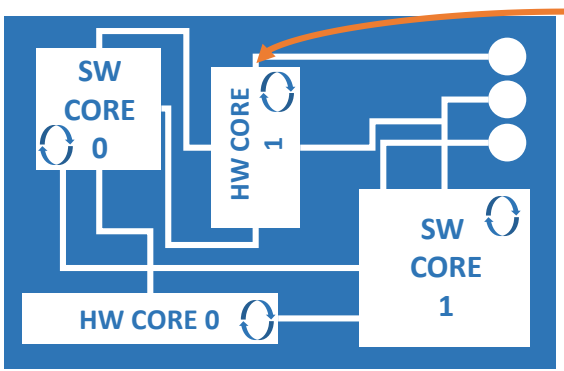
- Heterogeneity
- Adaptivity

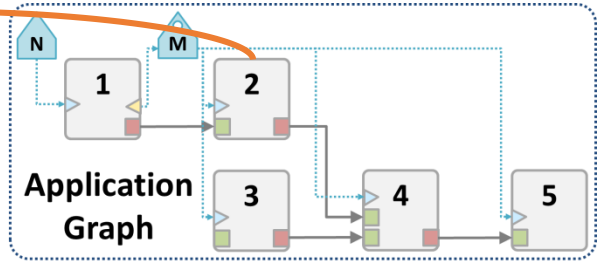
Dataflow Application

CERBERO Toolchain Solution

- Parallel SW Code Generation and Mapping
- Deployment of HW CGR Accelerator (fast reconf.)
- Deployment of HW DPR Multi-Accelerator (slow reconf., HW reuse)







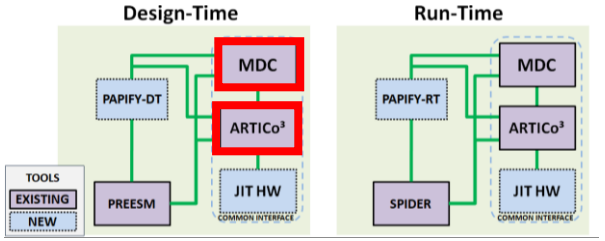
Computing System

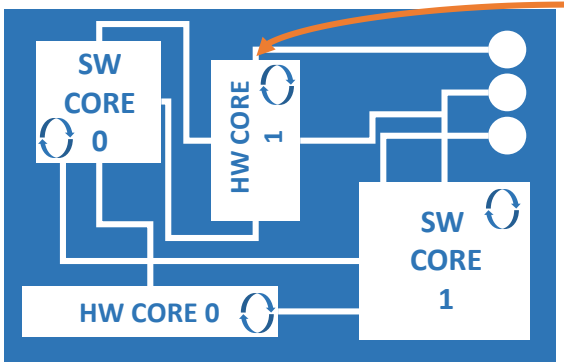
- Heterogeneity
- Adaptivity

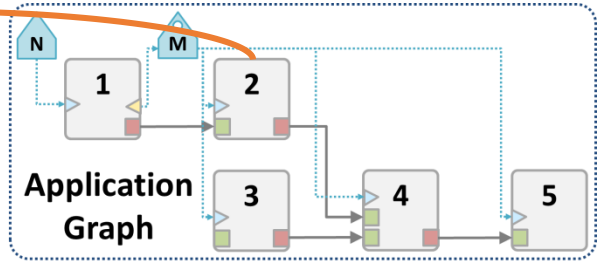
Dataflow Application

CERBERO Toolchain Solution

- Parallel SW Code Generation and Mapping
- Deployment of HW CGR Accelerator (fast reconf.)
- Deployment of HW DPR Multi-Accelerator (slow reconf., HW reuse)
  - Optionally with CGR inside (slow or fast reconf for adaptation)







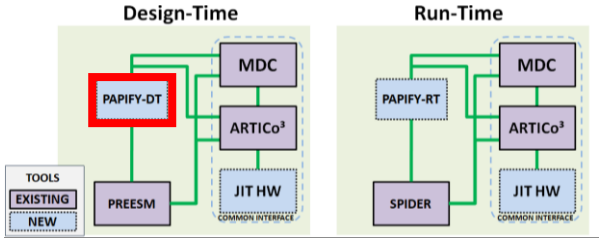
Computing System

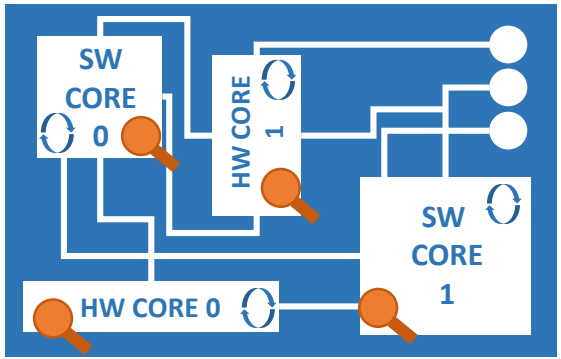
- Heterogeneity
- Adaptivity

Dataflow Application

CERBERO Toolchain Solution

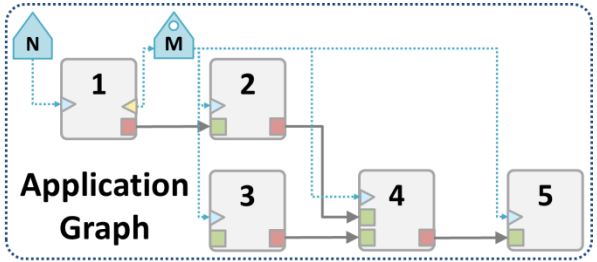
- Parallel SW Code Generation and Mapping
- Deployment of HW CGR Accelerator (fast reconf.)
- Deployment of HW DPR Multi-Accelerator (slow reconf., HW reuse)
  - Optionally with CGR inside (slow or fast reconf for adaptation)
- Instrumentation for monitoring





Computing System

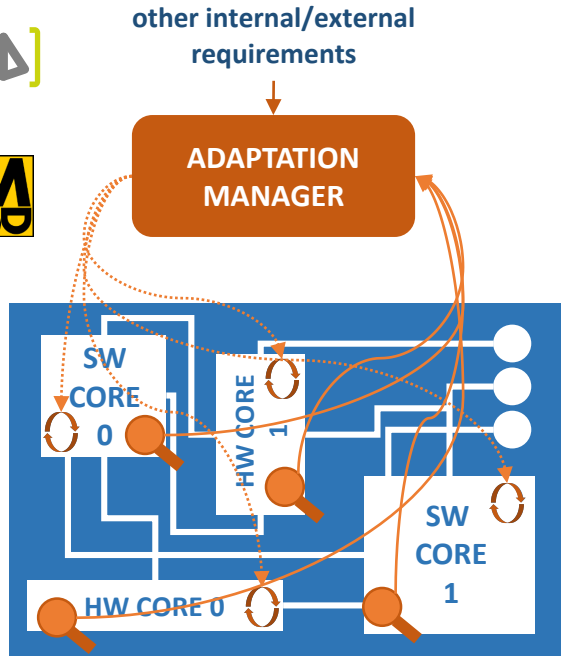
- Heterogeneity
- Adaptivity



Dataflow Application

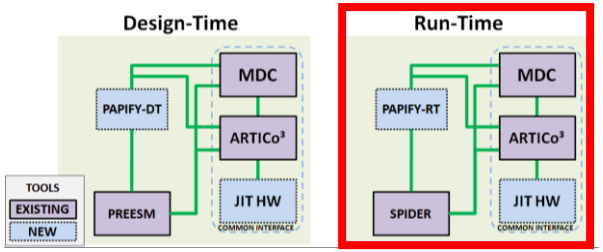
CERBERO Toolchain Solution

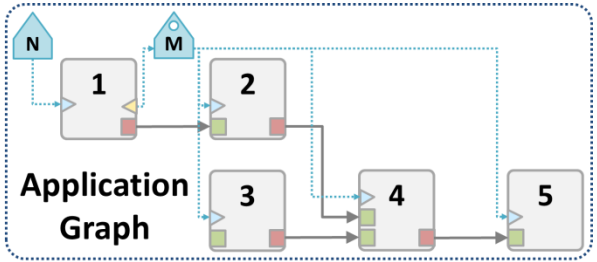
- Parallel SW Code Generation and Mapping
- Deployment of HW CGR Accelerator (fast reconf.)
- Deployment of HW DPR Multi-Accelerator (slow reconf., HW reuse)
  - Optionally with CGR inside (slow or fast reconf for adaptation)
- Instrumentation for monitoring
- Run-time management and self-adaptation (not covered in the tutorial)



Computing System

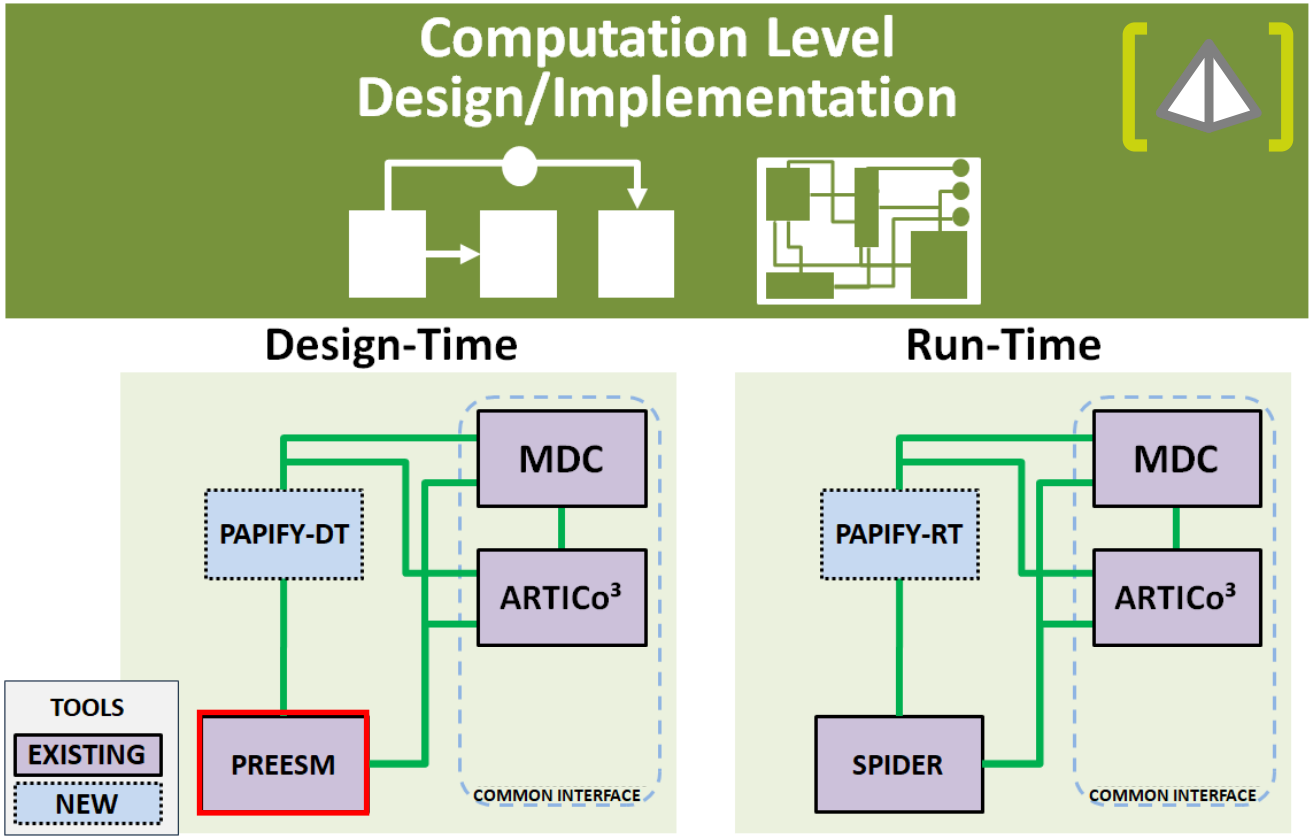
- Heterogeneity
- Adaptivity





Dataflow Application

CERBERO Framework - Computation Level



PREESM

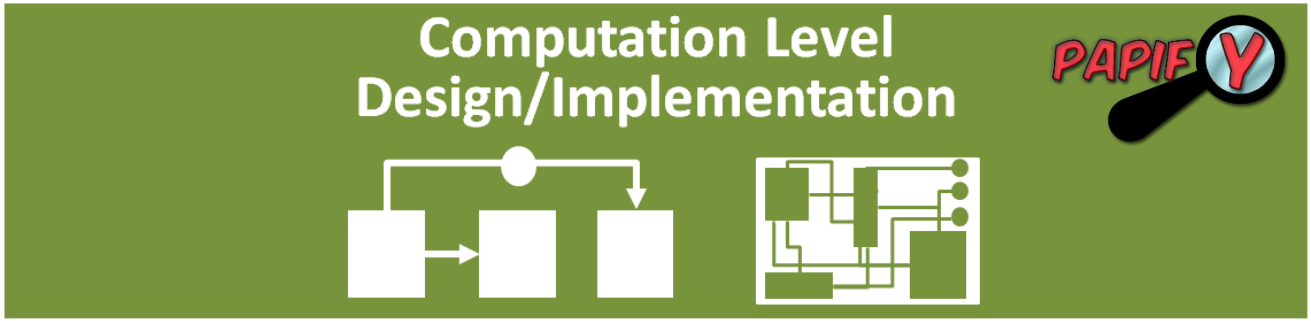
Design-time

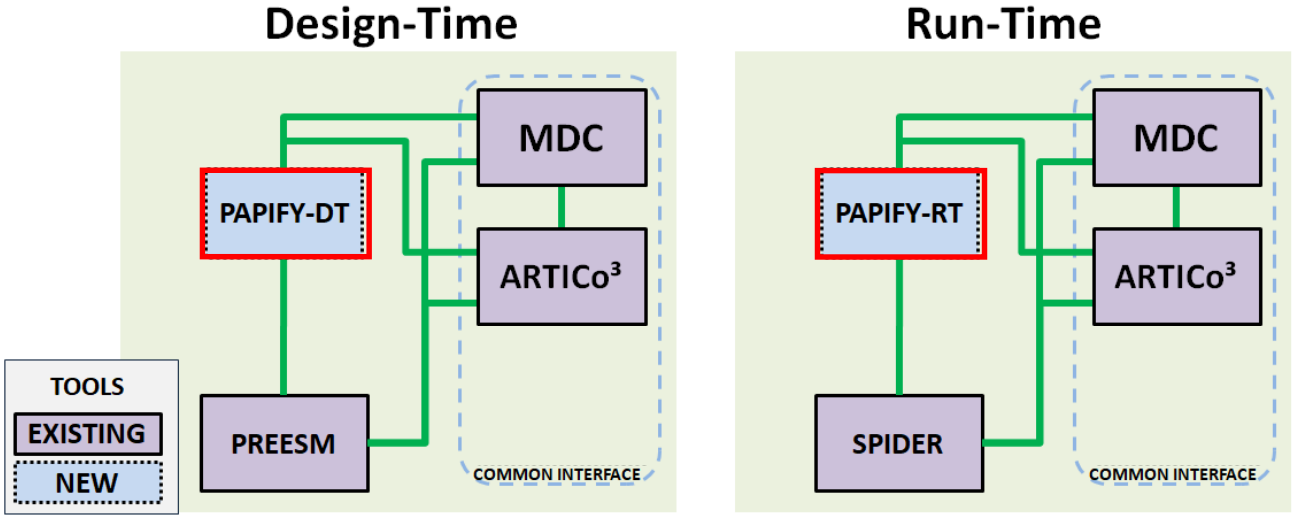
Parallel-application development. Code generation and re-use capabilities.

INSA

CERBERO

CERBERO Framework - Computation Level





PAPIFY

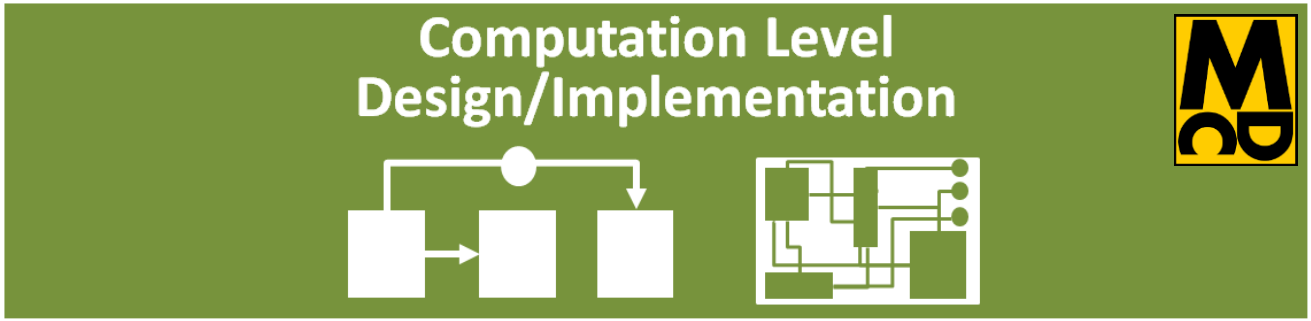
Design-time

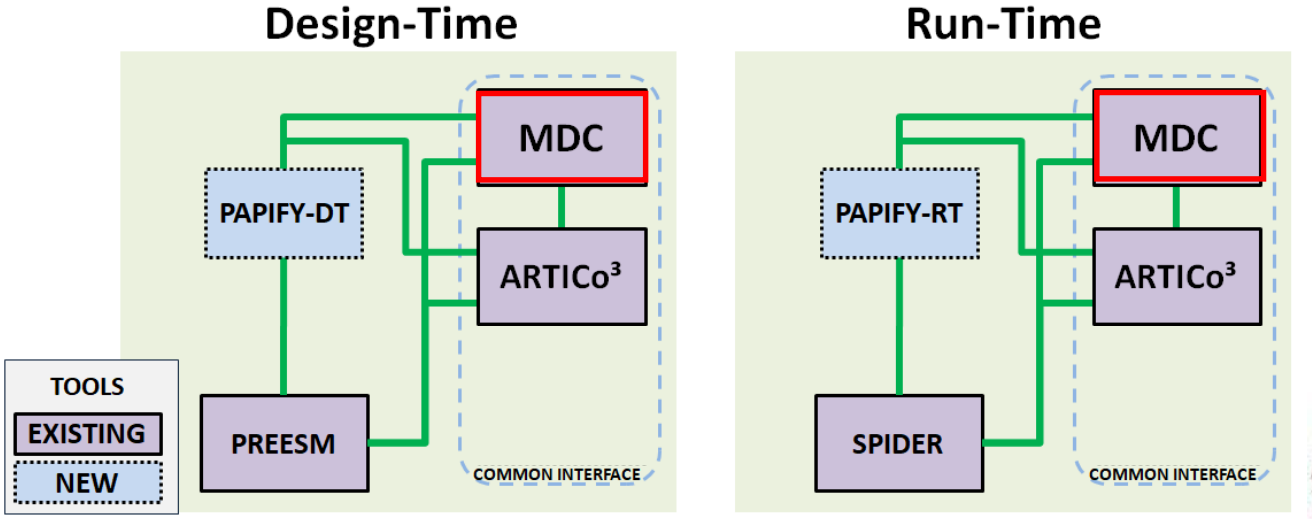
Run-time

Automatic code instrumentation for performance monitoring

Provides a large set of run-time execution information to SPIDER

CERBERO Framework - Computation Level





MDC

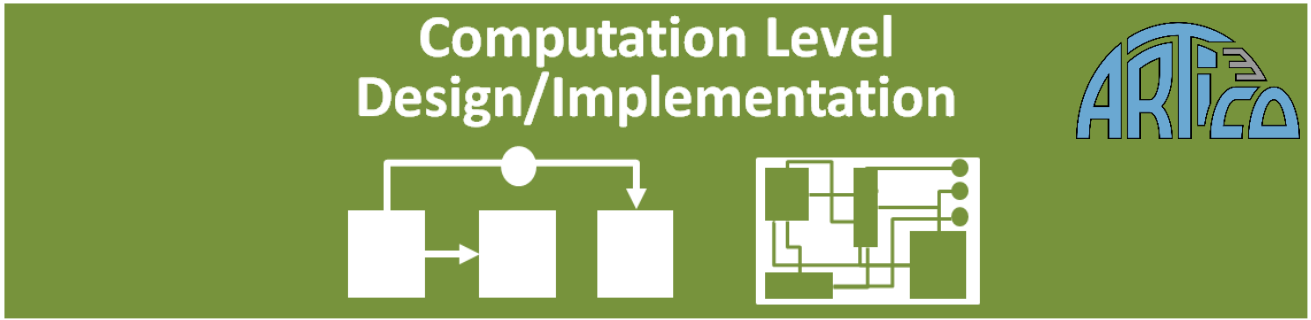
Design-time

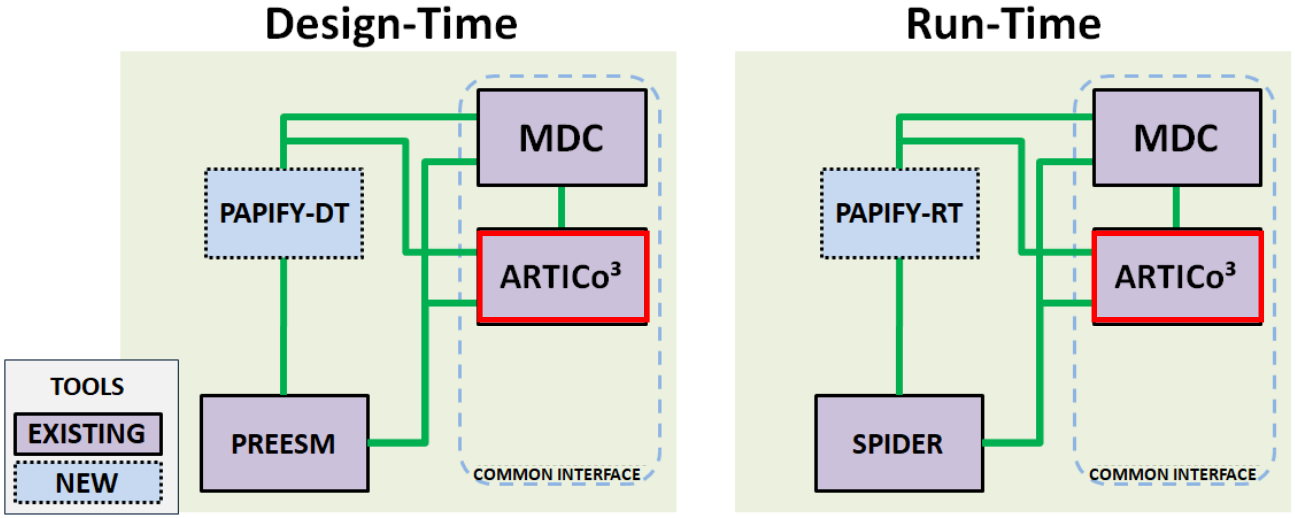
Run-time

Dataflow-to-hardware framework for generation of CGR accelerators

Fast HW reconfiguration

CERBERO Framework - Computation Level





ARTICo³

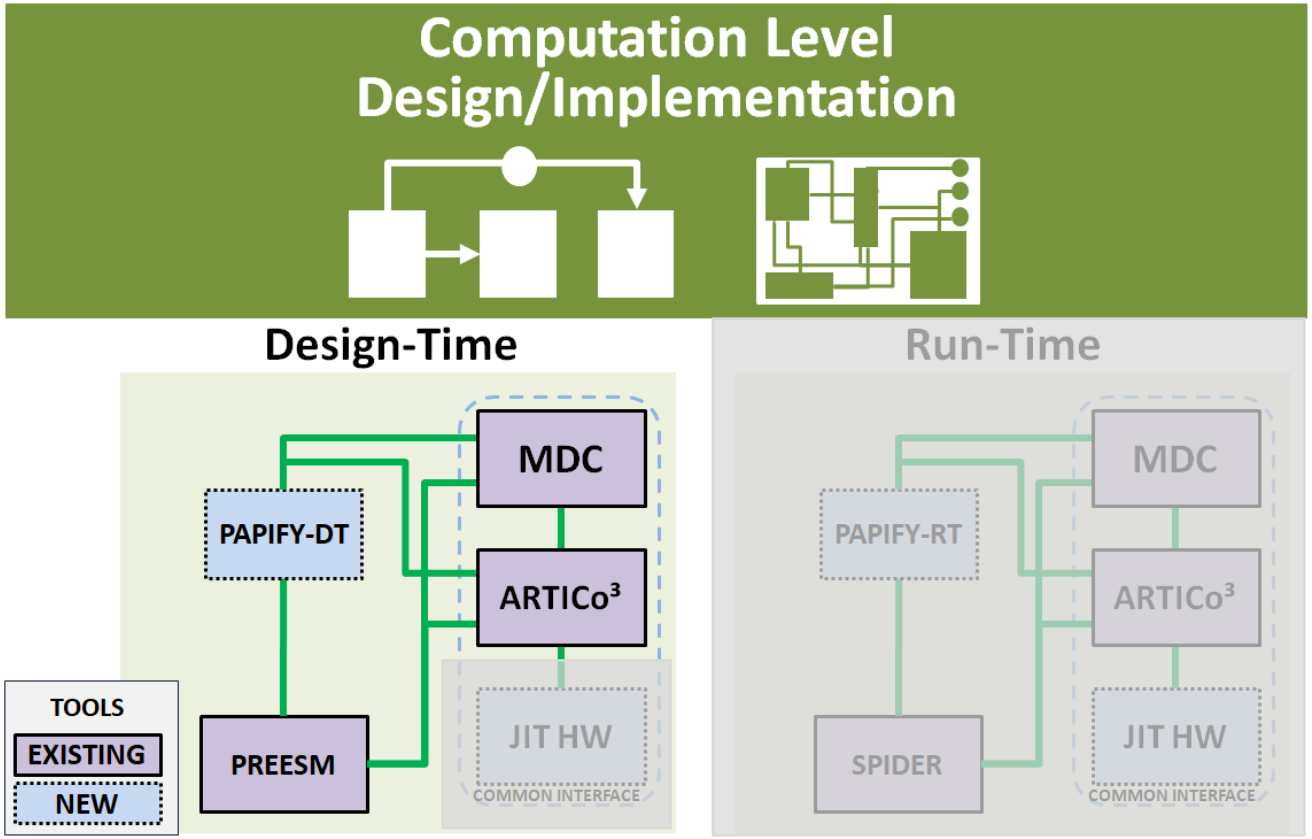
Design-time

Run-time

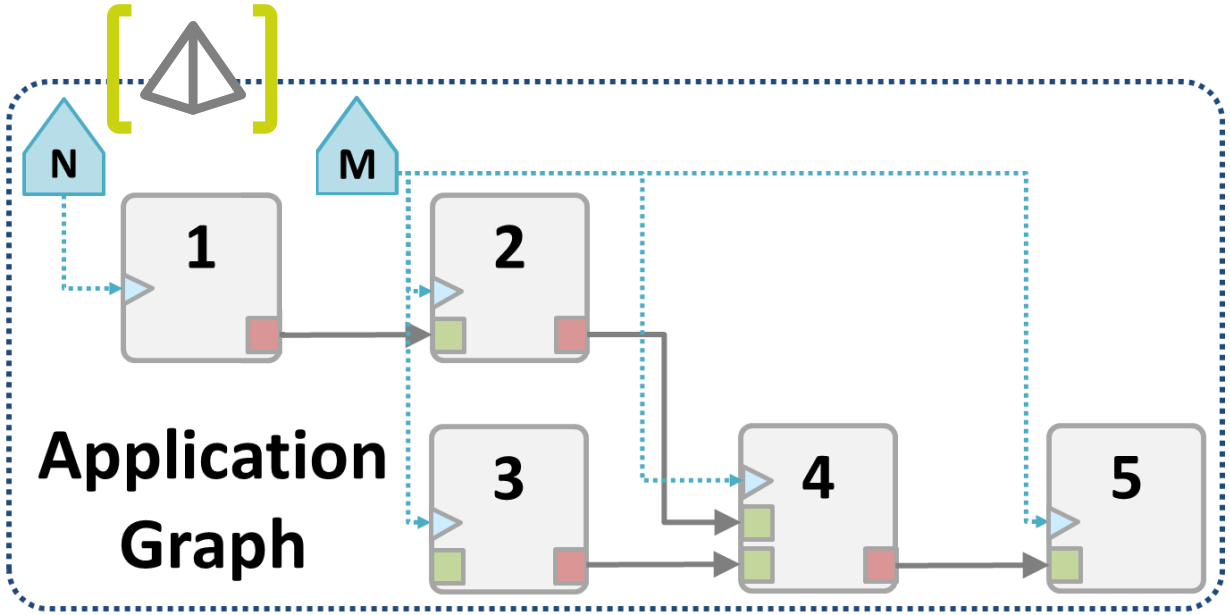
DPR-enabled multi-accelerator

Scalable parallelism and fault tolerance

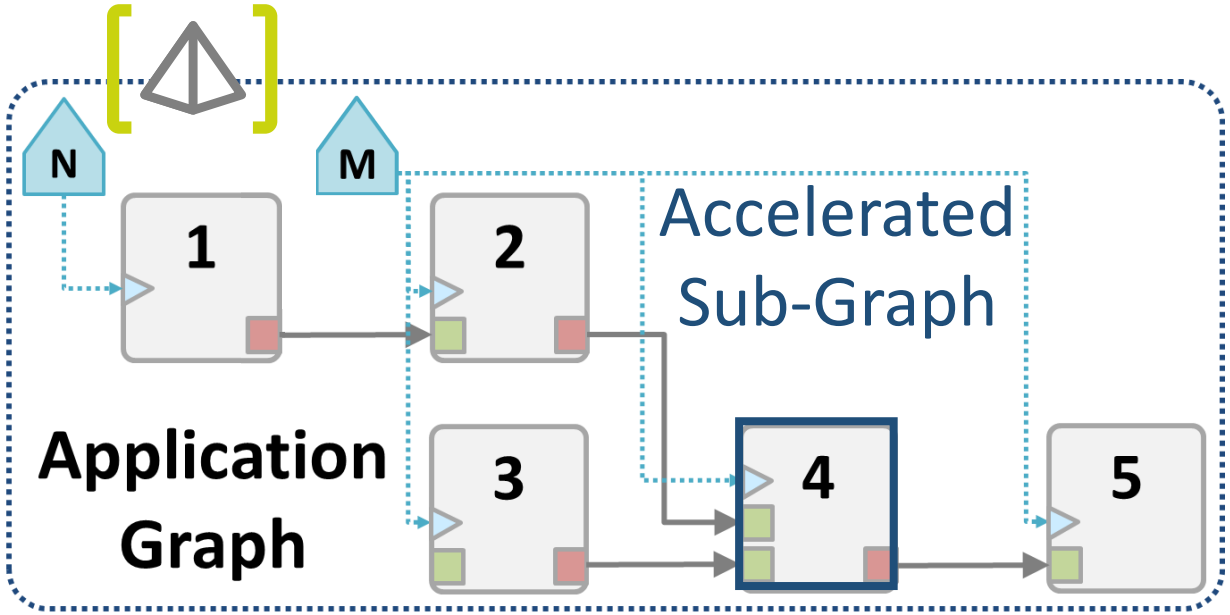
CERBERO Framework – Self-Adaptive System Design



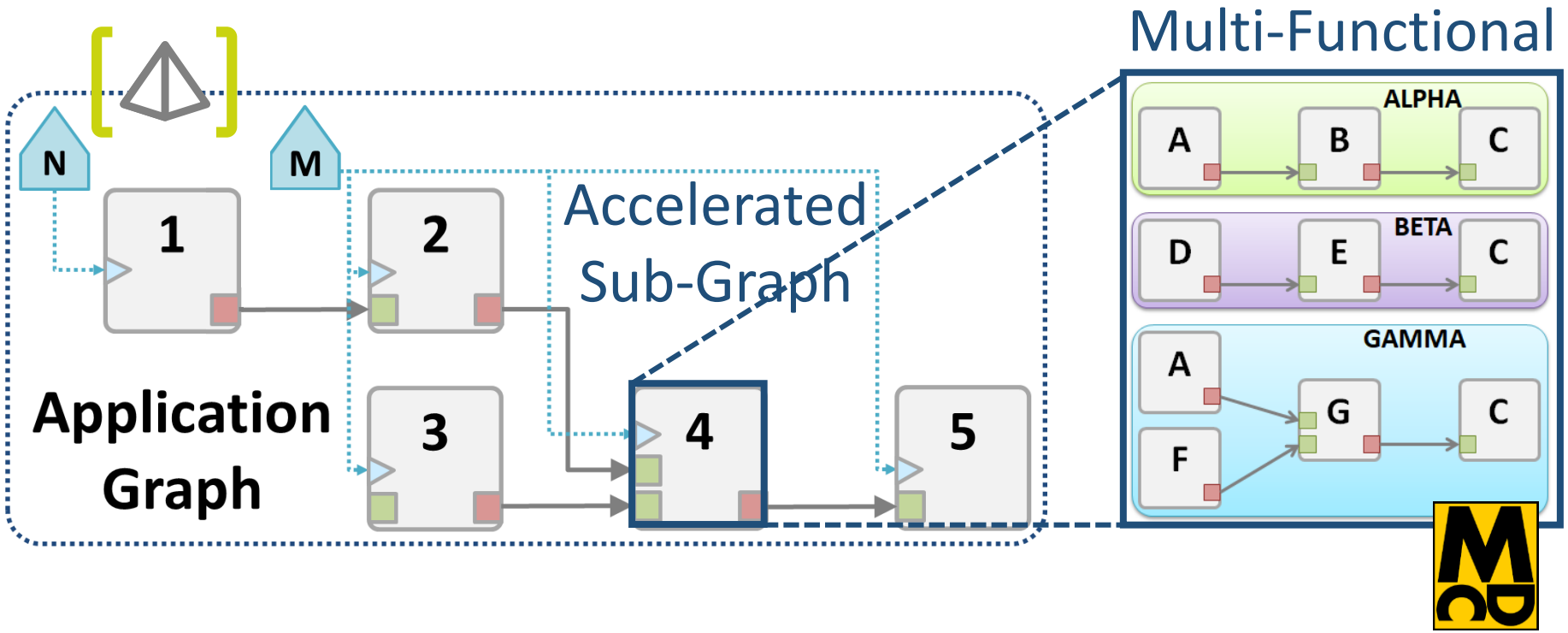
Design-Time Support: for HW Run-Time Adaptivity



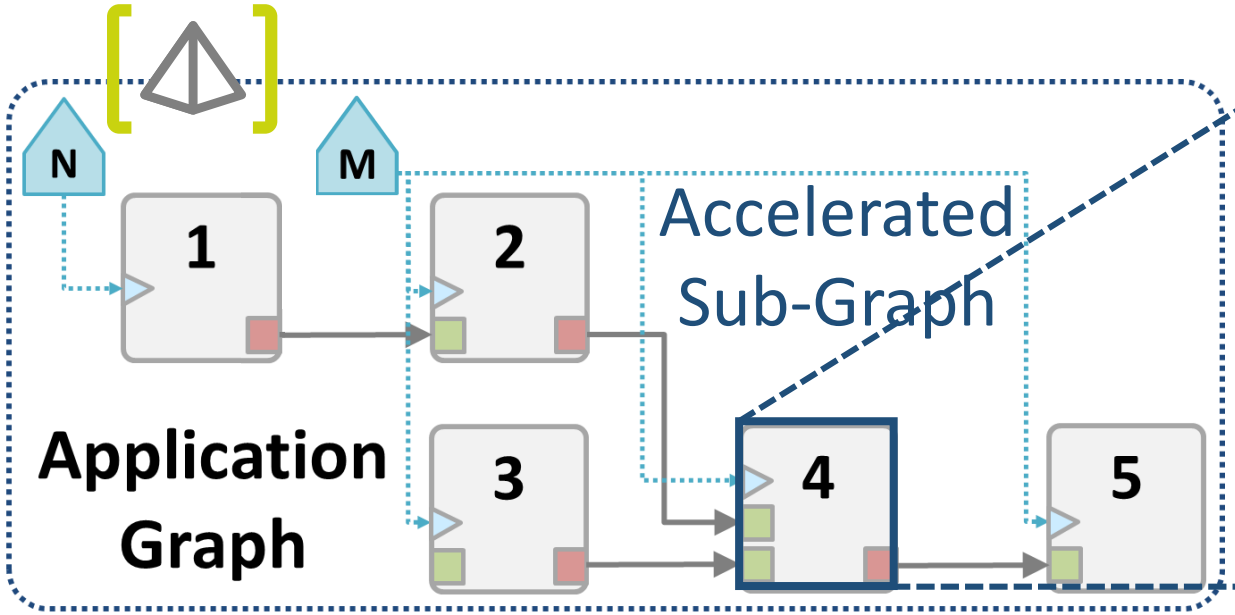
Design-Time Support: for HW Run-Time Adaptivity



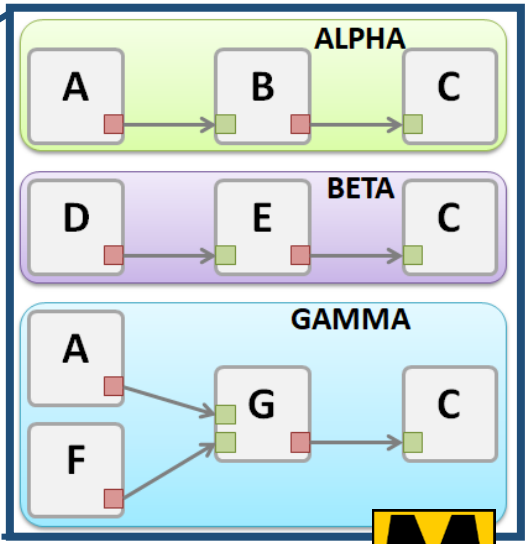
Design-Time Support: for HW Run-Time Adaptivity



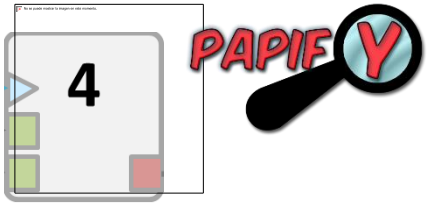
Design-Time Support: for HW Run-Time Adaptivity



Multi-Functional

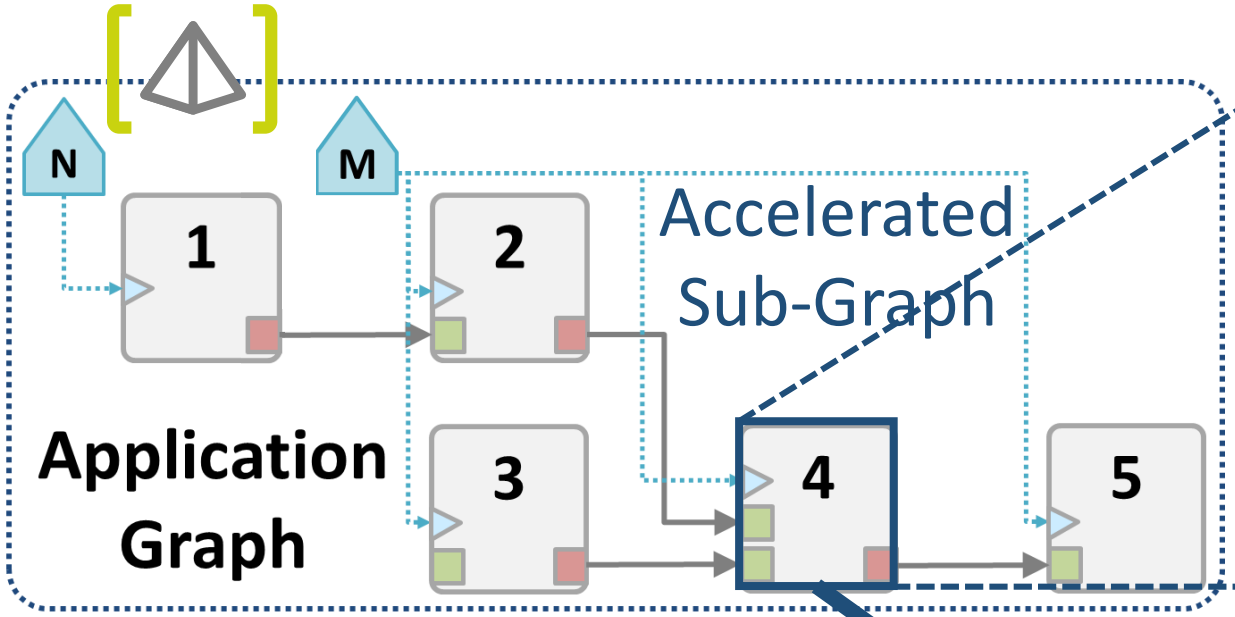


HW/SW Monitoring

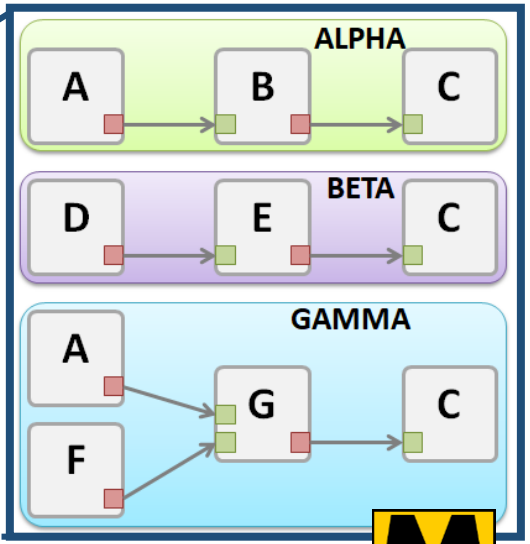


CERBERO

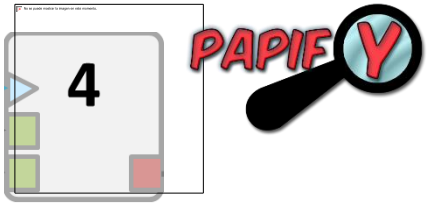
Design-Time Support: for HW Run-Time Adaptivity



Multi-Functional



HW/SW Monitoring



Parallel Execution

CERBERO

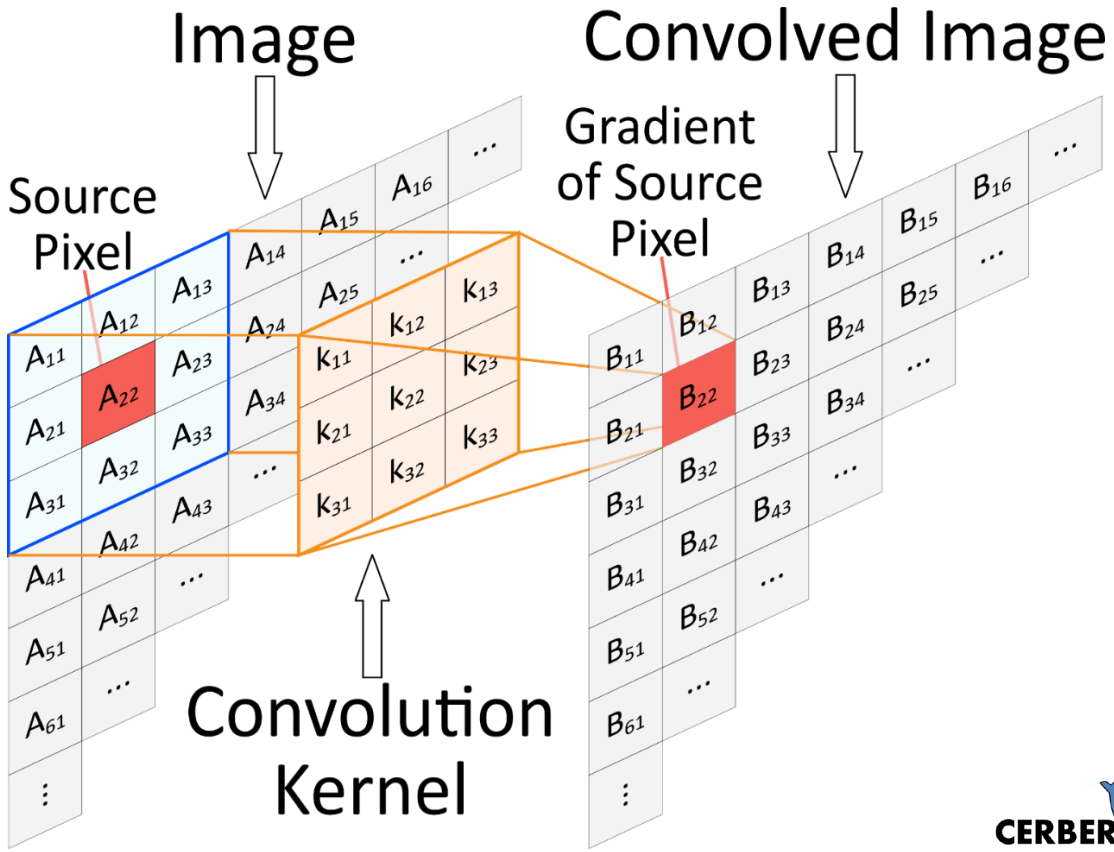
Tutorial Use Case

Sobel Operator

$$\mathbf{G} = \sqrt{\mathbf{G}_x^{\,2} + \mathbf{G}_y^{\,2}}$$

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}$$

$$\mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$



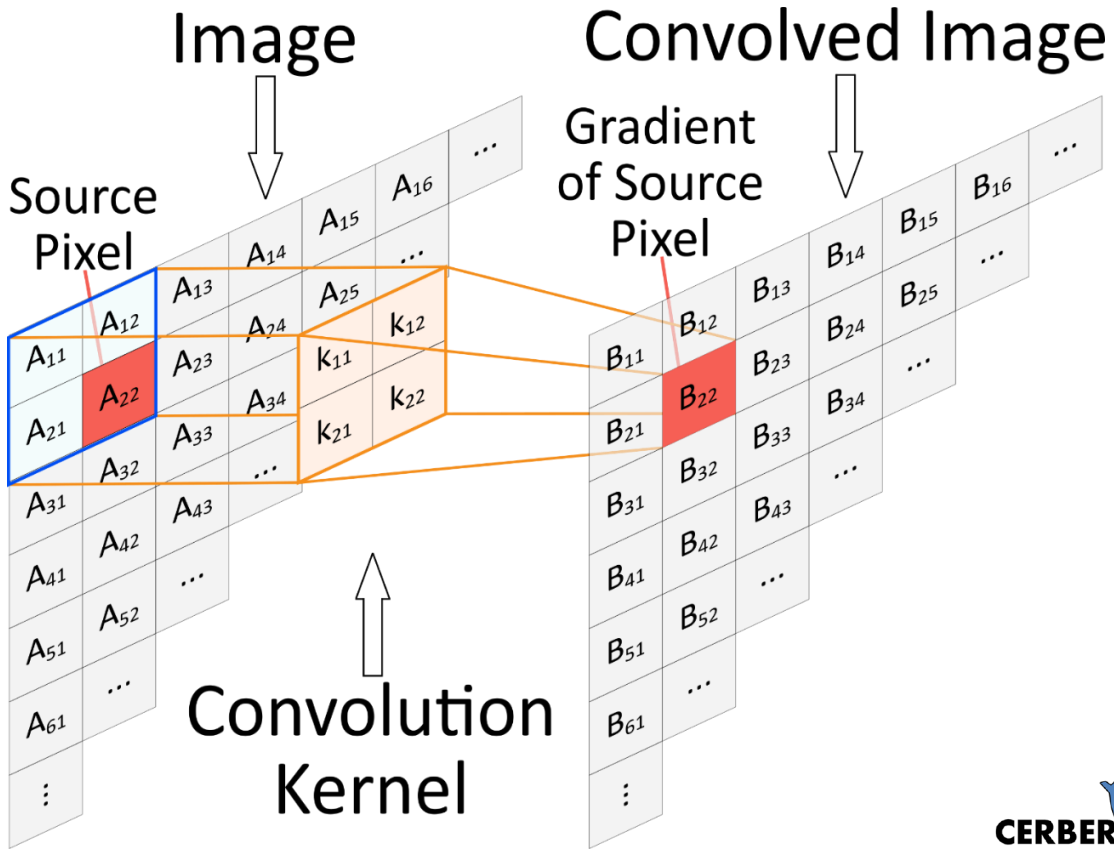
Tutorial Use Case

Roberts Operator

$$\mathbf{G} = \sqrt{\mathbf{G}_x^{\,2} + \mathbf{G}_y^{\,2}}$$

$$G_x = \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix}$$

$$G_y = \begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix}$$



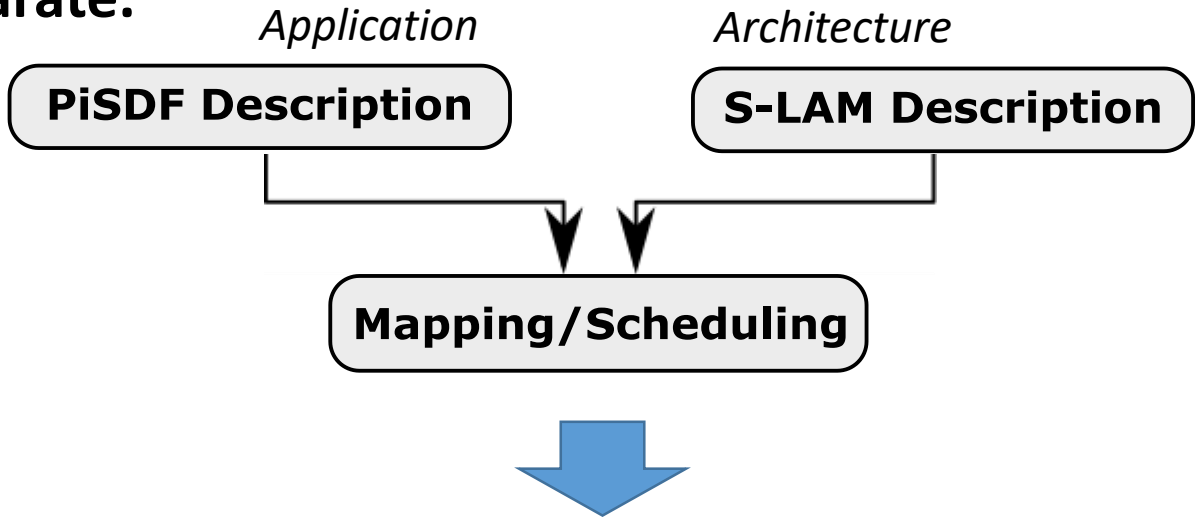
HiPEAC 2020
20-22 January, 2020, Bologna, Italy

Hardware/Software Model-Based Design for Design Space Exploration and Code Generation

Horizon 2020
European Union funding
for Research & Innovation

PREESM

- **Y co-design approach: keep architecture and application separate.**

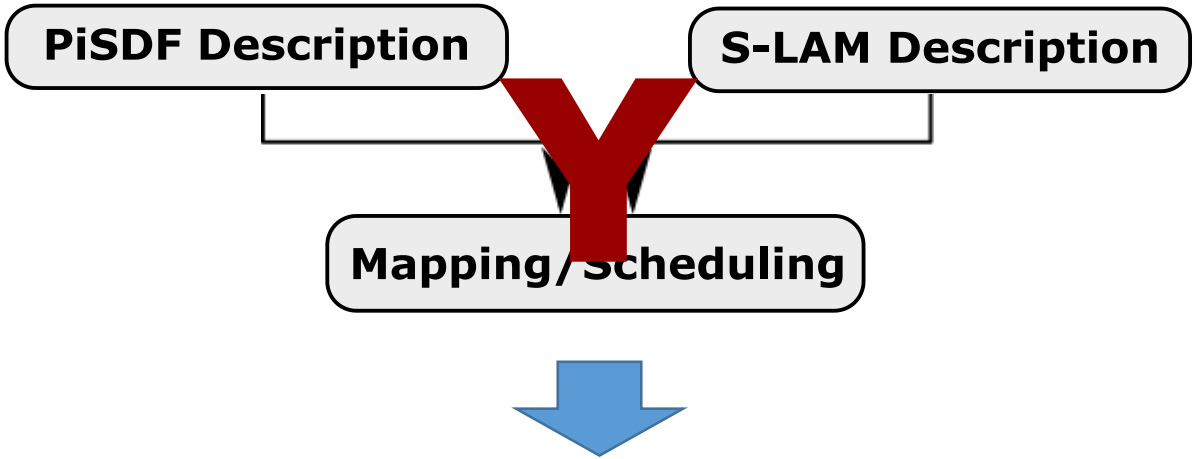


AUTOMATIC CODE GENERATION for RAPID PROTOTYPING

CERBERO

PREESM

- **Y co-design approach: keep architecture and application separate.**

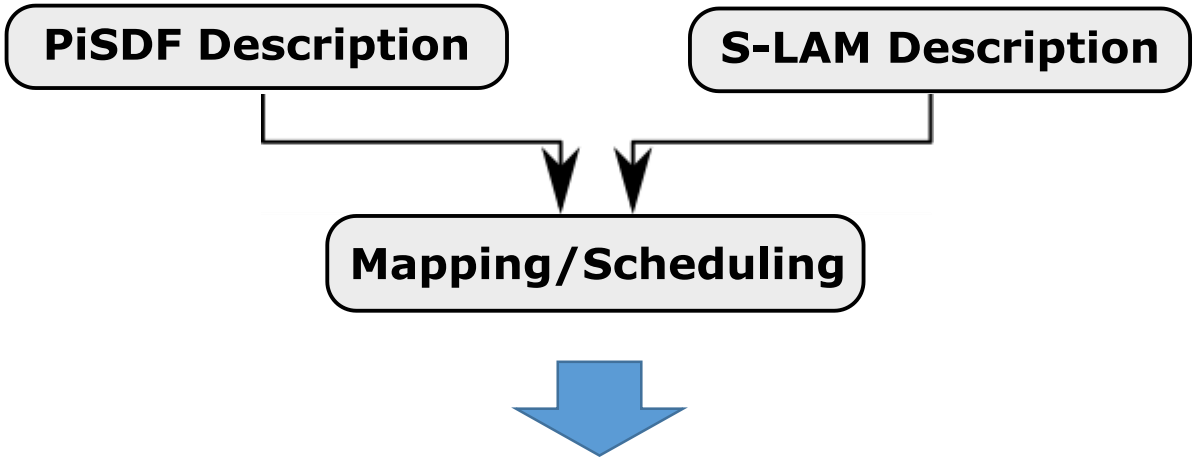


AUTOMATIC CODE GENERATION for RAPID PROTOTYPING

CERBERO

PREESM

- Y co-design approach: keep architecture and application separate.

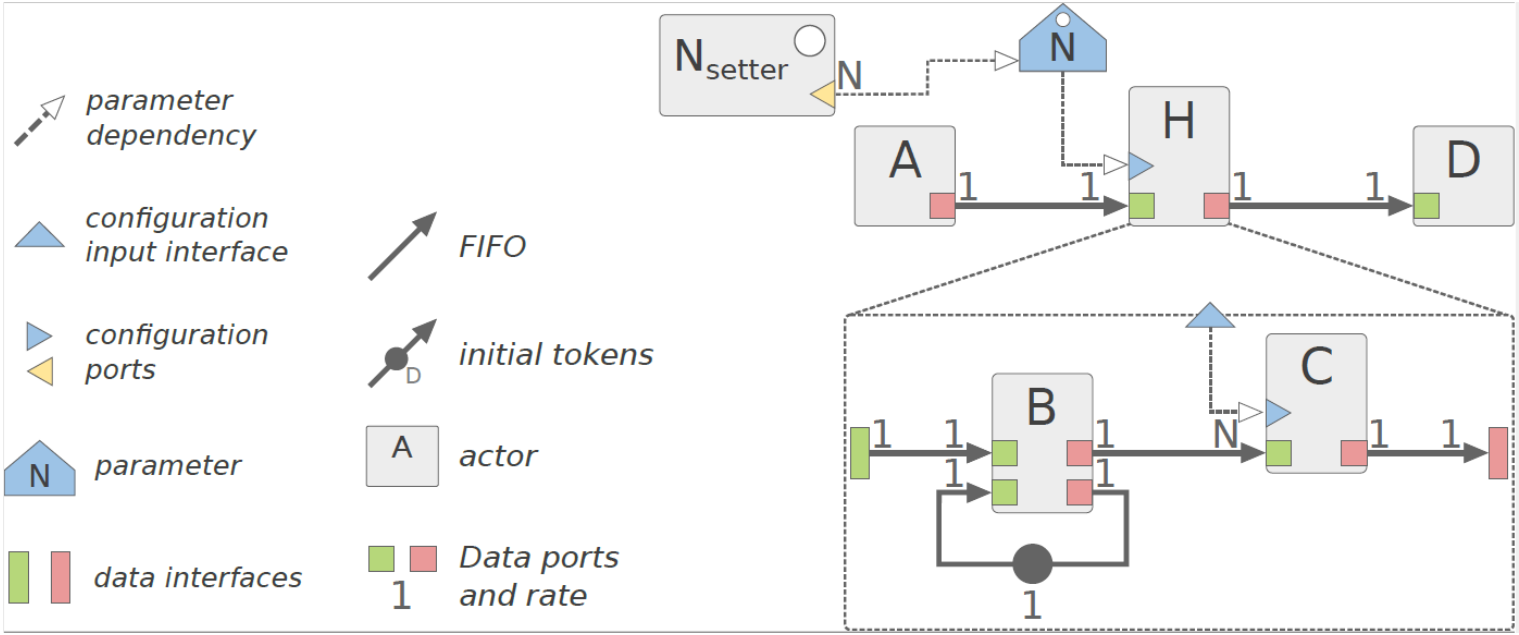


AUTOMATIC CODE GENERATION for RAPID PROTOTYPING for MPSoCs (CPUs + FPGA)

CERBERO

PiSDF: Parameterized and Interfaced SDF

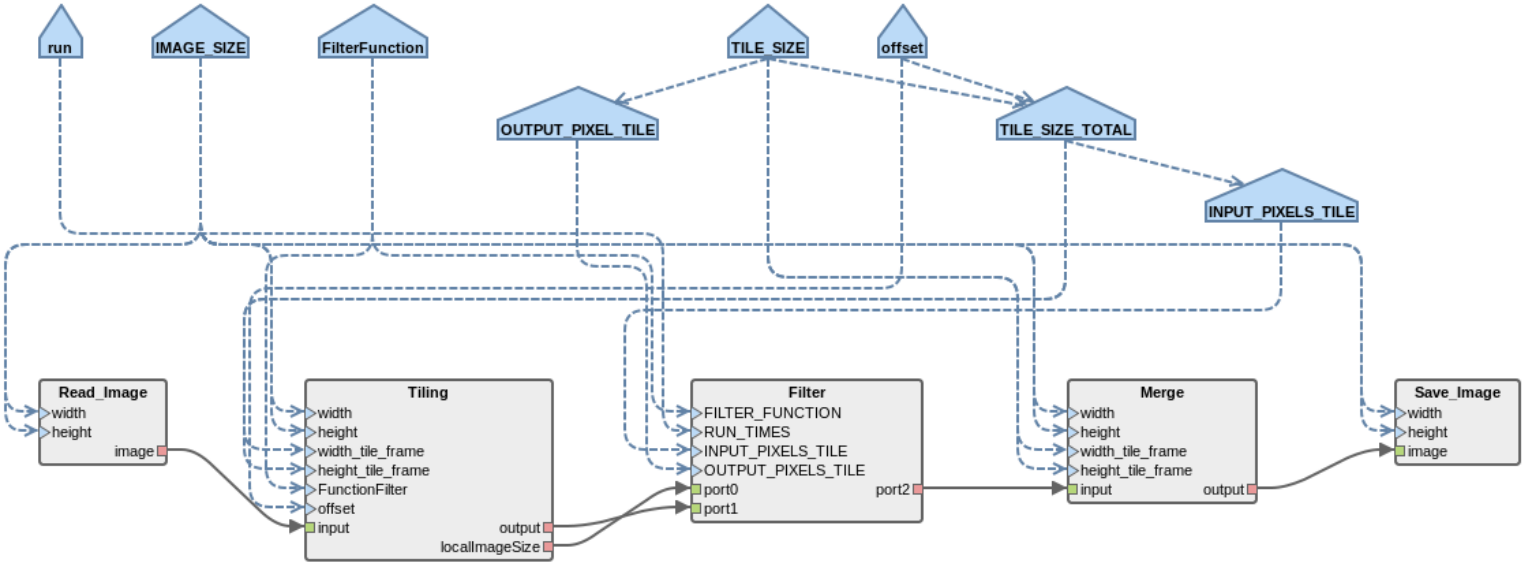
- **Semantics:**



CERBERO

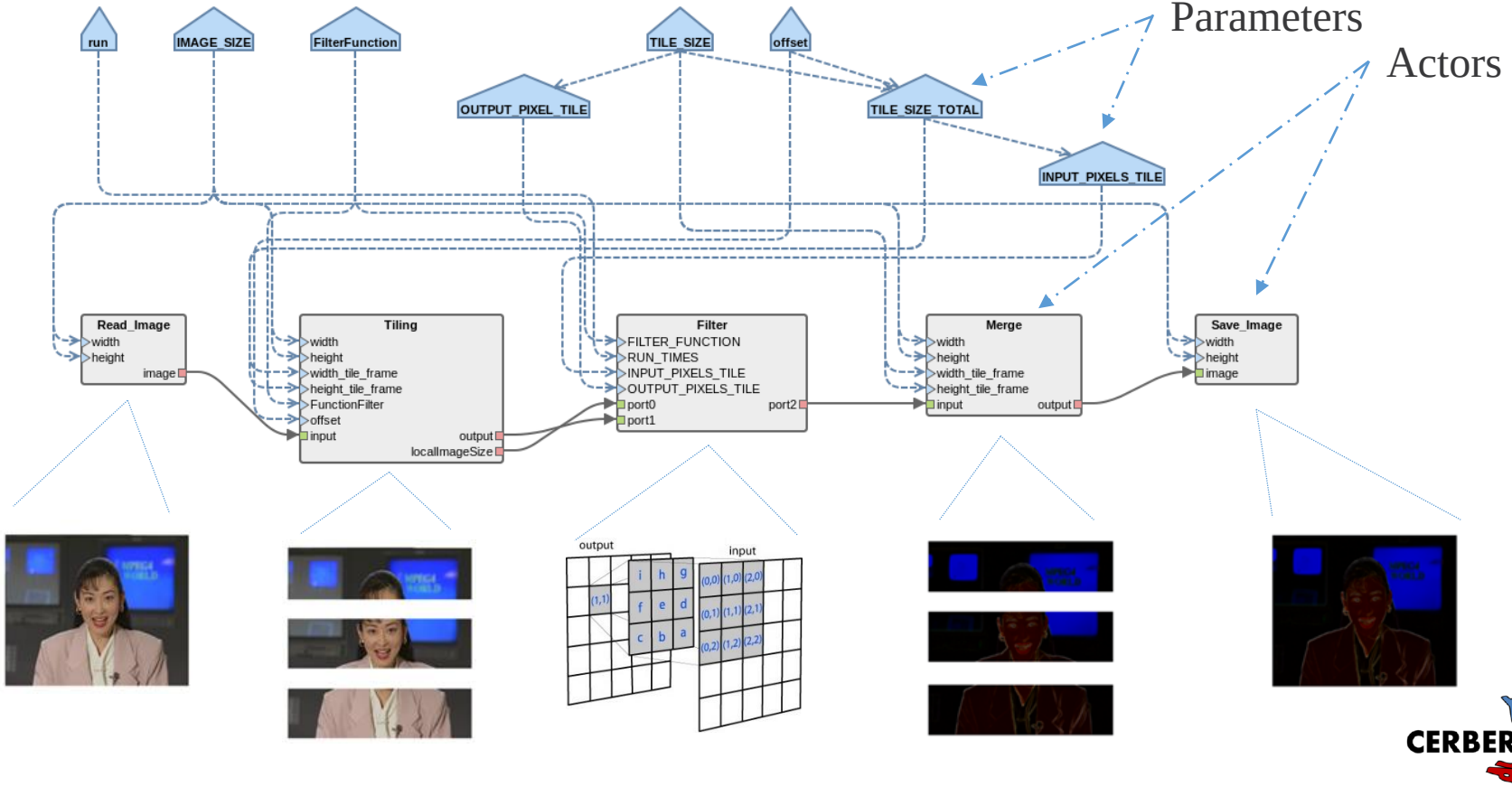
PiSDF: Parameterized and Interfaced SDF

- **Application :**



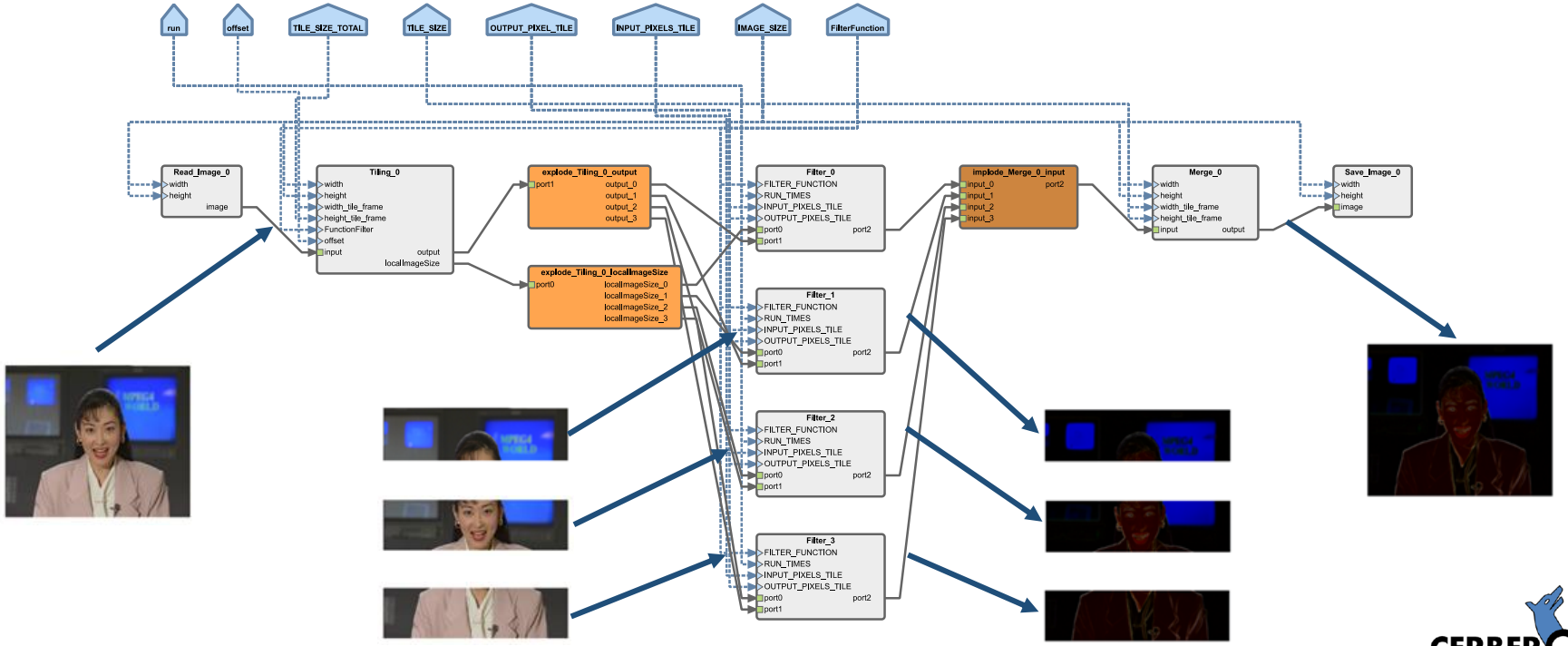
CERBERO

Actor Specification

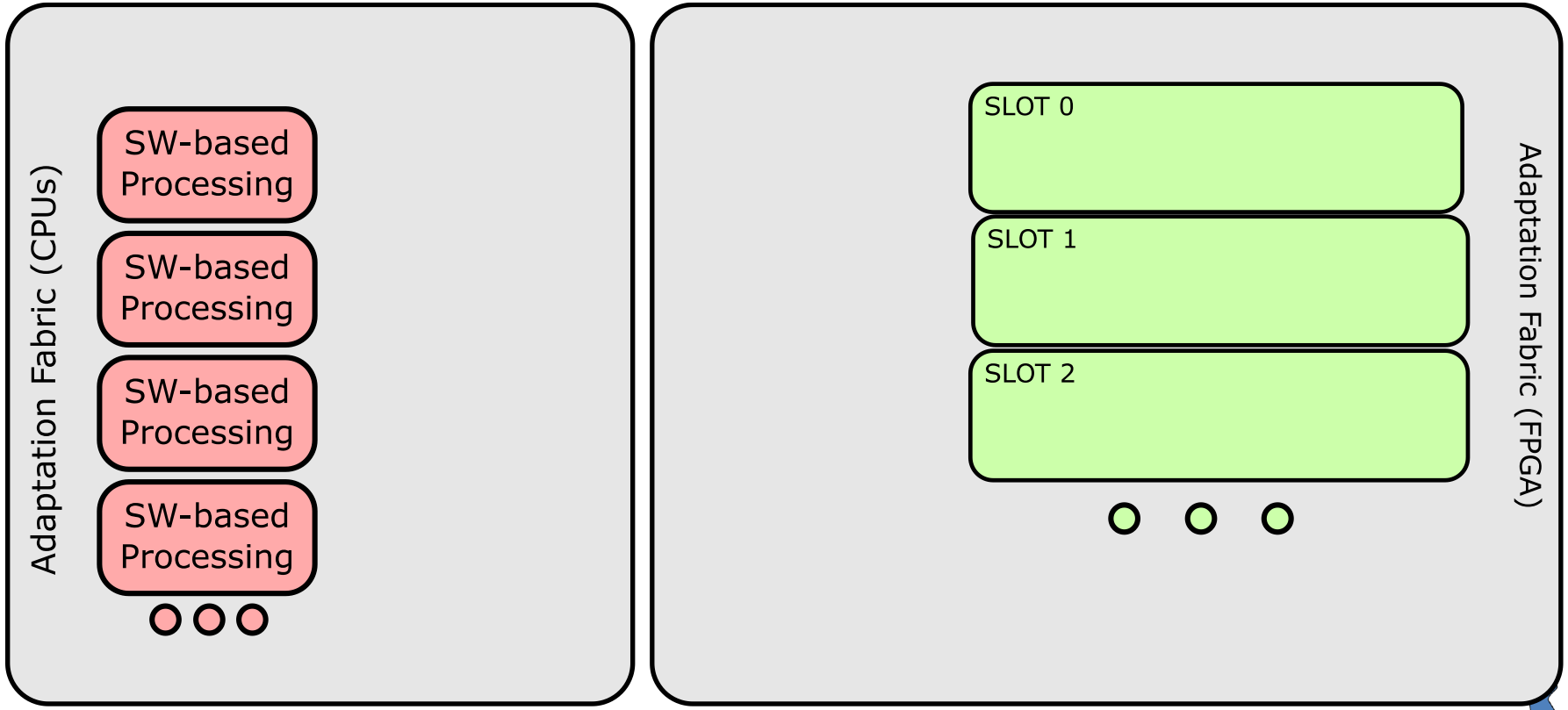


Exposed Parallelism

Equivalent single-rate graph where each edge has equal production and consumption rates of token.

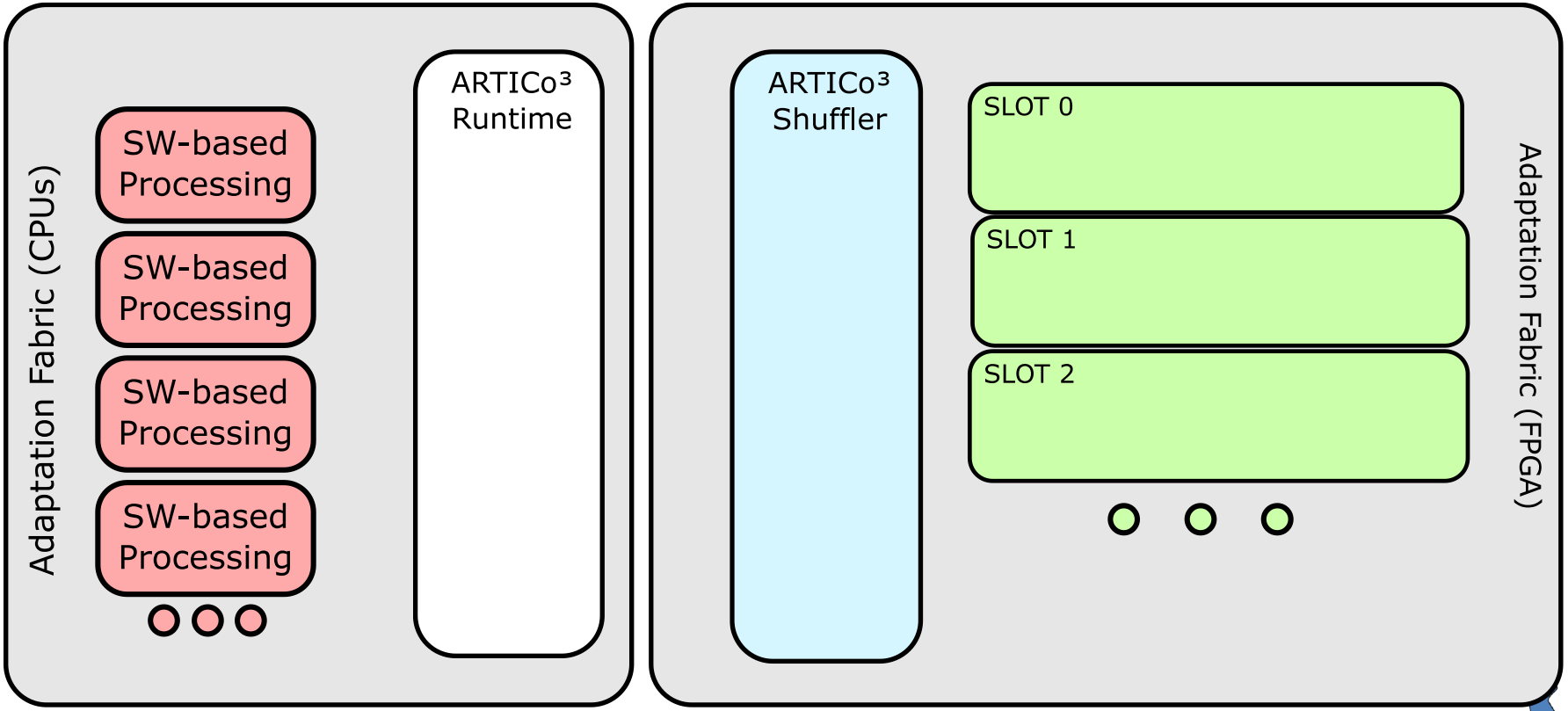


Heterogeneous Device

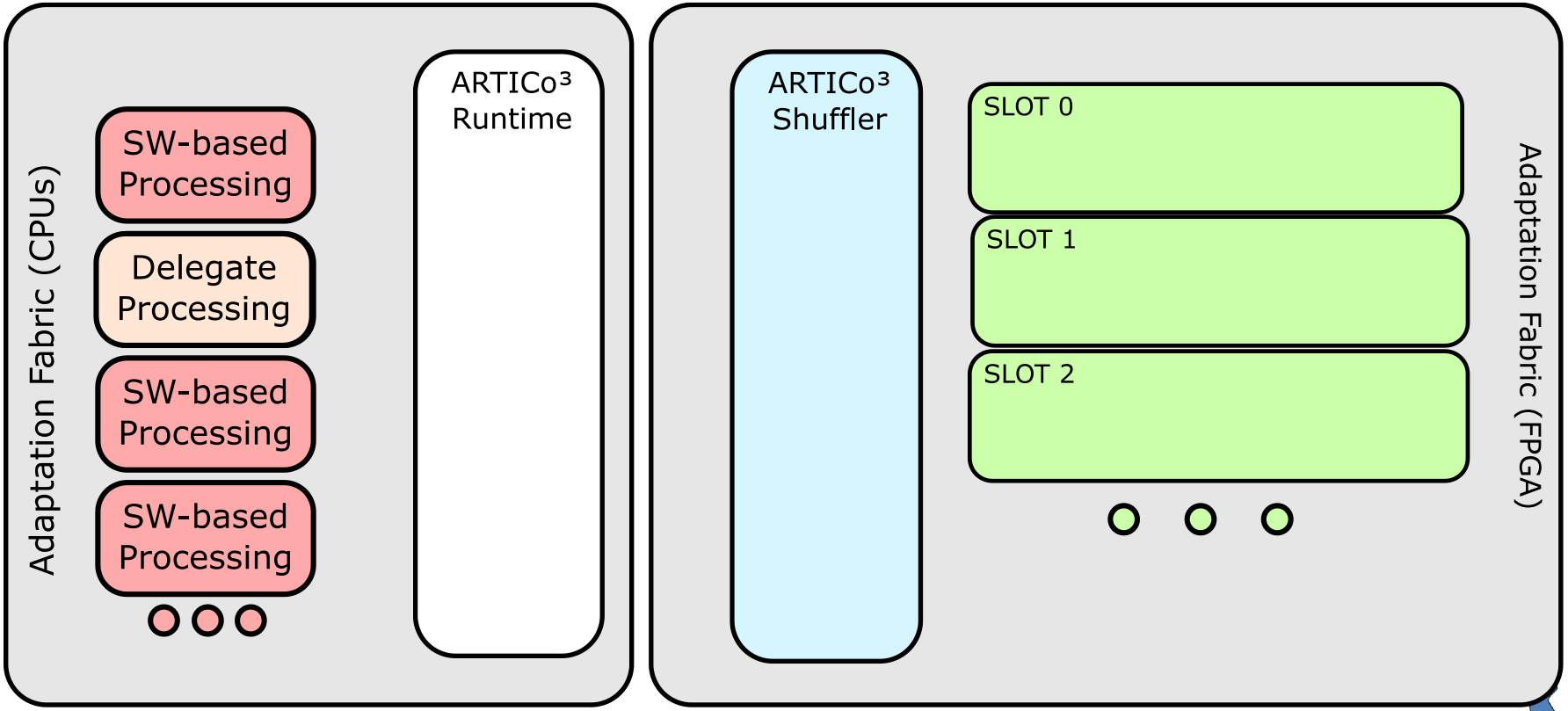


CERBERO

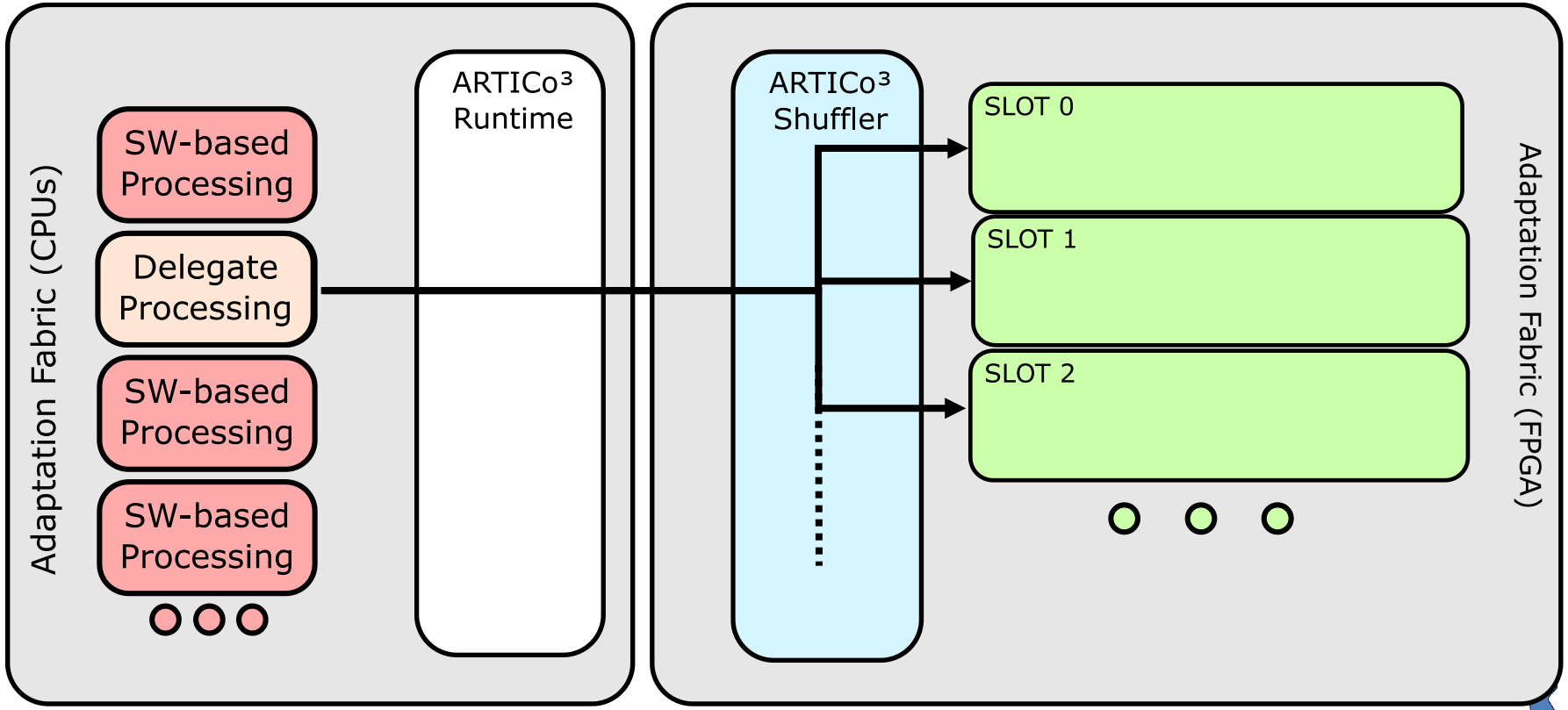
Heterogeneous Device



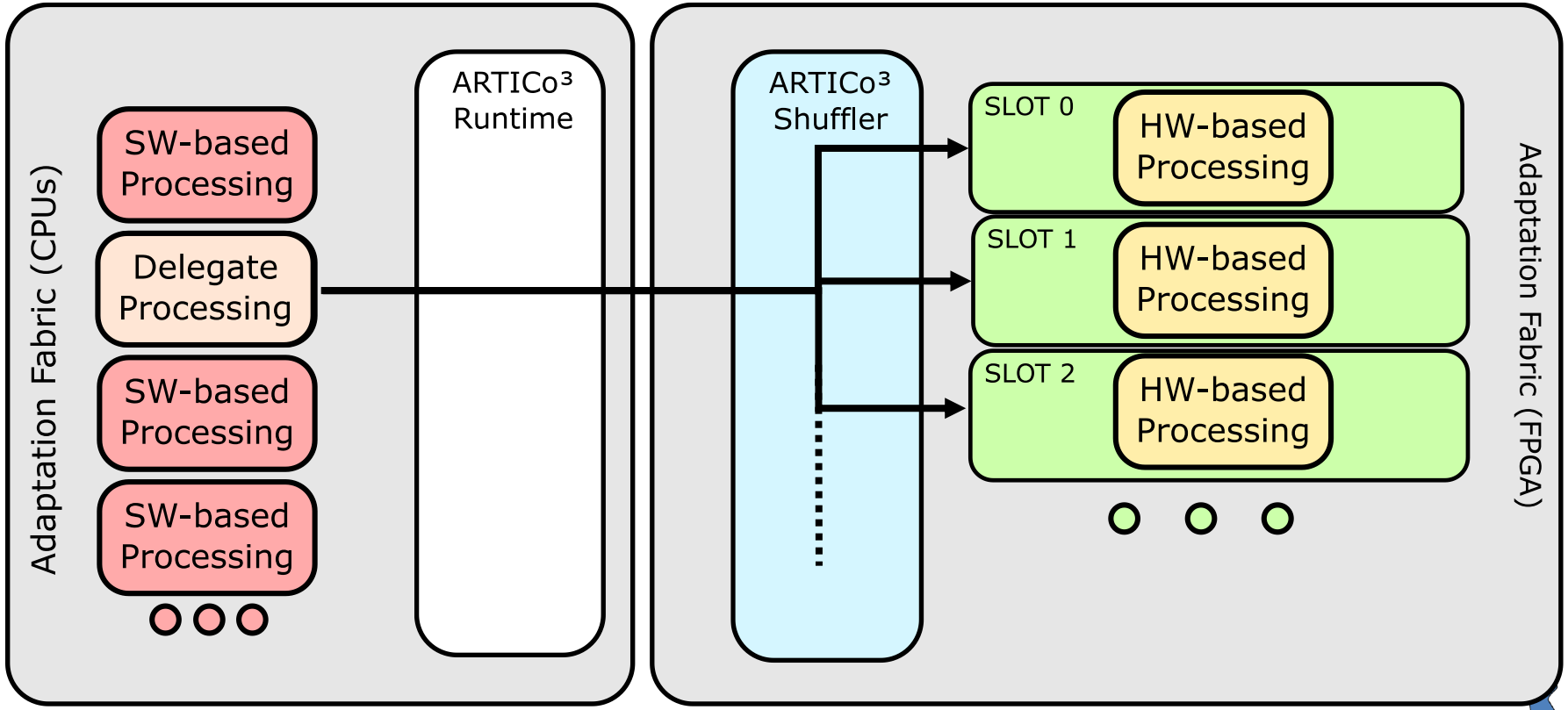
Heterogeneous Device



Heterogeneous Device

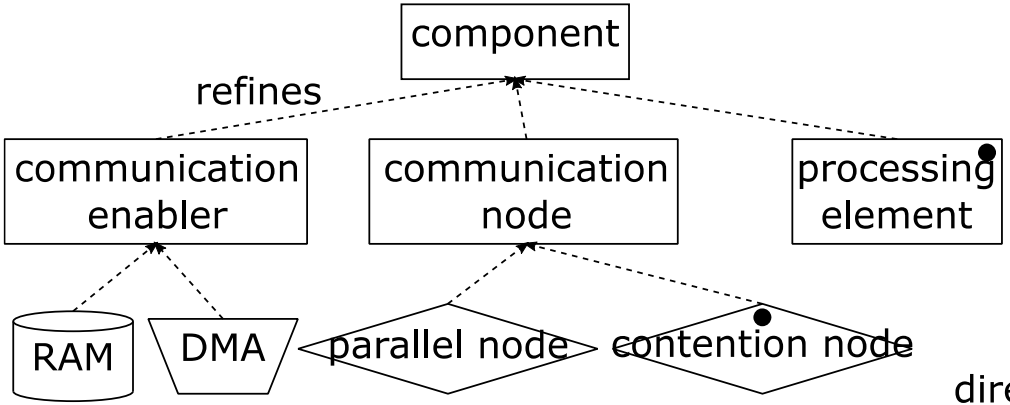


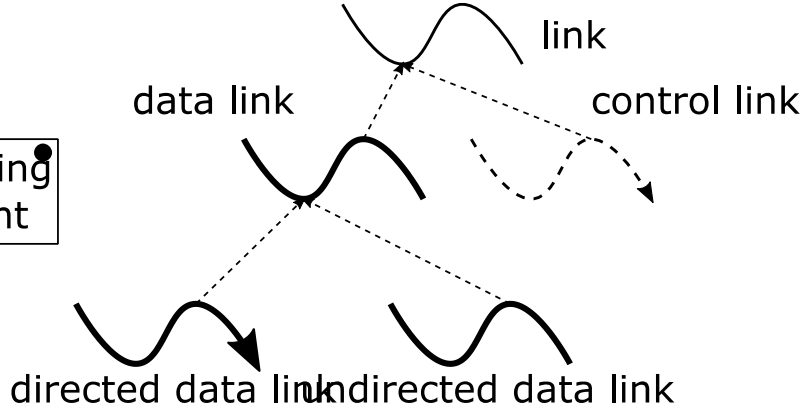
Heterogeneous Device



S-LAM: System-Level Architecture Model

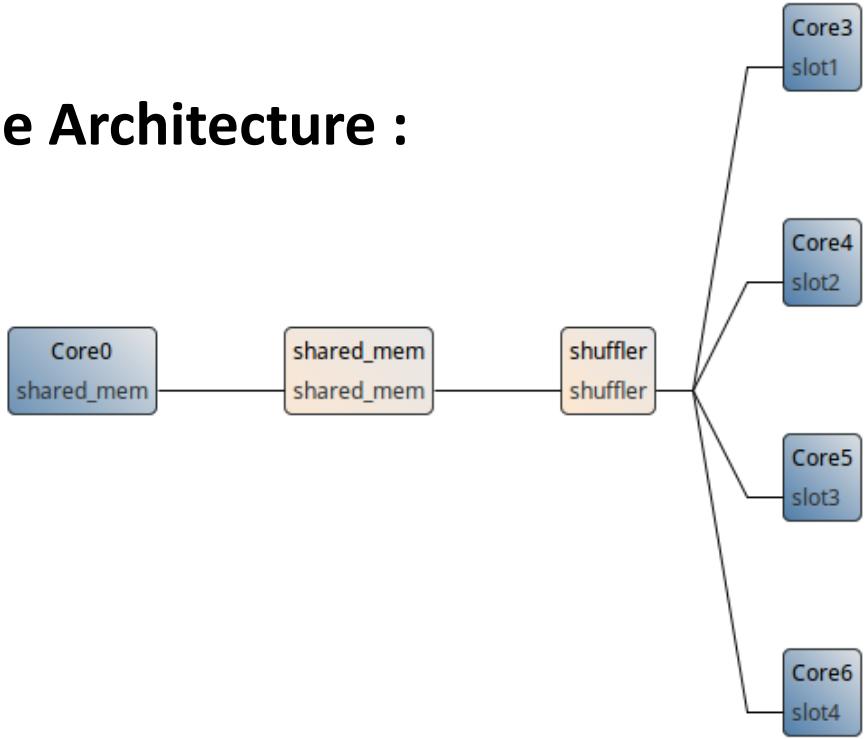
- **S-LAM typology:**



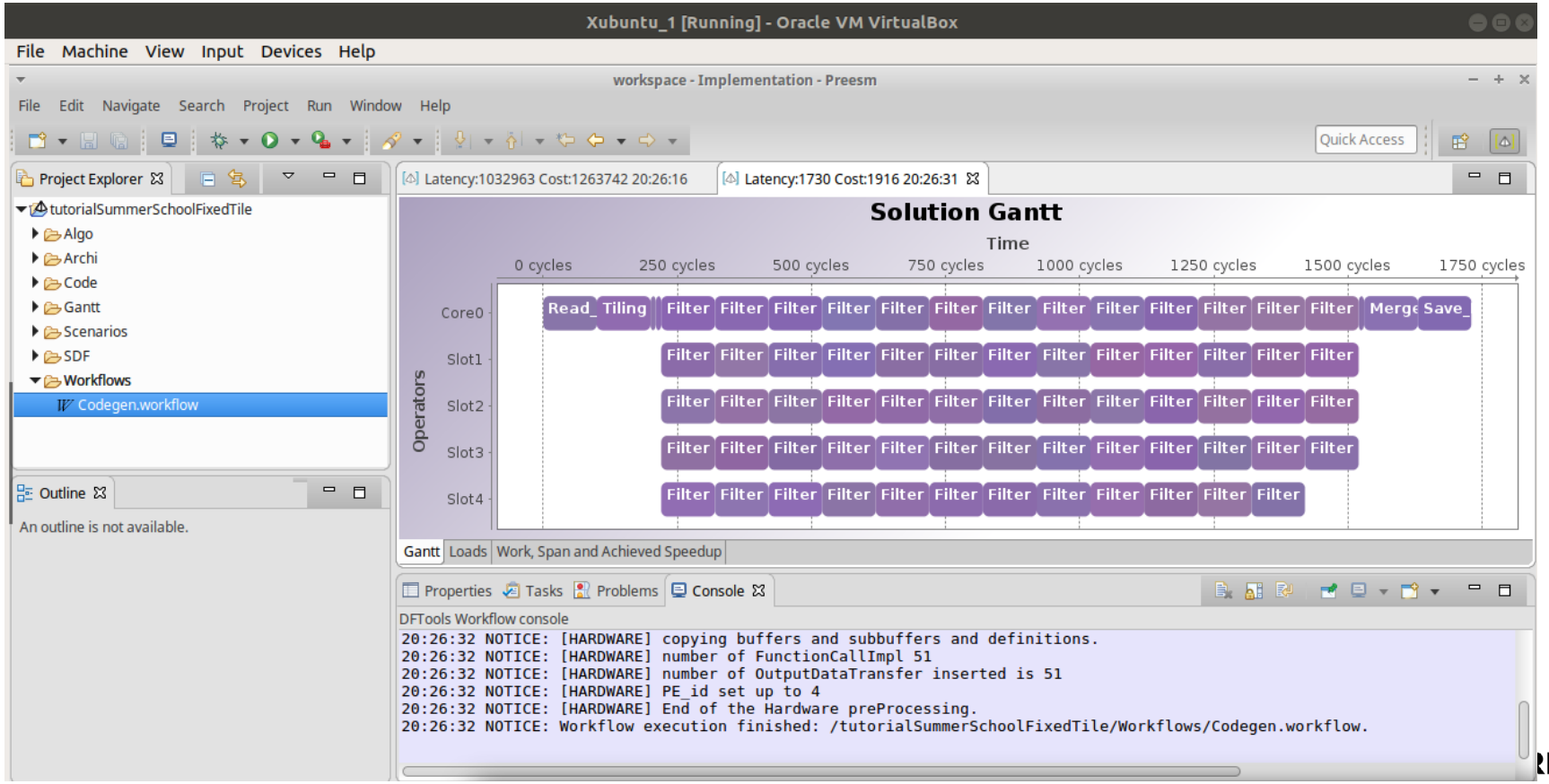


S-LAM: System-Level Architecture Model

- **S-LAM Example Architecture :**



PREESM Mapping and Scheduling

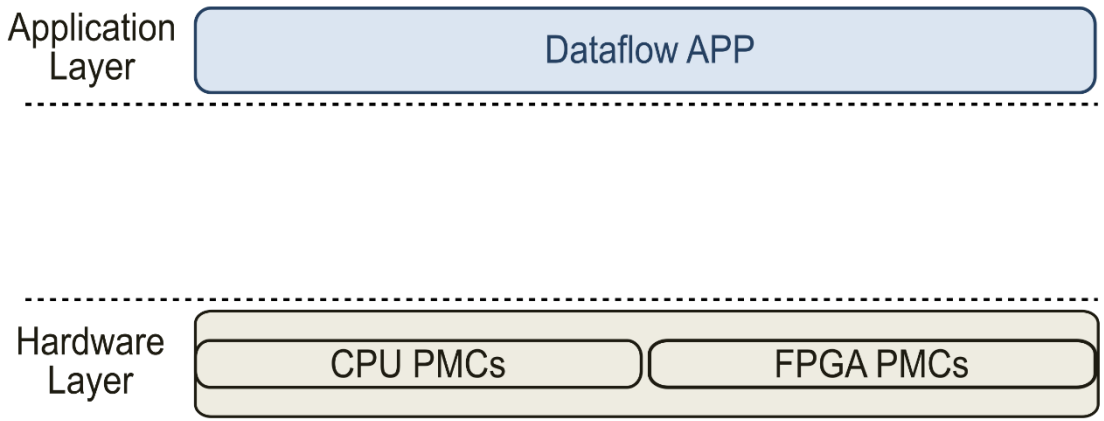


CERBERO

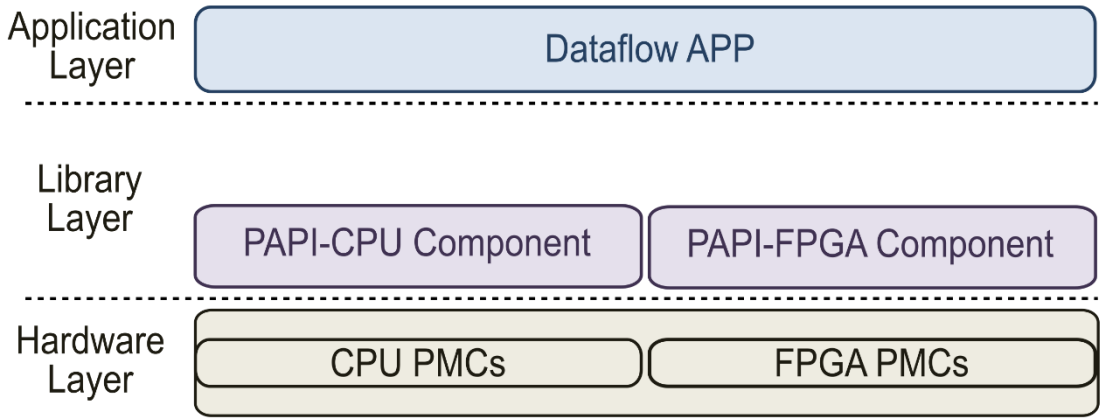
System Performance Monitoring

Horizon 2020
European Union funding
for Research & Innovation

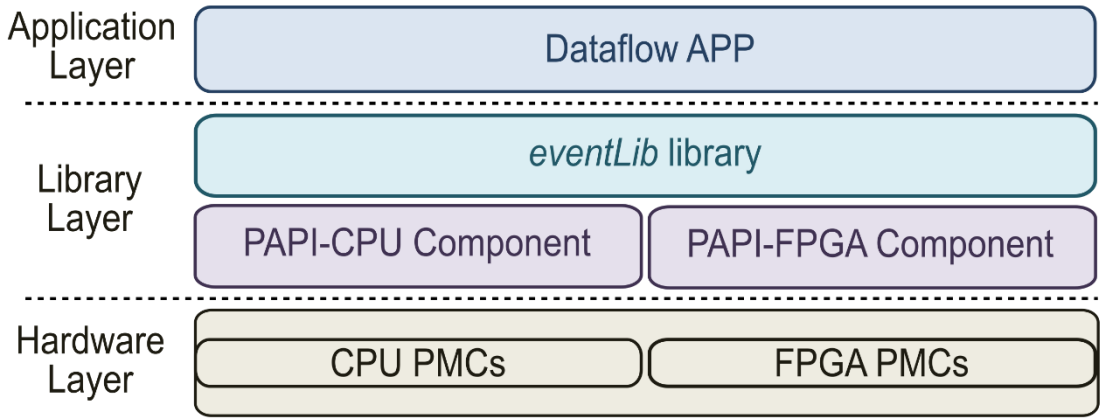
CERBERO Monitoring: PAPIFY



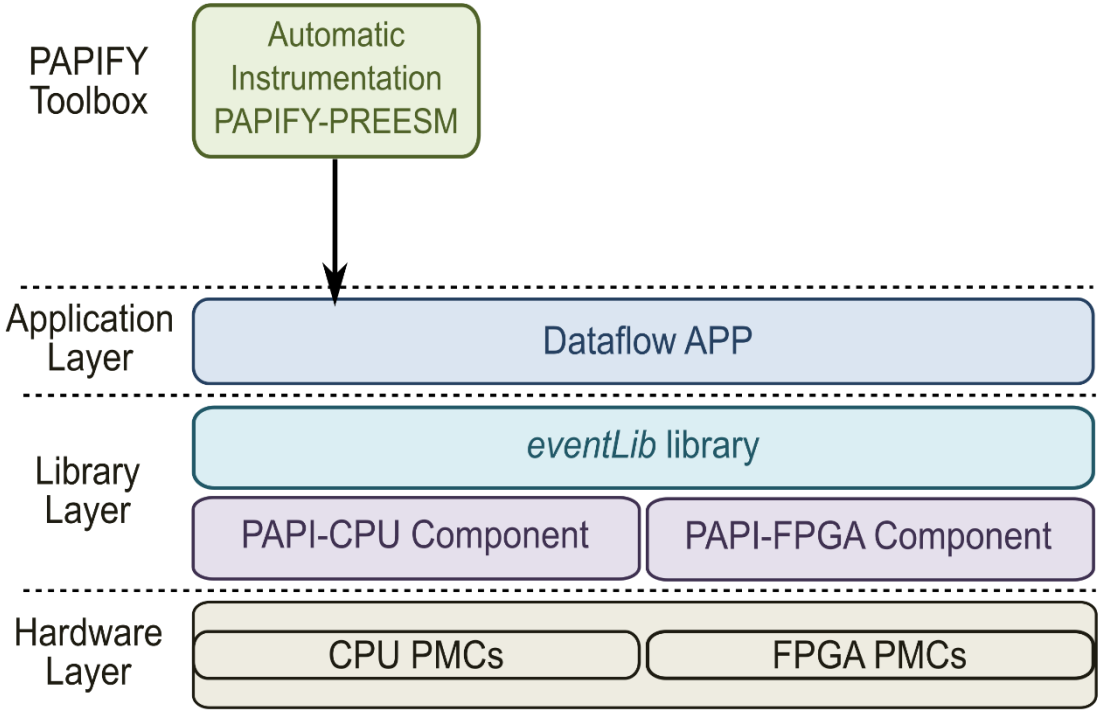
CERBERO Monitoring: PAPIFY



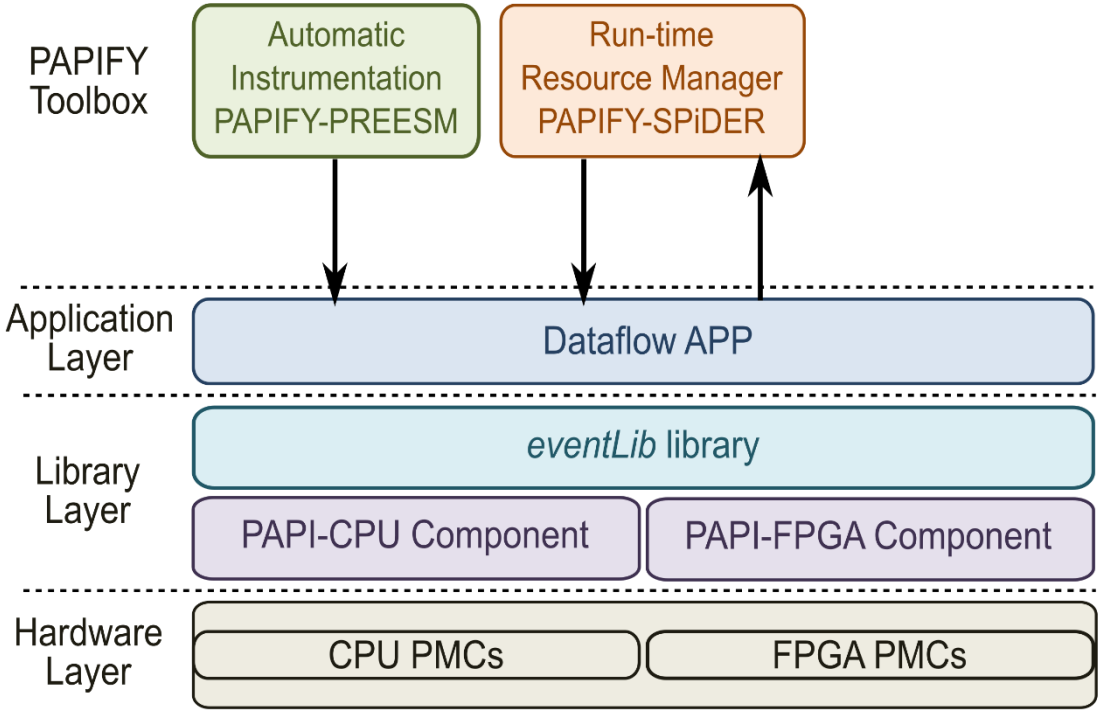
CERBERO Monitoring: PAPIFY



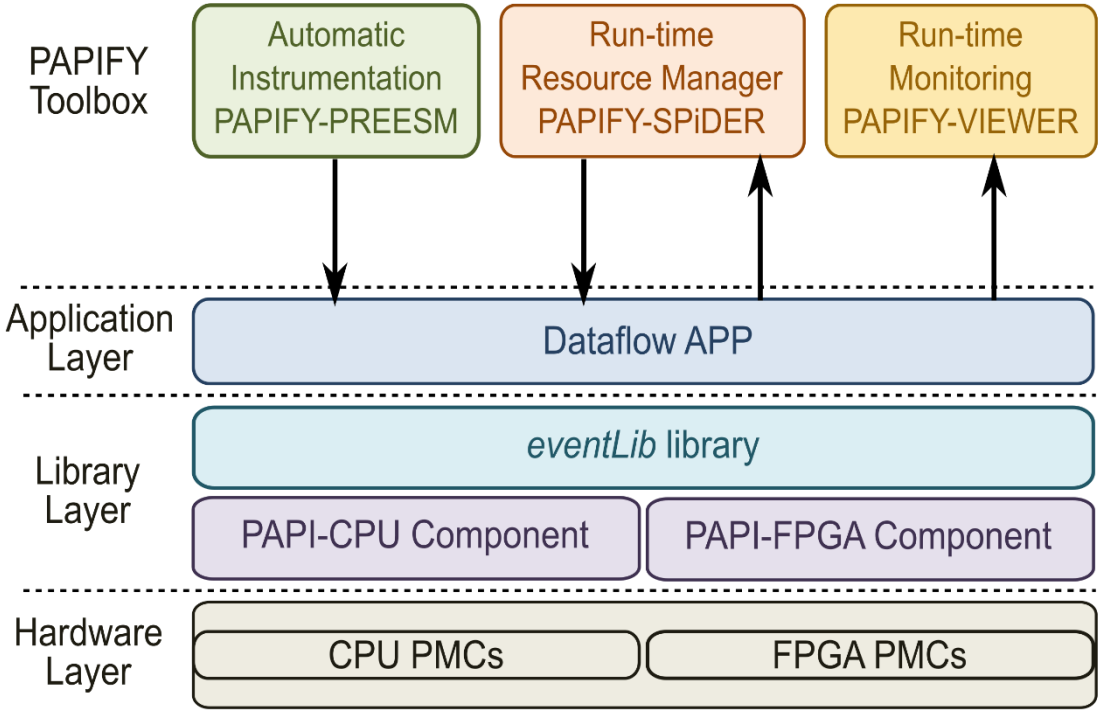
CERBERO Monitoring: PAPIFY



CERBERO Monitoring: PAPIFY



CERBERO Monitoring: PAPIFY



CERBERO Monitoring: PAPIFY

```
configure_papify_actor(...);      // Configure monitoring of each actor
configure_papify_PE(...);         // Configure each PE
```

CERBERO Monitoring: PAPIFY

```
configure_papify_actor(...);      // Configure monitoring of each actor
configure_papify_PE(...);         // Configure each PE
...
...



actor_execution();                // Execute the actor
```

CERBERO Monitoring: PAPIFY

```
configure_papify_actor(...);      // Configure monitoring of each actor
configure_papify_PE(...);         // Configure each PE
...
...
event_start(...);                 // Start monitoring PAPI events
event_start_papify_timing(...);   // Start monitoring timing

actor_execution();                // Execute the actor

event_events(...);                // Stop monitoring PAPI events
event_stop_papify_timing(...);    // Stop monitoring timing
event_write_file(…);              // Dump monitoring info
```

CERBERO

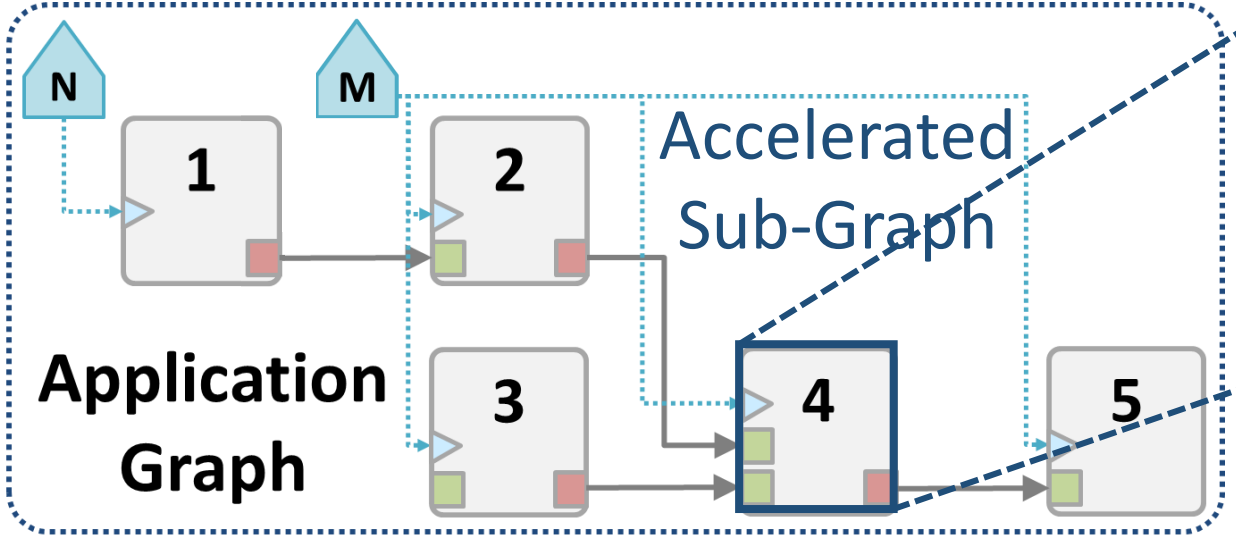
HiPEAC 2020
20-22 January, 2020, Bologna, Italy

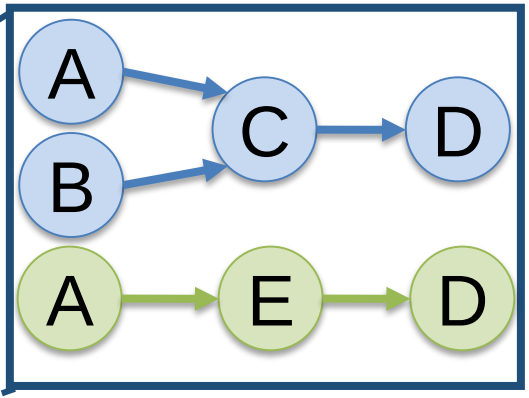
CERBERO

Creating Hardware Accelerators with DPR and CGR

Horizon 2020
European Union funding
for Research & Innovation

MDC Suite





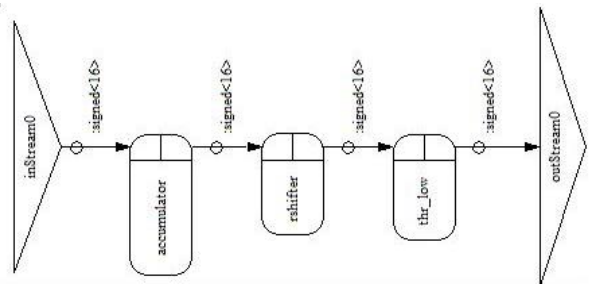
specification of the algorithm at a **fine granularity**

specification of the algorithm at a **coarse granularity**

CERBERO

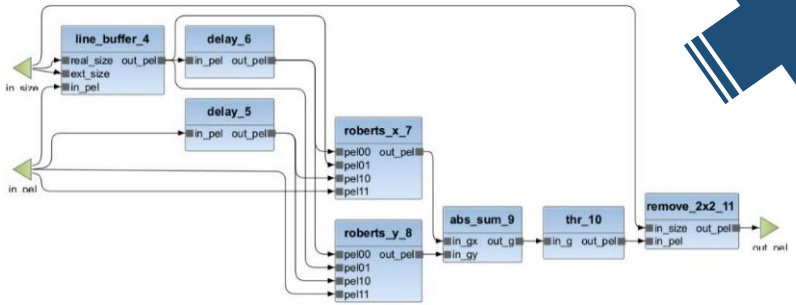
MDC Suite

CAPH



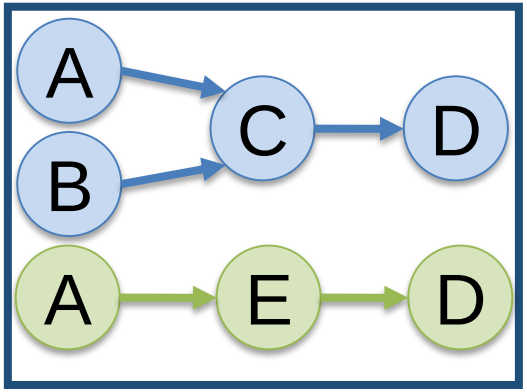
fully integrated flow, from **dataflow** to **hardware**

ORCC + Vivado HLS



two steps flow: **ORCC** for dataflow **network**, **Vivado HLS** deals for **actors** hardware code generation

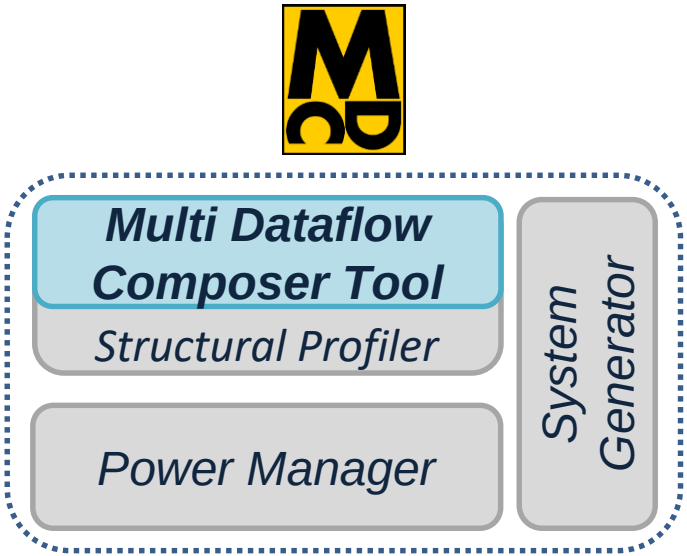
Multi-Functional



specification of the algorithm at a **fine granularity**

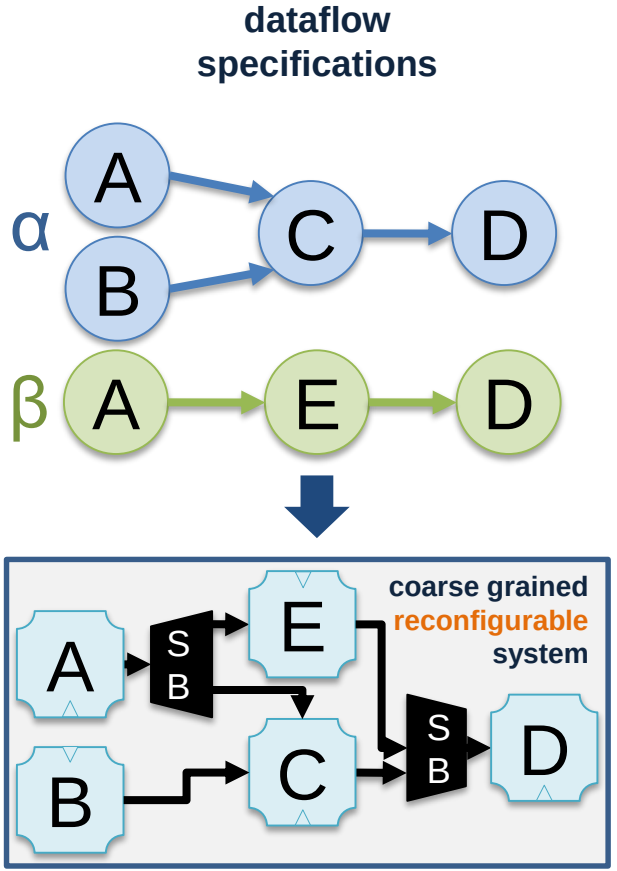
CERBERO

MDC Suite



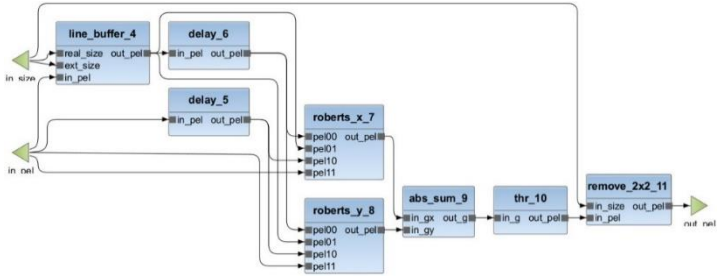
MDC design suite

http://sites.unica.it/rpct/

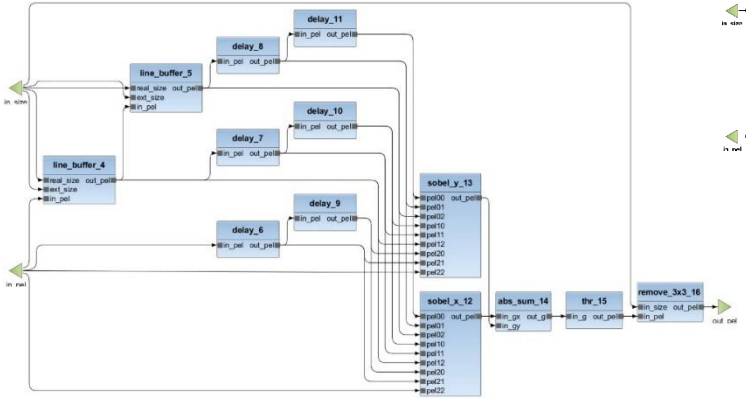


MDC: Merging

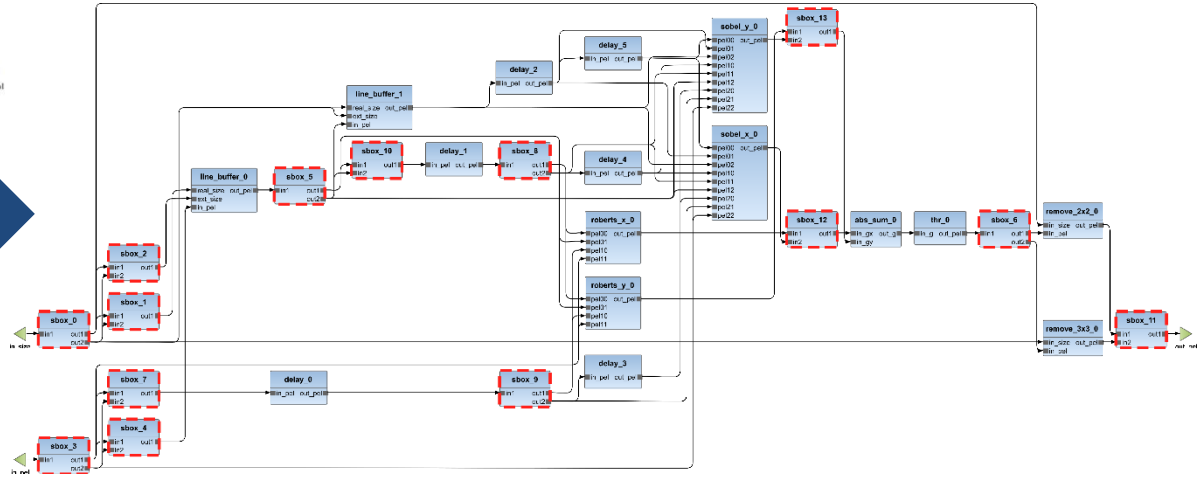
Roberts network



Sobel network

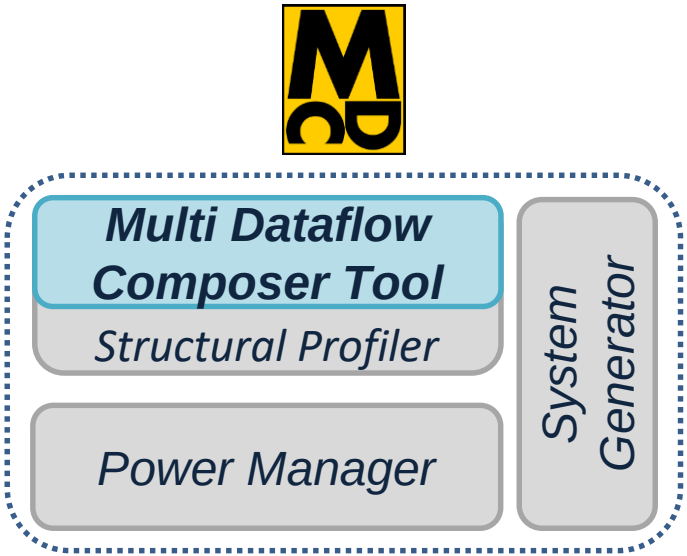


Multi-dataflow network



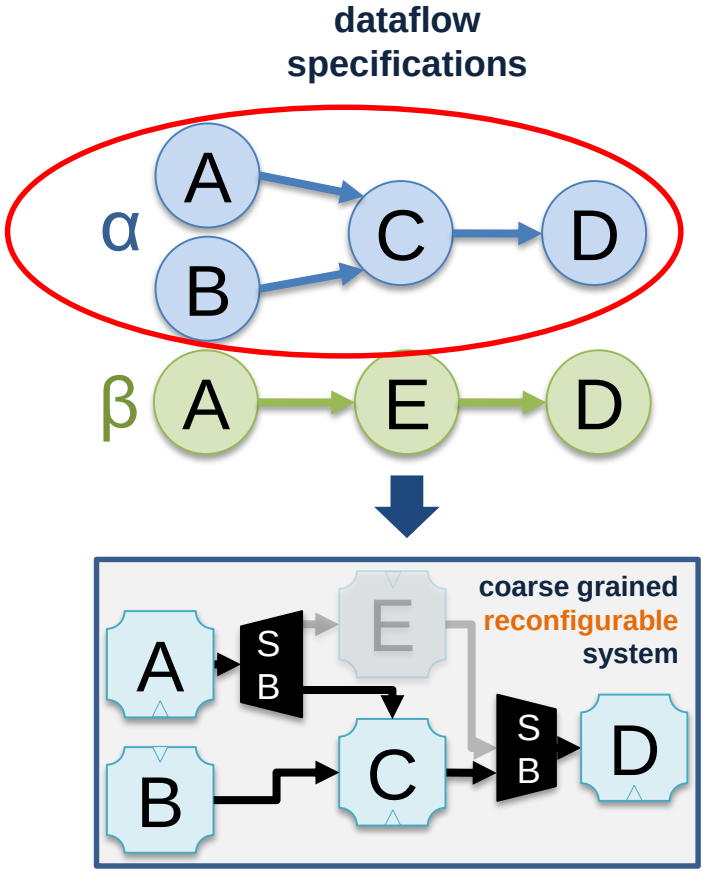
CERBERO

MDC Suite



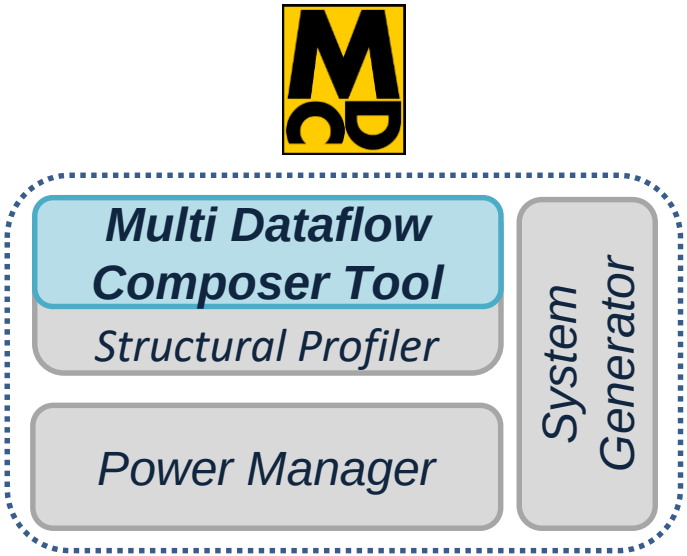
MDC design suite

http://sites.unica.it/rpct/



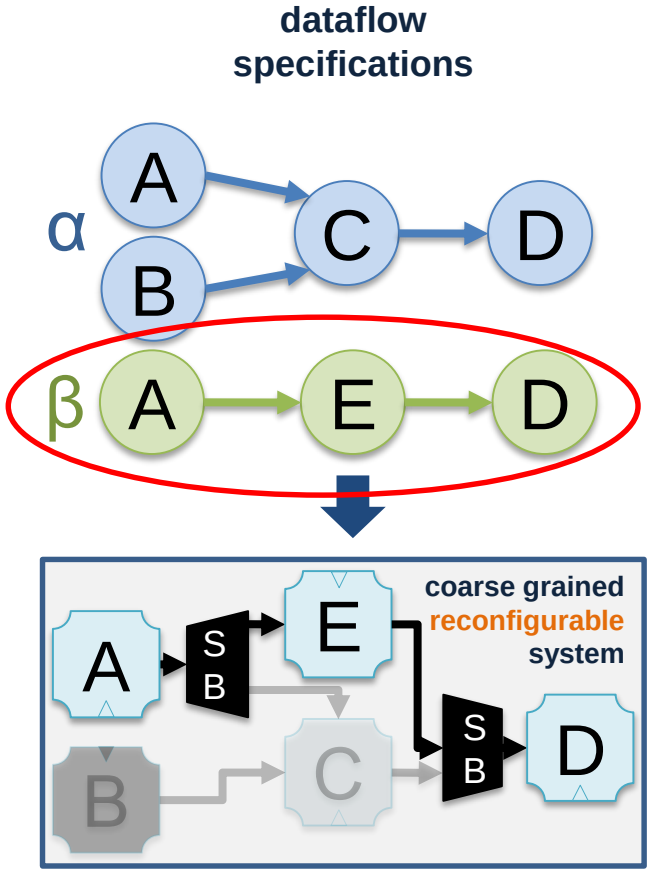
CERBERO

MDC Suite



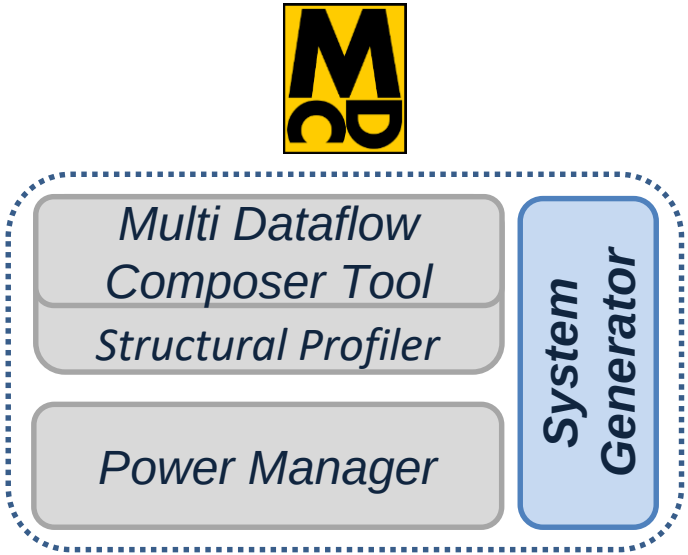
MDC design suite

http://sites.unica.it/rpct/



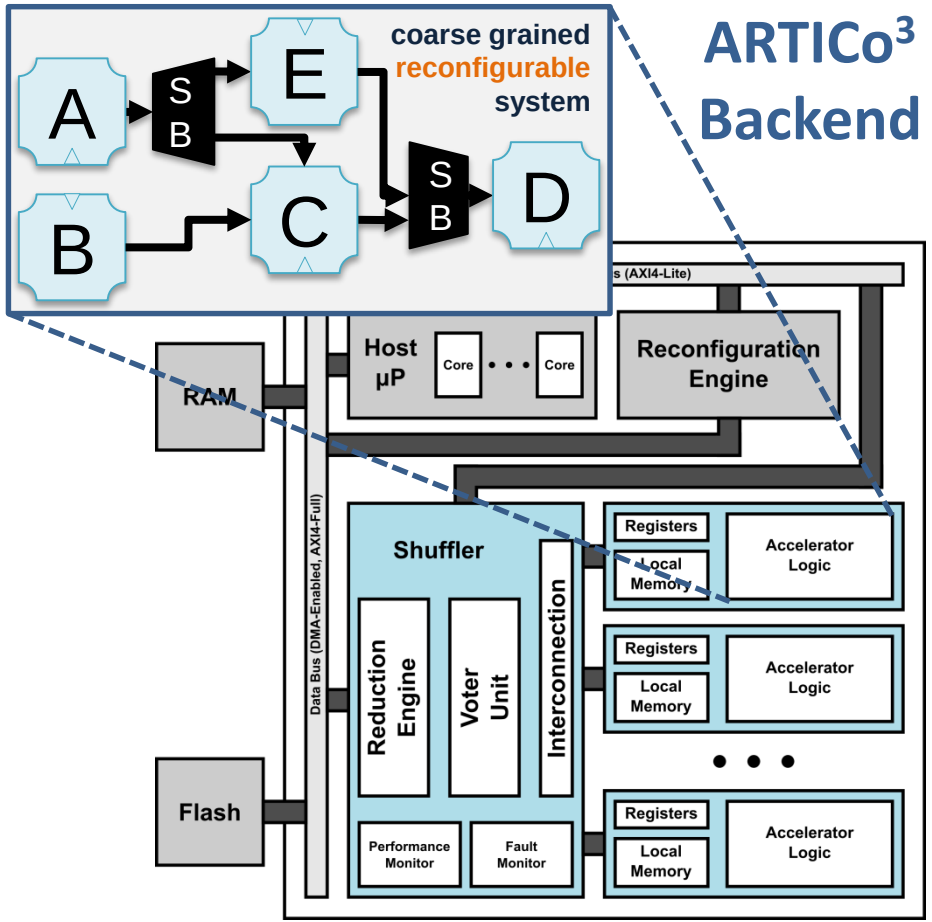
CERBERO

MDC Suite



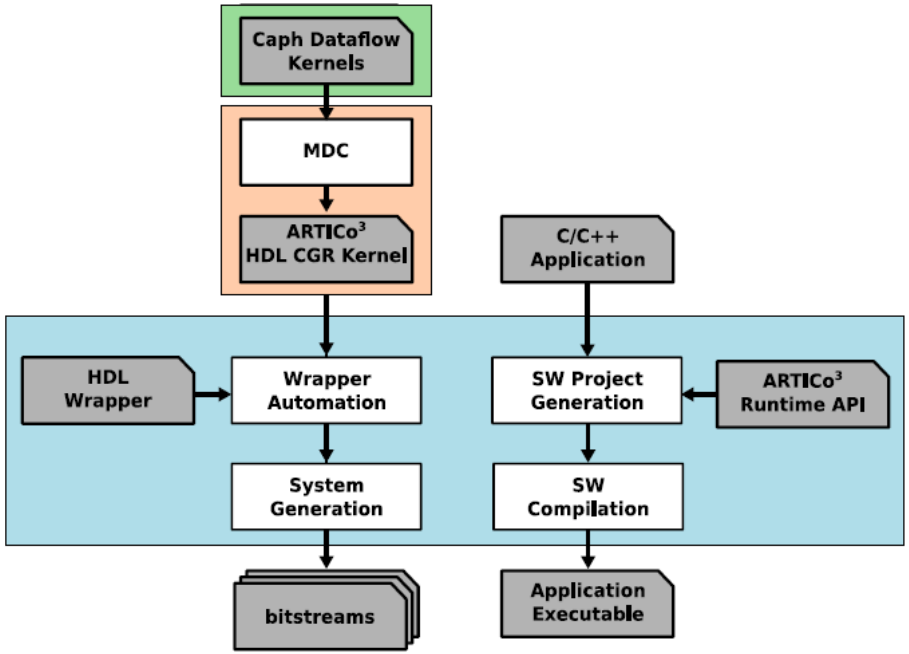
MDC design suite

http://sites.unica.it/rpct/



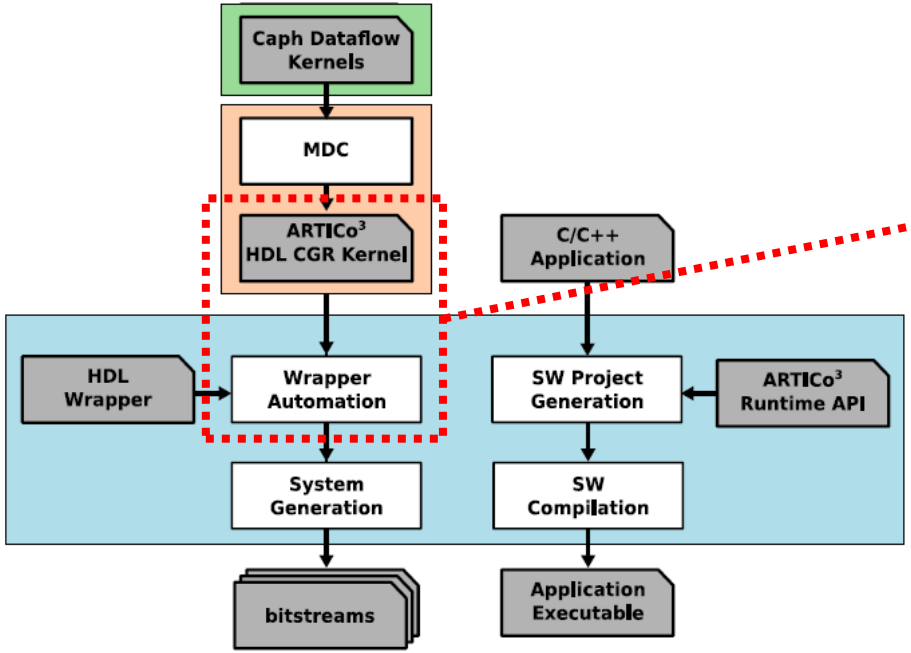
CERBERO

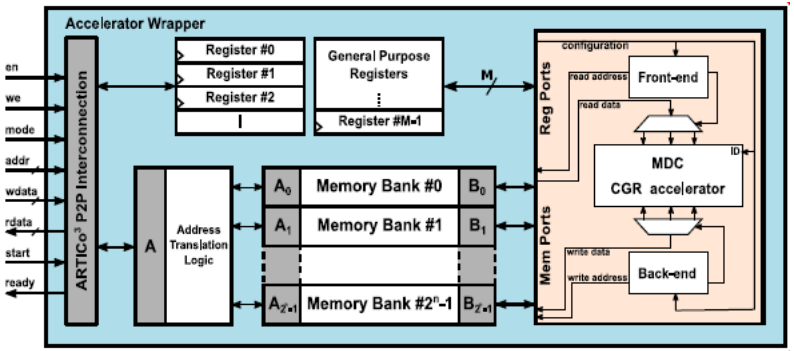
ARTICo³ Design Flow



CERBERO

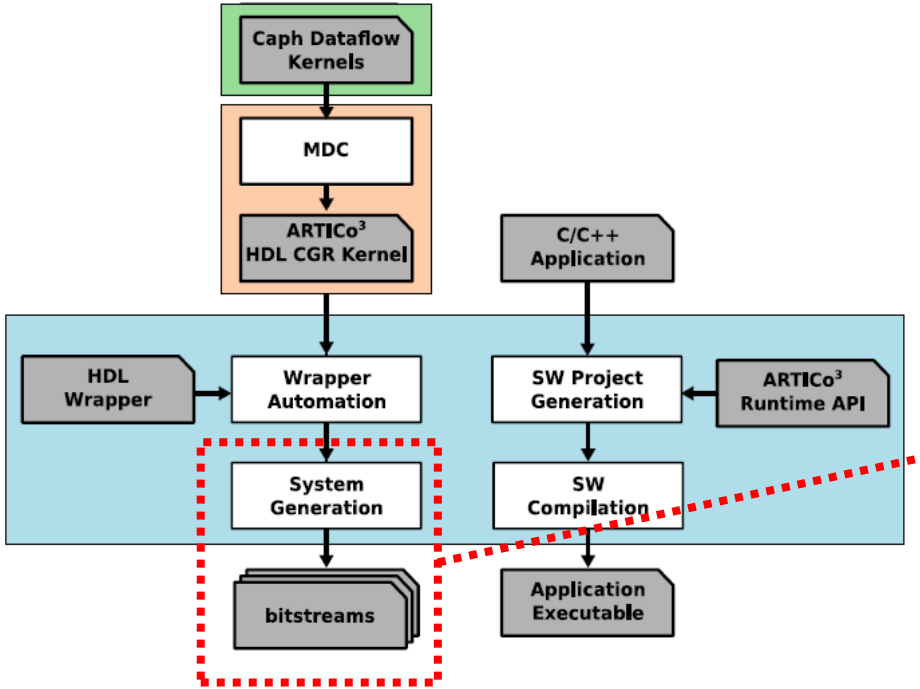
ARTICo³ Design Flow

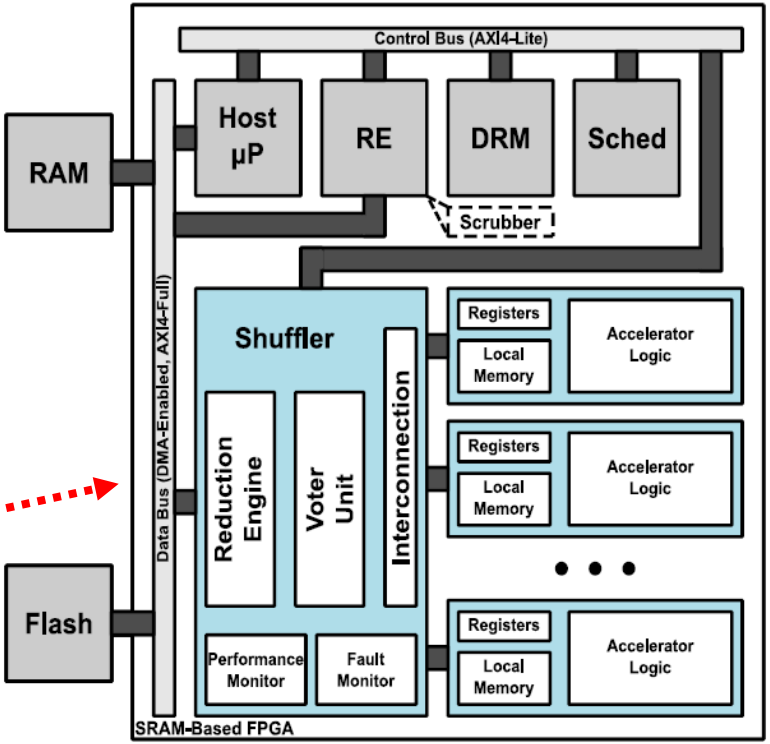




CERBERO

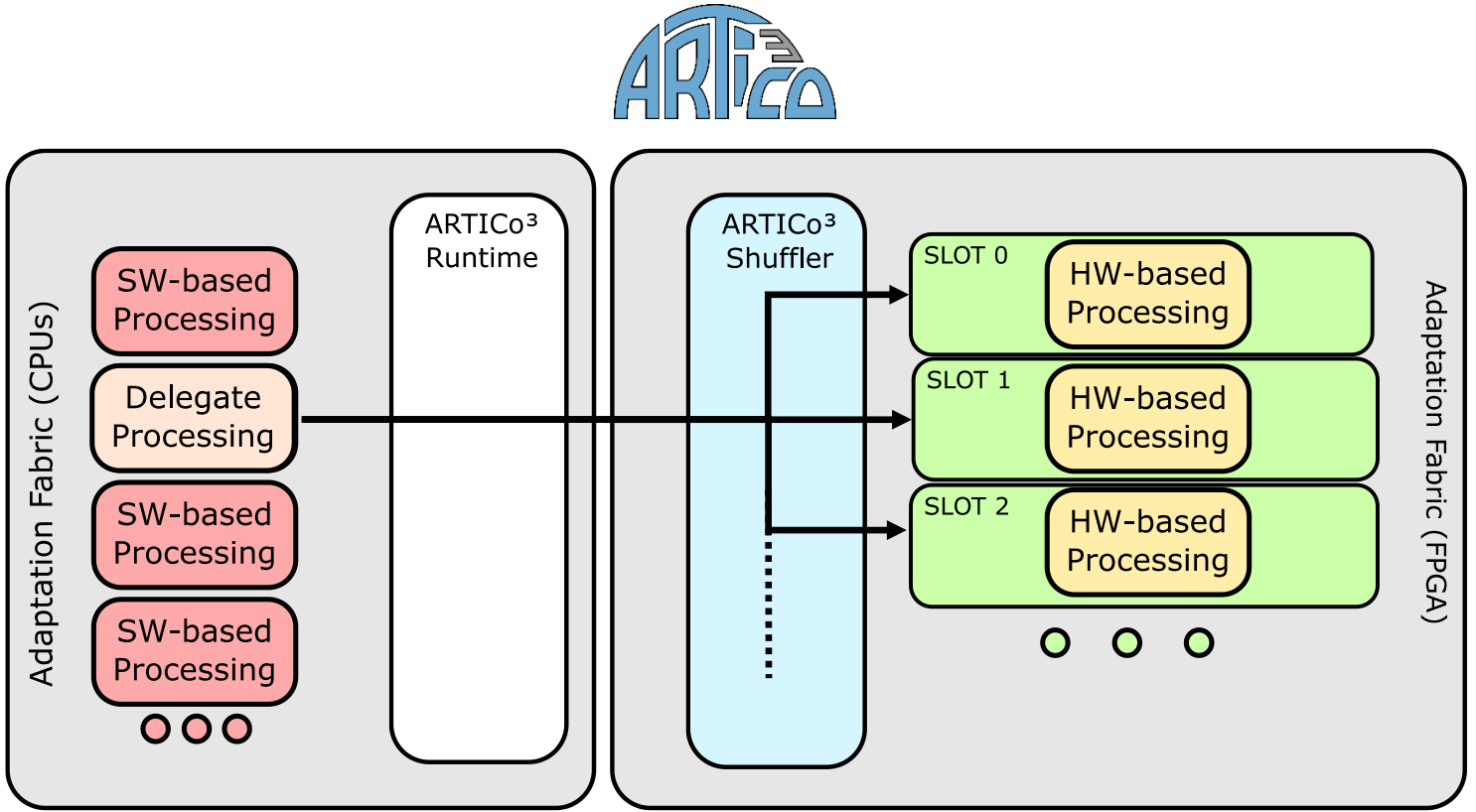
ARTICo³ Design Flow



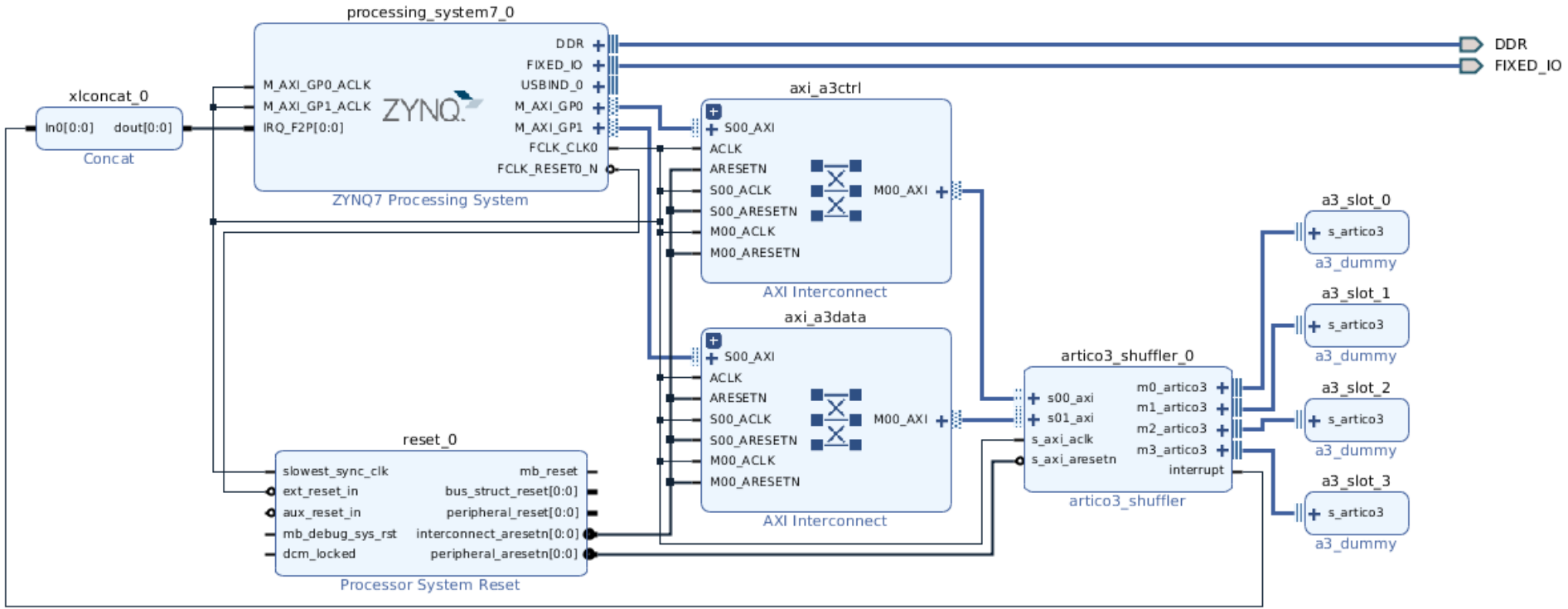


CERBERO

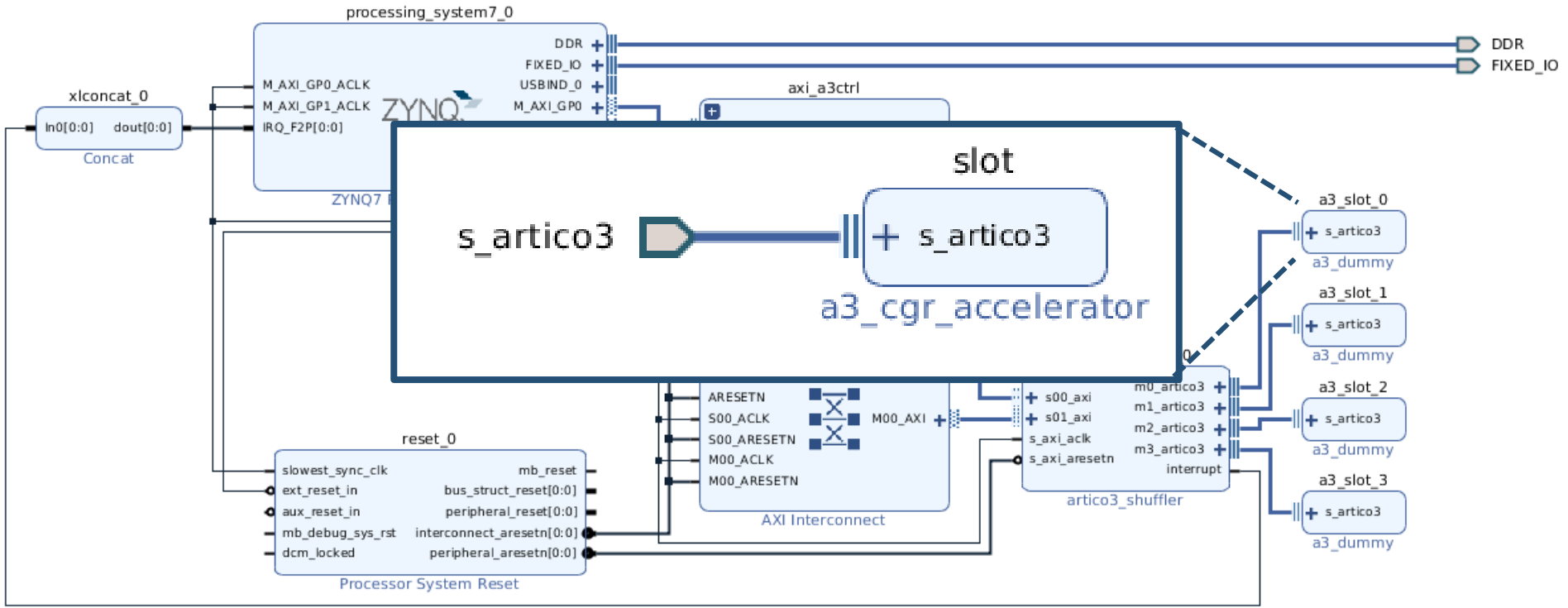
Heterogeneous Device



Architecture Overview in Vivado



Architecture Overview in Vivado



HiPEAC 2020
20-22 January, 2020, Bologna, Italy

CERBERO H2020 Project – Tutorial: Design Flow for Heterogeneous Embedded Computing Infrastructures

Horizon 2020
European Union funding
for Research & Innovation

Additional Slides