EUROPEAN PROCESSOR INITIATIVE: CHALLENGES & OPPORTUNITIES FOR RISC-V ACCELERATORS IN AN HPC PLATFORM

European
Processor
Initiative

FRAMEWORK PARTNERSHIP AGREEMENT IN EUROPEAN LOW-POWER MICROPROCESSOR TECHNOLOGIES

THIS PROJECT HAS RECEIVED FUNDING FROM THE EUROPEAN UNION'S HORIZON 2020 RESEARCH AND INNOVATION PROGRAMME UNDER GRANT AGREEMENT NO 826647

INTRODUCTION TO THE EPI

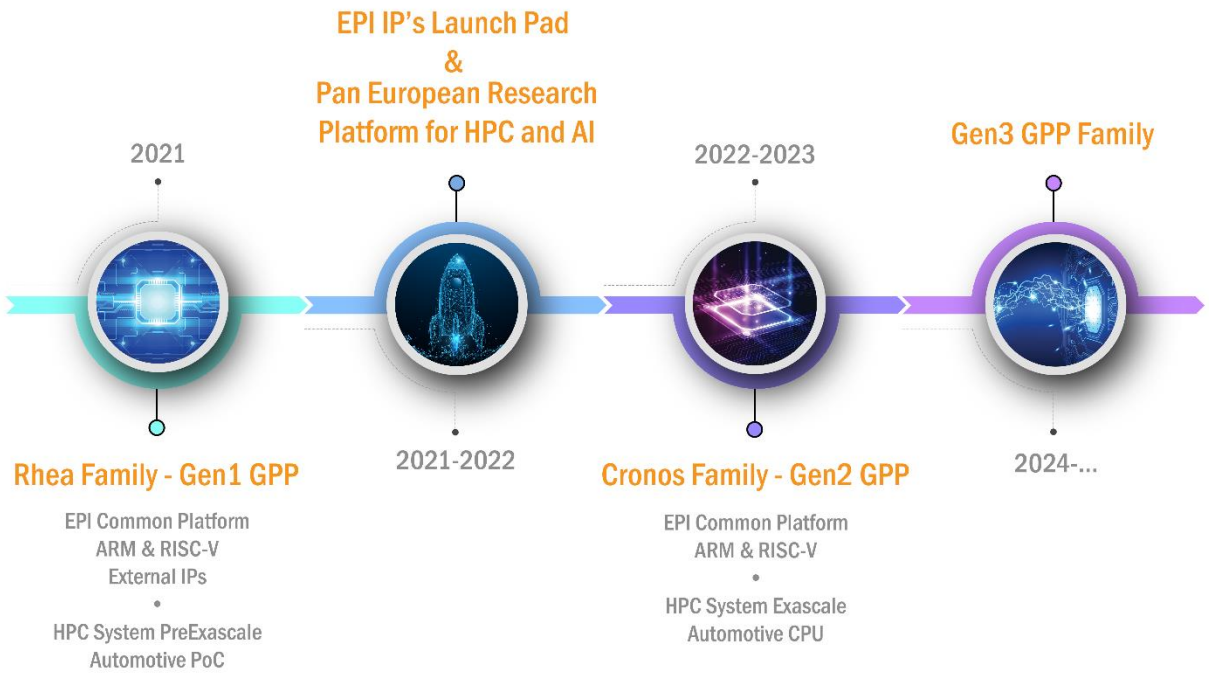
EUROPEAN PROCESSOR INITIATIVE

- High Performance General Purpose Processor for HPC
- High-performance RISC-V based accelerator
- Computing platform for autonomous cars
- Will also target the AI, Big Data and other markets in order to be economically sustainable

EUROPEAN PROCESSOR INITIATIVE

European Processor Initiative

www.european-processor-initiative.eu

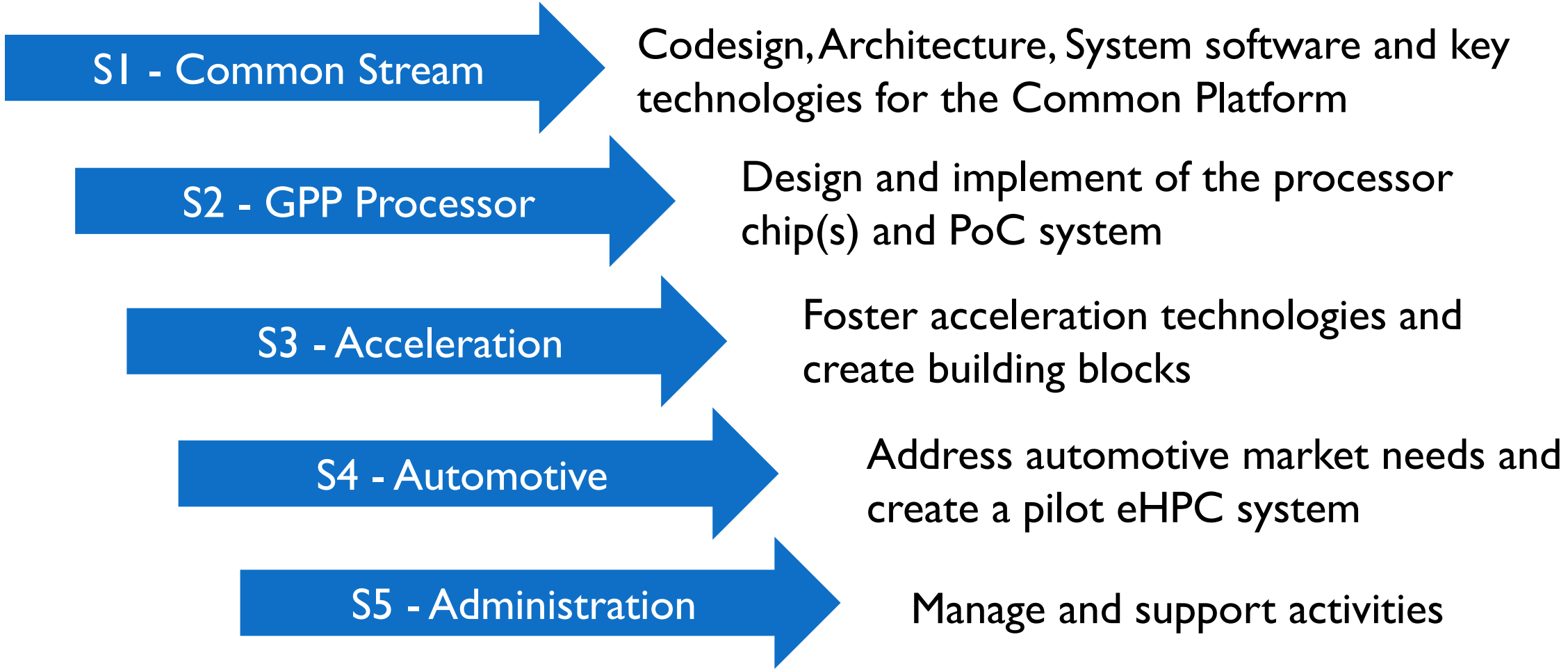


PROJECT PILLARS

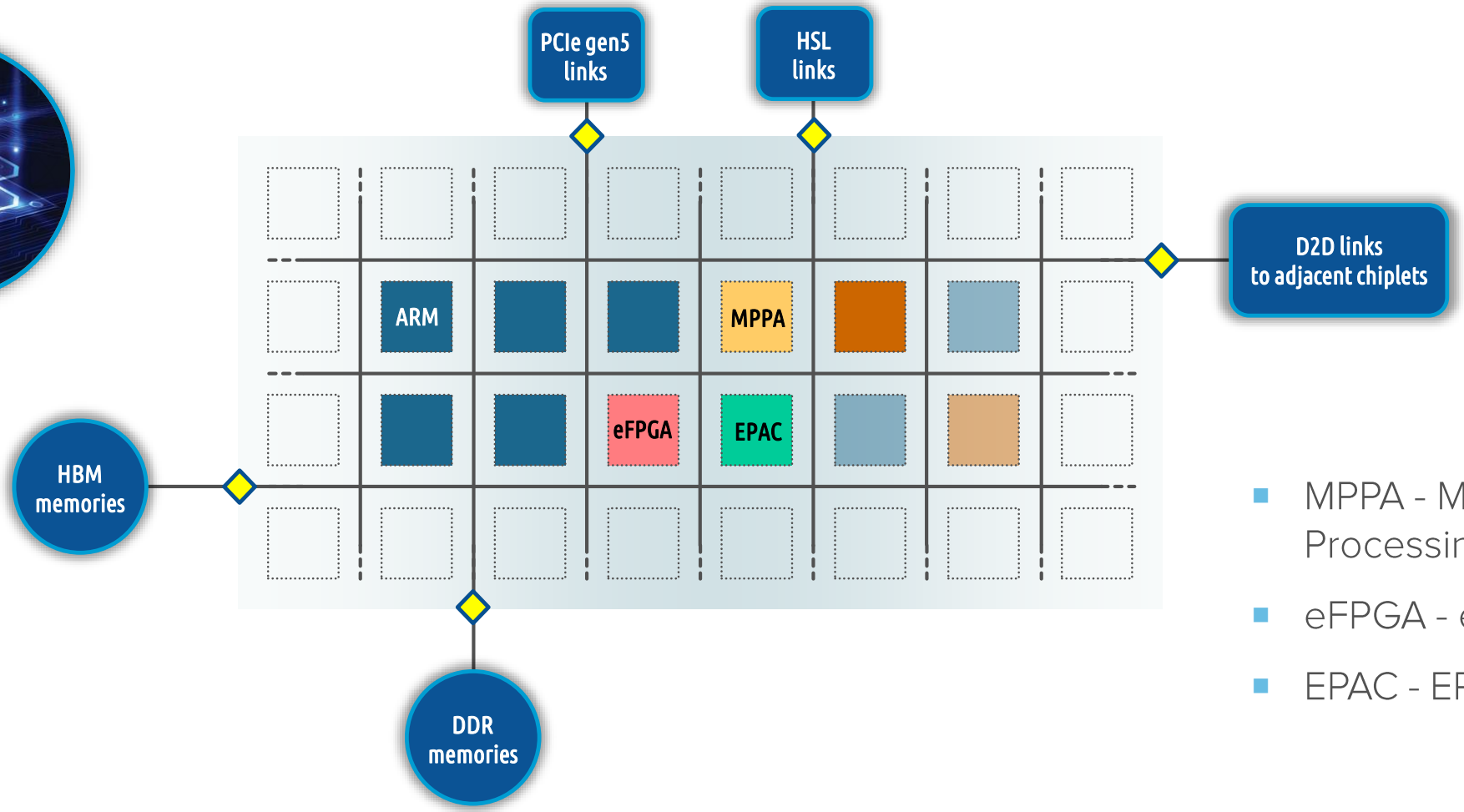
- Common platform and global architecture stream
- HPC general purpose processor stream
- Accelerator stream
- Automotive platform stream

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 826647

EPI STREAMS

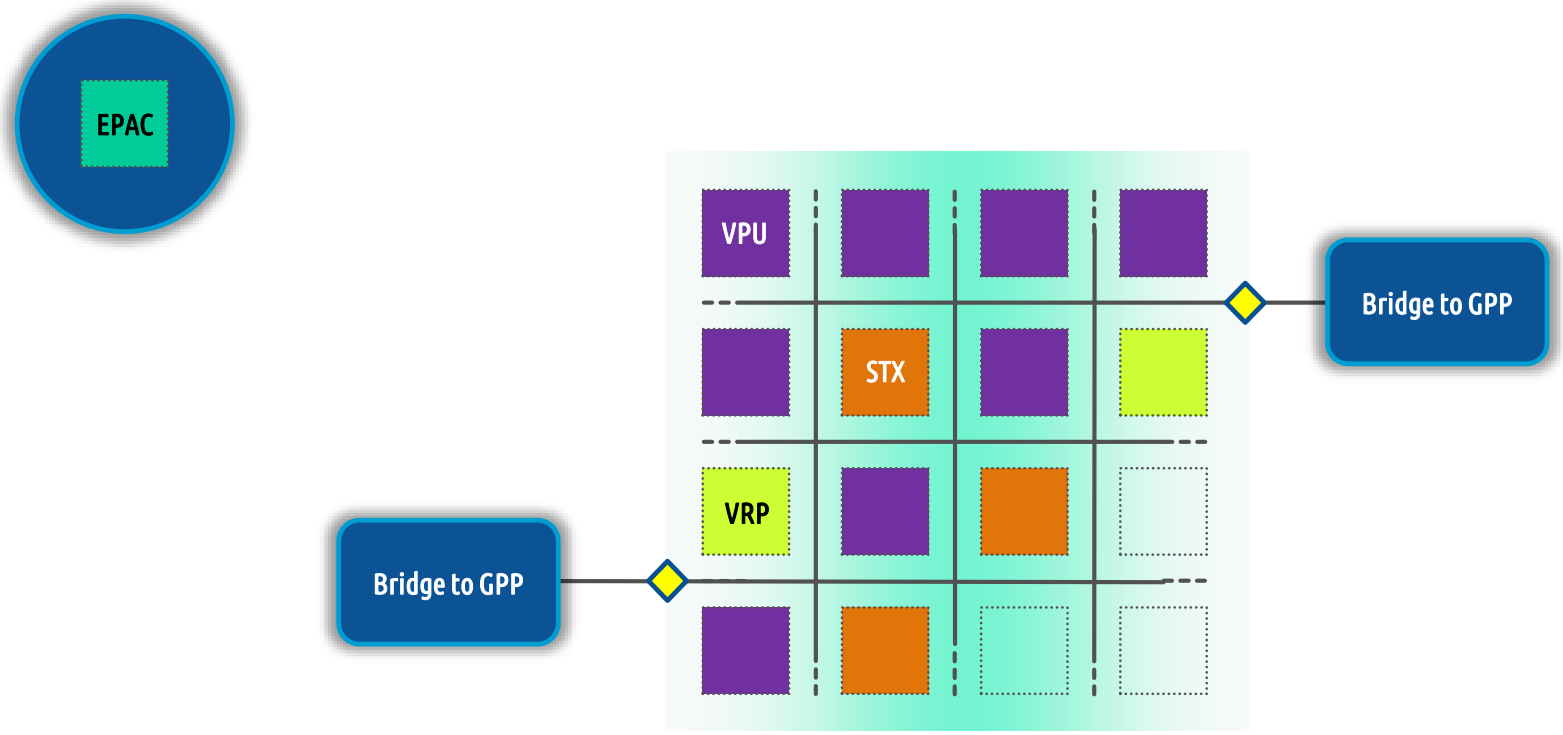


GPP AND COMMON ARCHITECTURE



- MPPA - Multi-Purpose Processing Array
- eFPGA - embedded FPGA
- EPAC - EPI Accelerator

EPAC – RISC-V ACCELERATOR

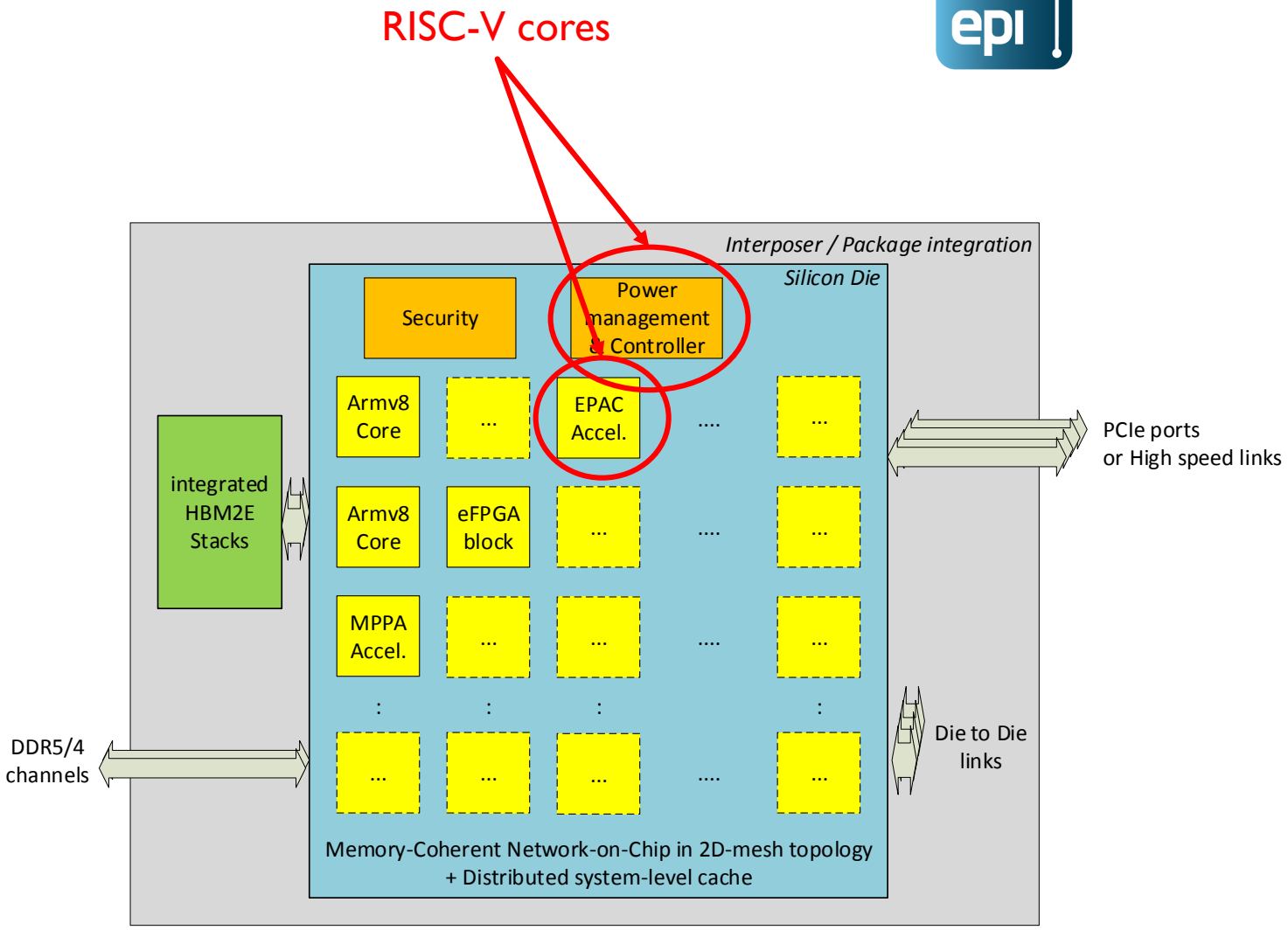


- EPAC - EPI Accelerator
- VPU – Vector Processing Unit
- STX – Stencil/Tensor accelerator
- VRP - VaRiable Precision co-processor

EPI HARDWARE

GENERAL ARCHITECTURE

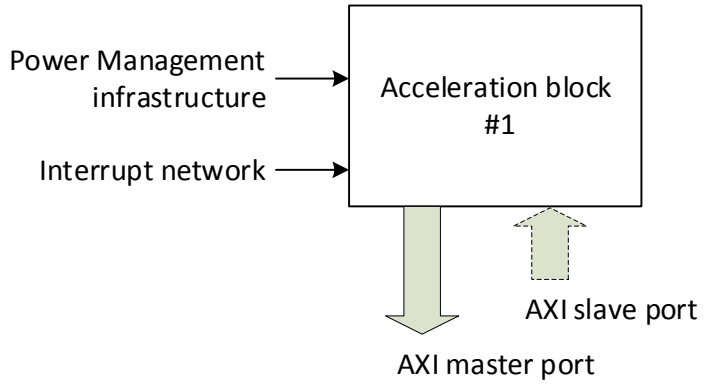
- Memory-coherent NoC connects
  - Array of computing units (CU)
  - Memory and I/O controllers
  - Bridge to links
- High speed links
  - D2D links to connect on-package dies
  - HSL links to connect on-board packages
- Top level infrastructures
  - Power management & controller
  - Security

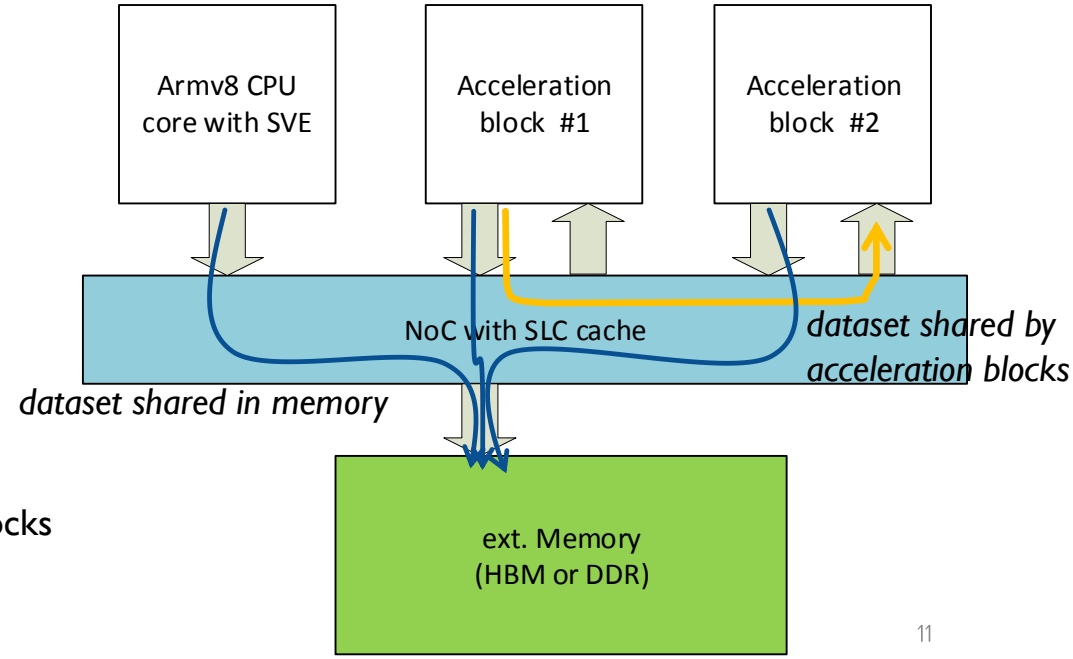


NoC: network on chip
HSL: High speed links (with memory coherent support)

GENERAL ARCHITECTURE (#2)

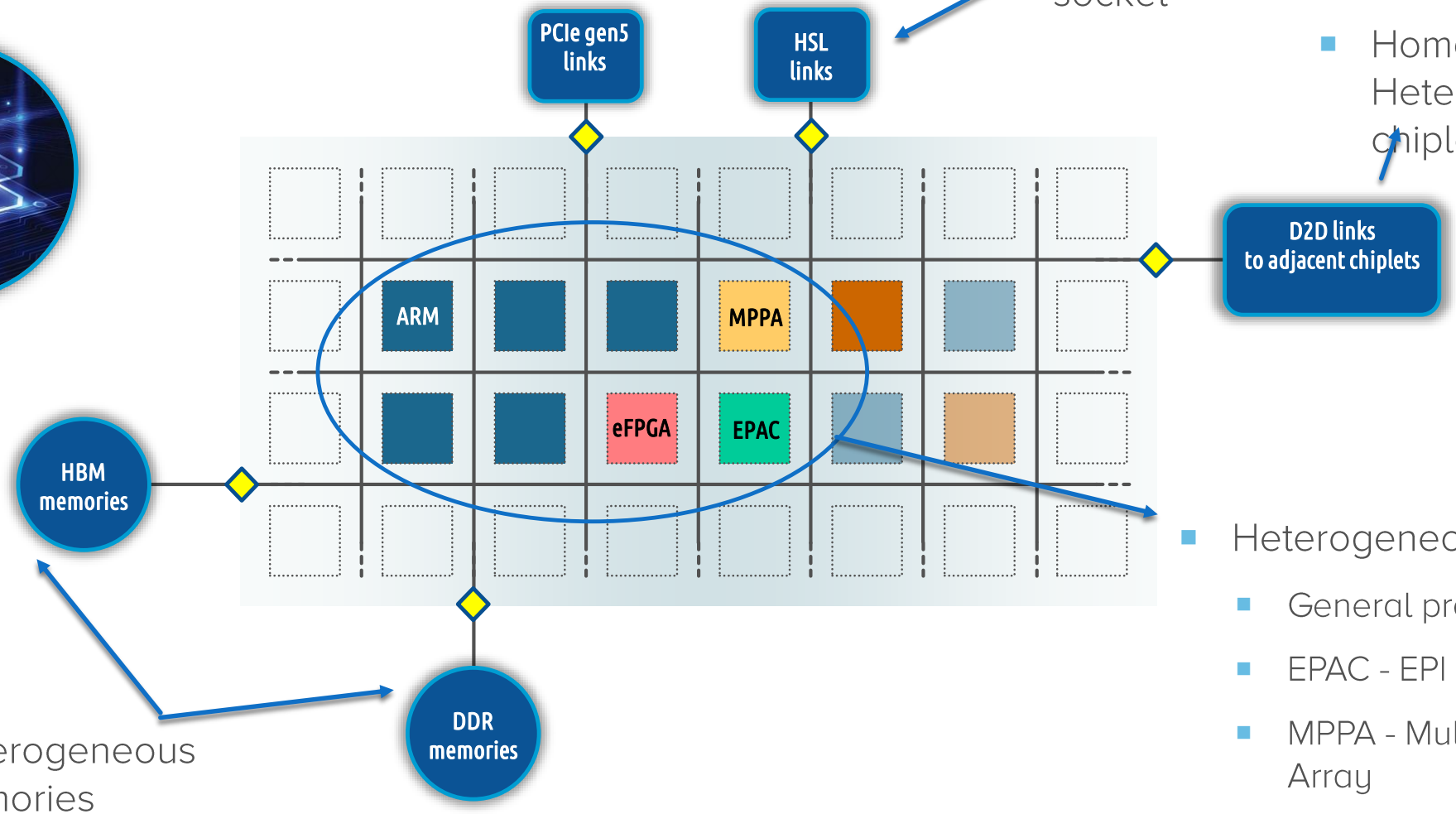
- Interfaces to connect acceleration functions to the NoC
  - Data access and sharing throught AXI ports
  - Receiving interrupt
  - Power management
- Enable memory-centric computations
  - Same copy of dataset is shared by multiple CUs
    - In the ext. memory (DDR or HBM) cached by SLC cache
    - In the local scratch memories near or local the acceleration blocks
  - System MMU to provide same virtual memory view





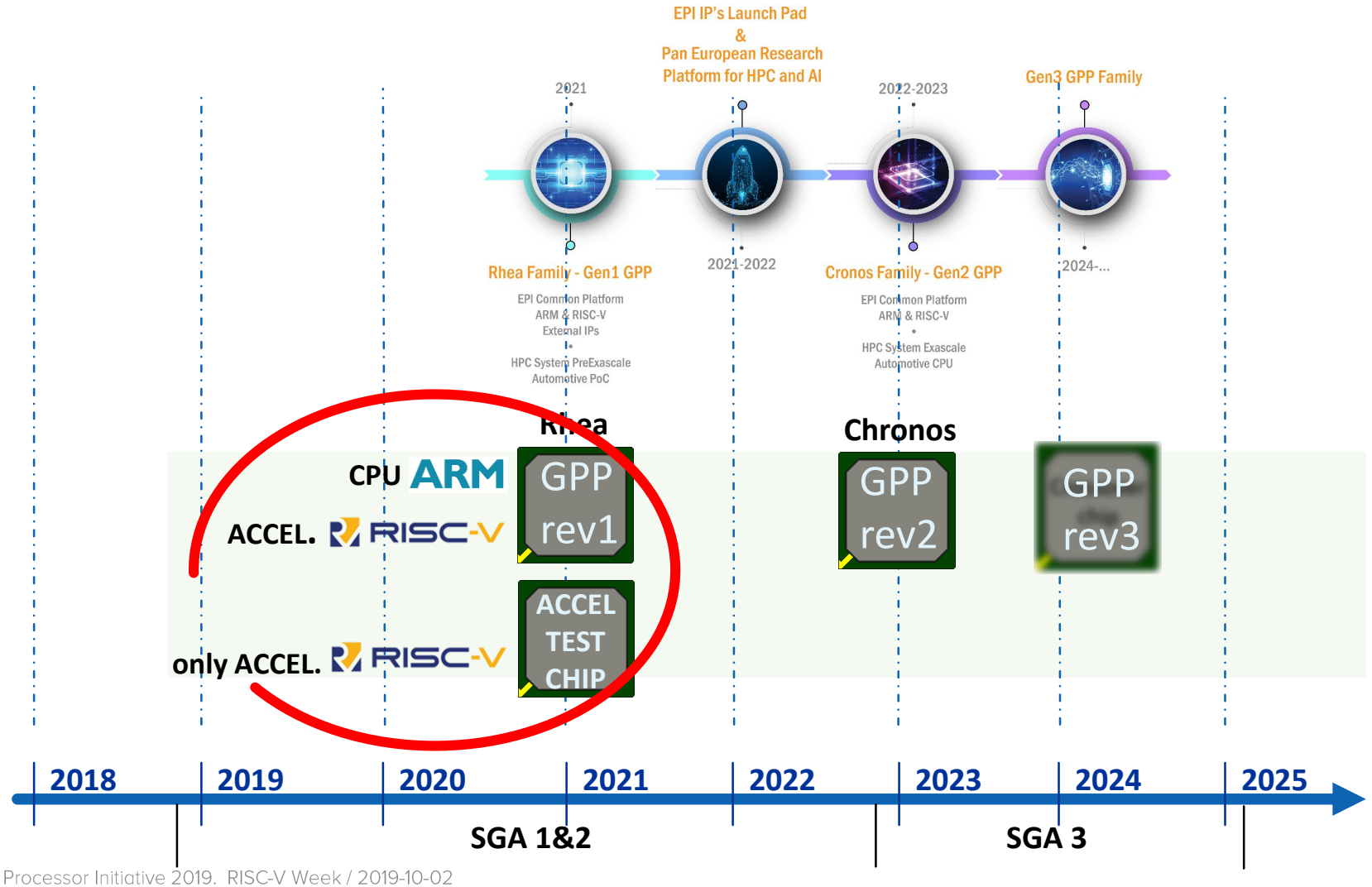
CU: Computing Unit; either Armv8 core with SVE or the EPAC/MPPA acceleration blocks
SLC: System Level Cache; a last-level cache before ext. memories

HETEROGENEOUS INTEGRATION

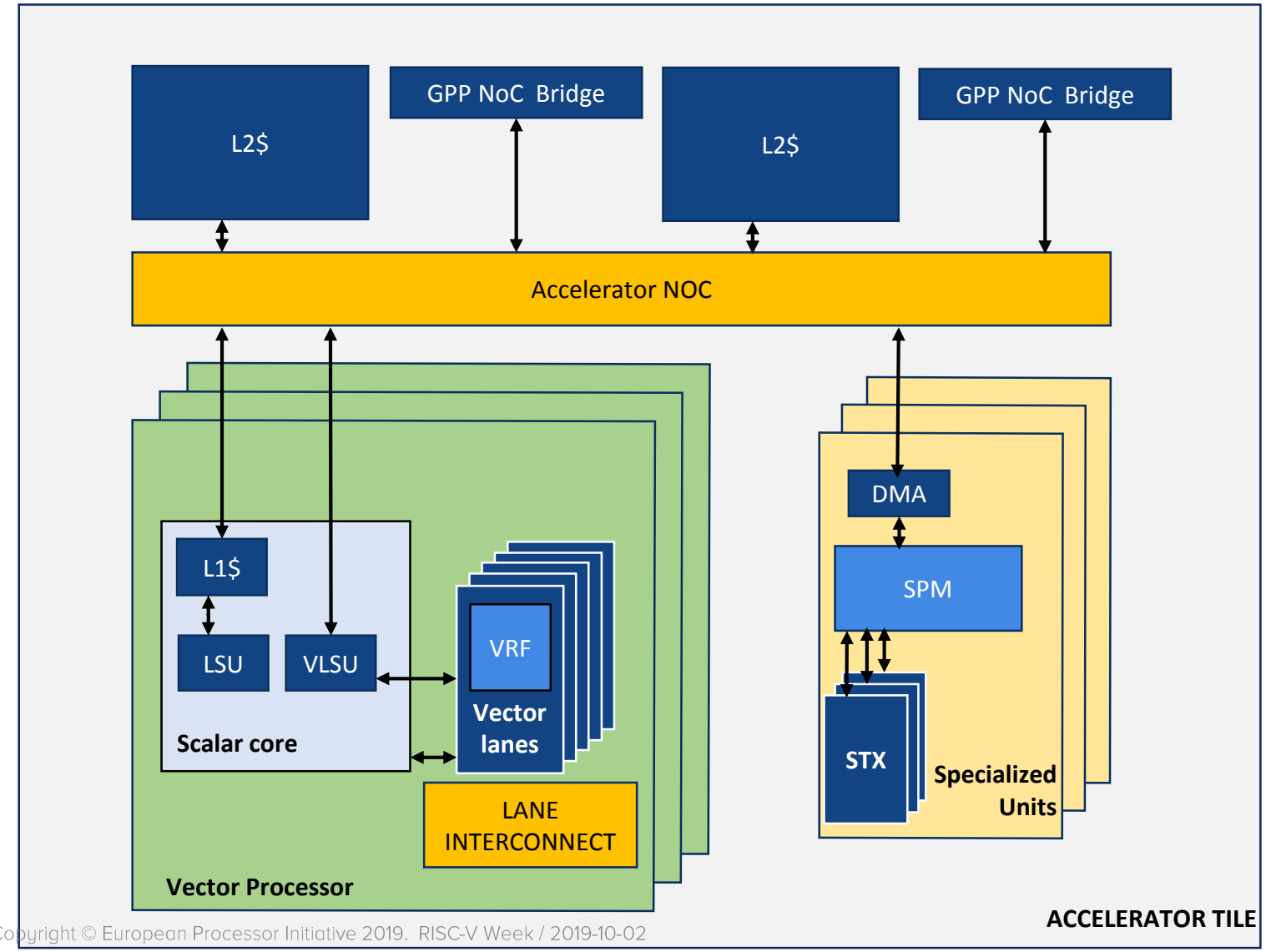


- Heterogeneous socket
- Homogenegous/ Heterogeneous chiplet
- Heterogeneous memories
- Heterogeneous SoC:
  - General processing core
  - EPAC - EPI Accelerator
  - MPPA - Multi-Purpose Processing Array
  - eFPGA - embedded FPGA

ACCELERATOR STREAM IN EPI ROADMAP



EPAC ARCHITECTURE VIEW



- The Vector Lanes act as tighlty coupled acceleration units to the scalar core
- The Specialized Units act as loosely coupled acceleration units to the scalar cores
- Up to 8 vector processors per tile
- Up to 8 STX units per tile
- Shared L2 cache banks
- Cache coherent NoC

EPI SOFTWARE

EPI SOFTWARE AMBITIONS IN HPC

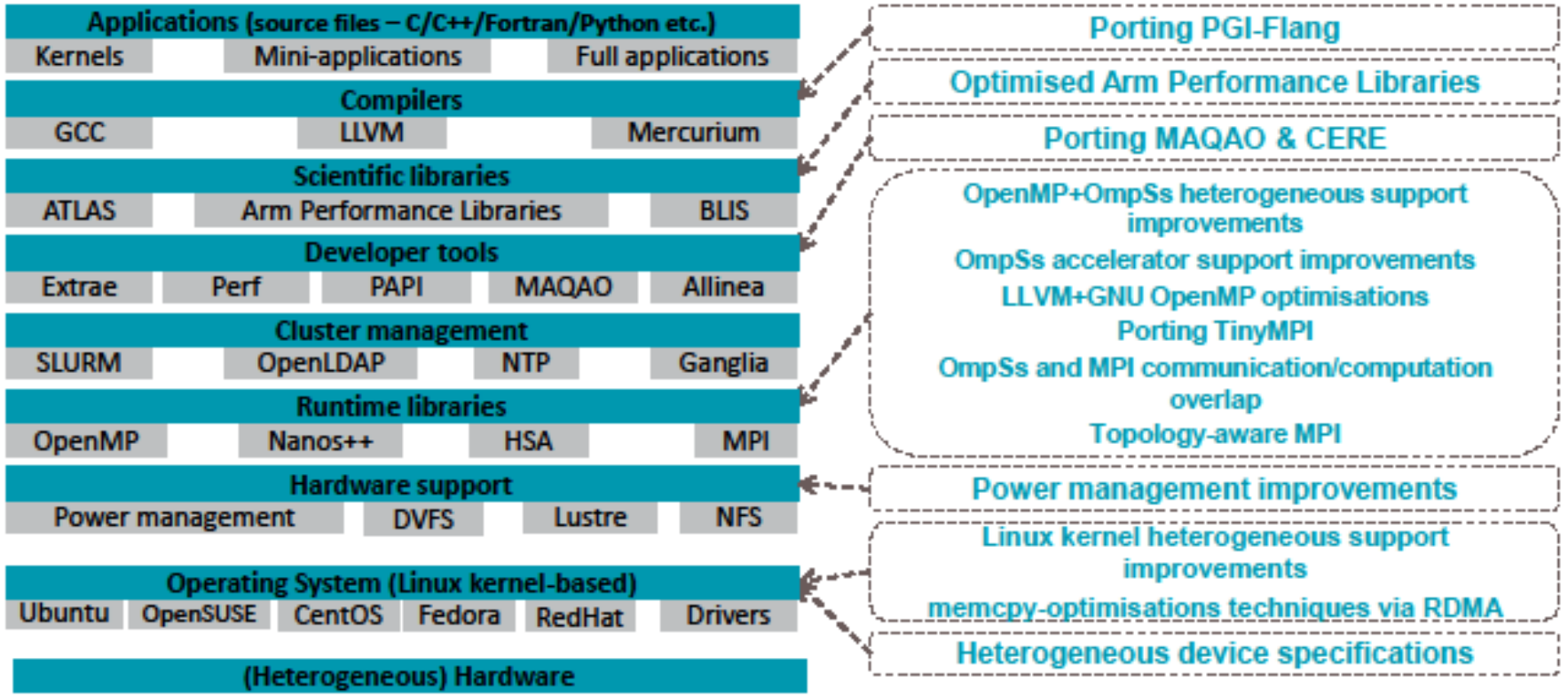
- Complement the already existing software to supply an entire HPC production stack
- Deployment/administrative side
  - Securing the node
  - Power managing the node
  - Booting the node
  - (Remote) controlling the node
  - Running a Linux distribution on the node
  - Managing various nodes in a large system
  - Monitoring & accounting various nodes in a large system
- End-user side
  - Compiling software for the General Purpose Processor
  - Compiling software for the Accelerators
  - Combining the use of GPP & Accelerators in a software
  - Leveraging standard libraries tuned for the node
  - Running the software on a node
  - Running the software on multiple nodes in a system
- Automotive requirements
  - Security, power
  - Predictable performance for autonomous driving
    - Multiple inputs, complex models

STANDING ON THE SHOULDERS OF SUCCESSFUL EUROPEAN PROJECTS

- Mont-Blanc 3 helped create and stabilize most of the foundations of a full HPC software stack on Arm

arm

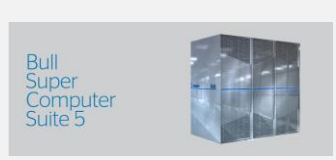
HPC Arm Software Stack



+ Contribution to OpenHPC (from 1.2)

EPI ECOSYSTEM
(INCLUDING POTENTIAL OUTSIDE PARTNERS)

Full HPC Environment for the Reference Platform



Co-design exploration space

Automotive eHPC software support

Programming tools & Libraries: LLVM/GCC with OpenMP; OpenMPI; FFTW; BLIS; OpenBLAS, …

Security, Low-level software, power management

Linux Operating System

EPI Processor

EPI Reference Hardware

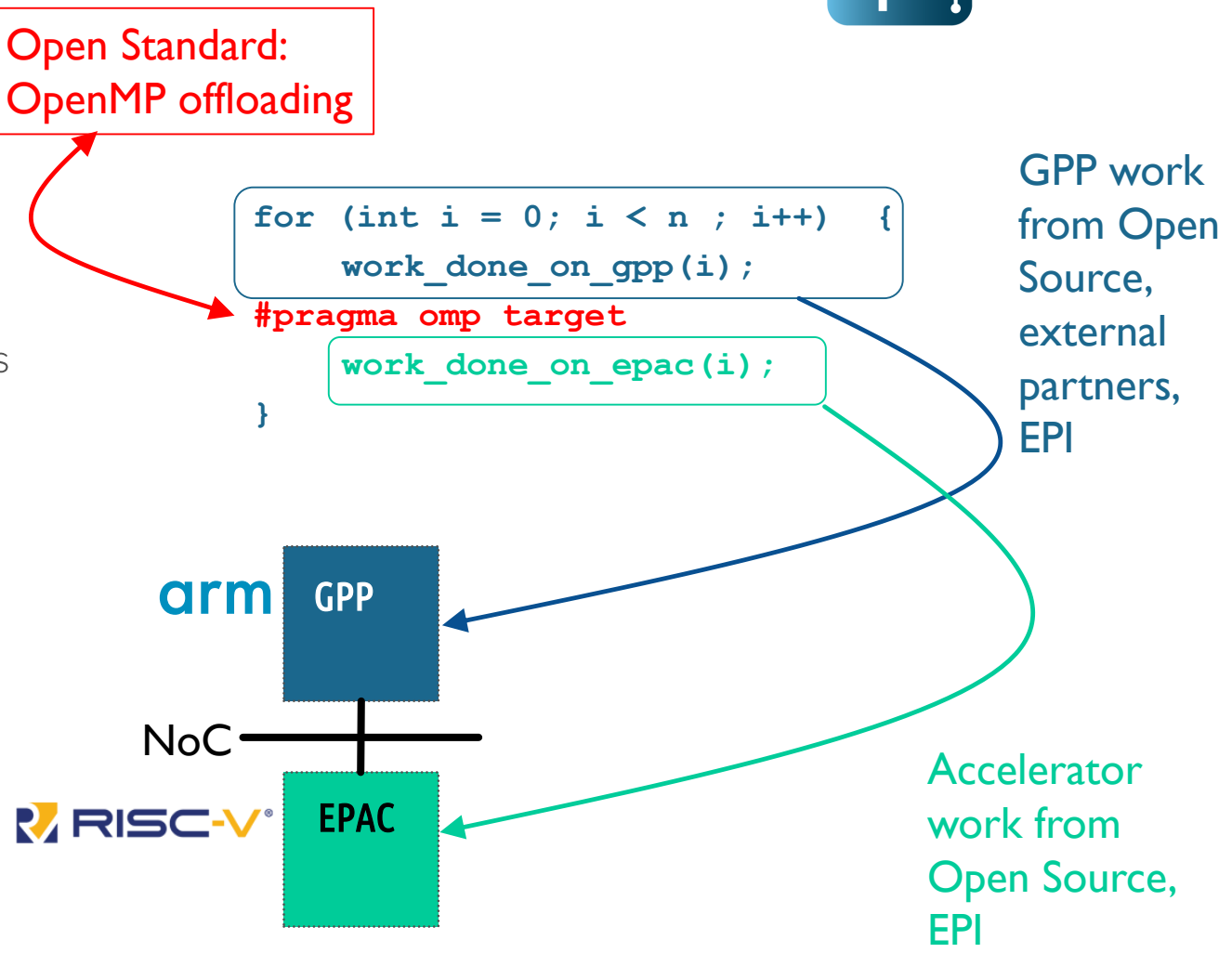
… and those supporting the ecosystem & EPI, including EPI partners, external partners & OpenSource contributors

EXPANDING TO RISC-V

- The RISC-V architecture is used extensively in EPI
  - EPAC accelerator, Power Management, …
- It is fully open and not reliant on one company for its definition
- The software ecosystem is not yet has developed as the Arm one
  - Mont-Blanc projects were instrumental in maturing the Arm ecosystem
- EPI software work includes work to bring RISC-V closer to the need of a production-ready general-purpose processor
- MPI work on RISC-V & hybrid
- OpenMP runtime on RISC-V
  - For offloading & native mode
- Compiler work
  - Including OpenMP SIMD
- Using RISC-V in the industrial world as a full-fledged, linux-capable processor and not just a microcontroller
  - Real-life use to strengthen the software

ACHIEVING AMBITIONS – END-USER

- Compiling, combining
  - Leveraging work from existing Open Source projects: GNU, LLVM
  - Choosing open standard over proprietary solutions
    - Emphasis on OpenMP in the project
    - OpenVX for automotive
  - Working with external partners
    - Arm work in LLVM & libraries for the GPP
  - EPI adding missing pieces to fully exploit the Common Platform design
    - Better vectorization in EPAC
    - OpenMP offloading
    - OpenMP SIMD

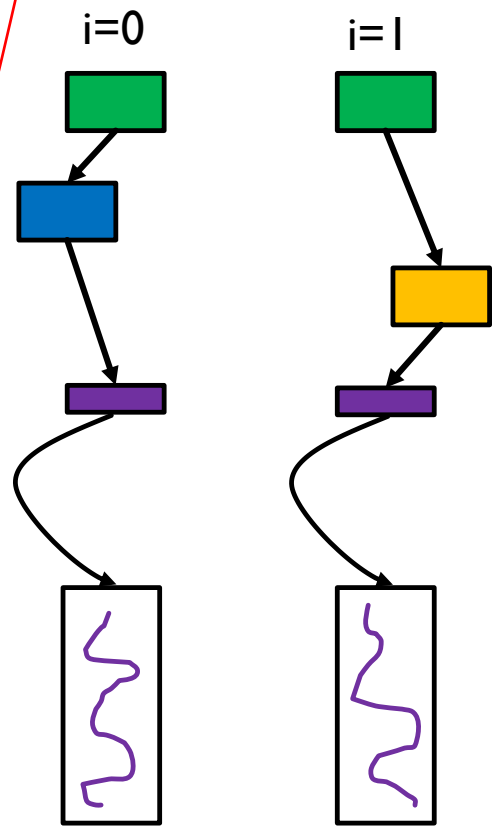


ACHIEVING AMBITIONS – END-USER

Open Standard: OpenMP SIMD

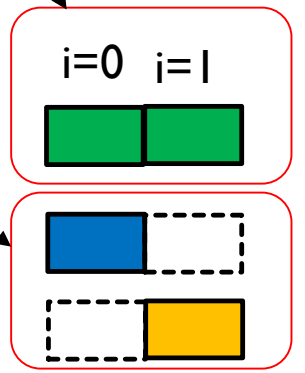
```
#pragma omp simd
for (int i = 0 ; i < n ; i++) {
  green_work(i);
  if (cond(i)) {
    blue_work(i);
  } else {
    orange_work(i);
  }
  purple_function(i);
}

#pragma omp simd declare
void purple_function(int i) {
  (…)
}
```



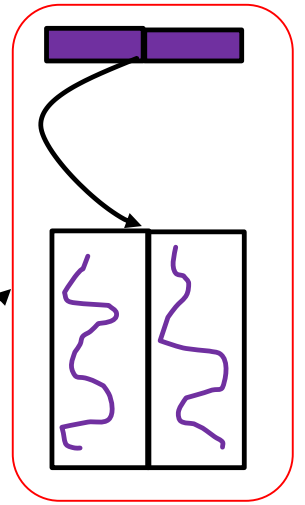
State-of-the-art (SSE, AVX, …)

Enabled by ARM SVE, RISC-V "V"



fully SIMD vectorized

masked SIMD

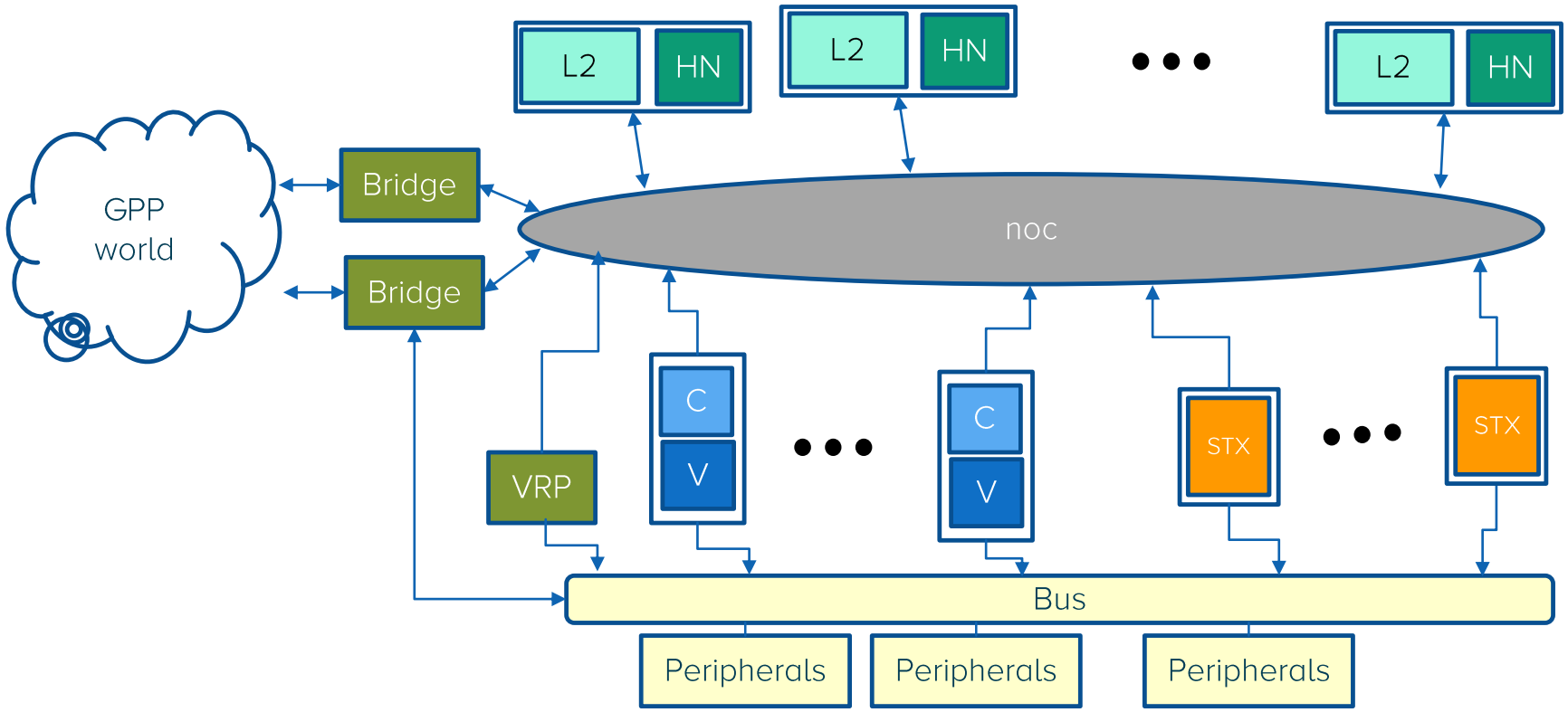


Enabled by OpenMP 4.x

OpenMP SIMD "declare" to generate vector function call & vector function body

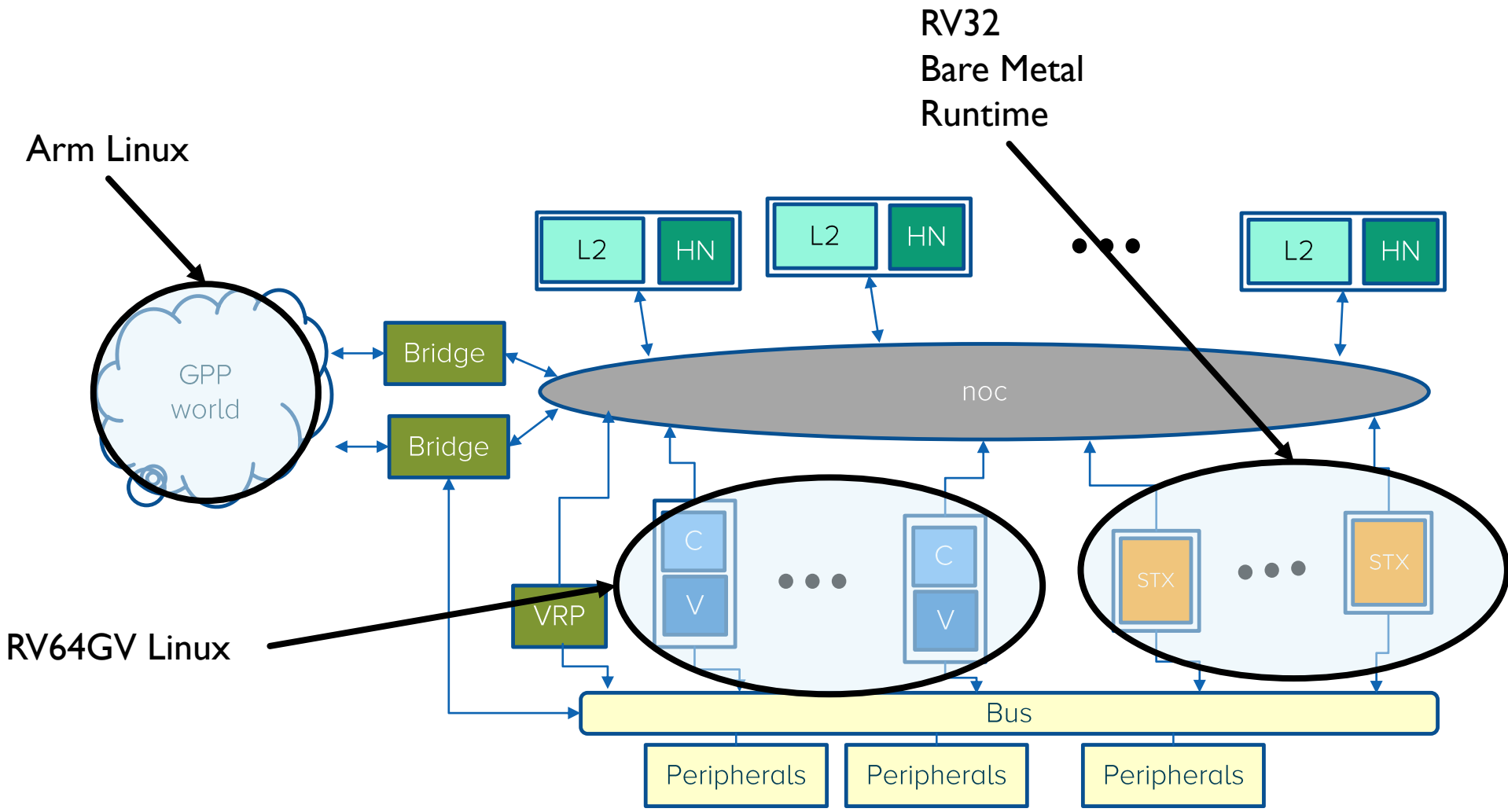
EPAC TILE ARCHITECTURE (1)

- Tile: Vector cores, STX, VRP, Shared L2 cache
- Vector Core :
  - RISC-V Vector extensions
- STX
  - RISC-V, NTX + Stencil
- VRP
  - RISC-V + Extended precision FPU
- Shared cache
  - ~ small; 8 banks
- Tile NOC
  - BW: ~ 1 line / cycle
- Bridge
  - I/O coherence to GPP



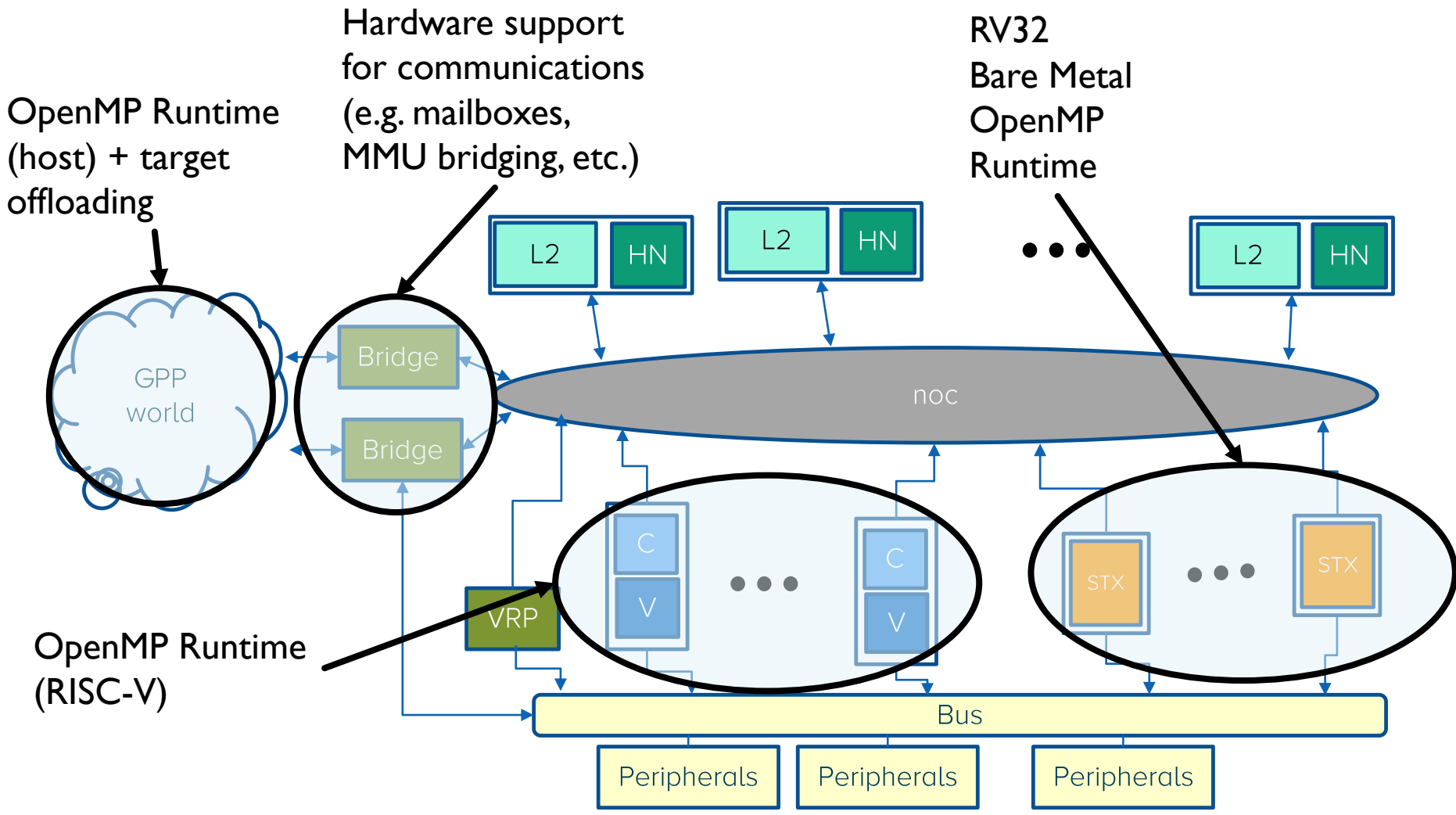
EPAC TILE ARCHITECTURE (2)

- Multiple OSes and runtimes running simultaneously
- Some accelerators Linux-capable, some targeting bare-metal runtime



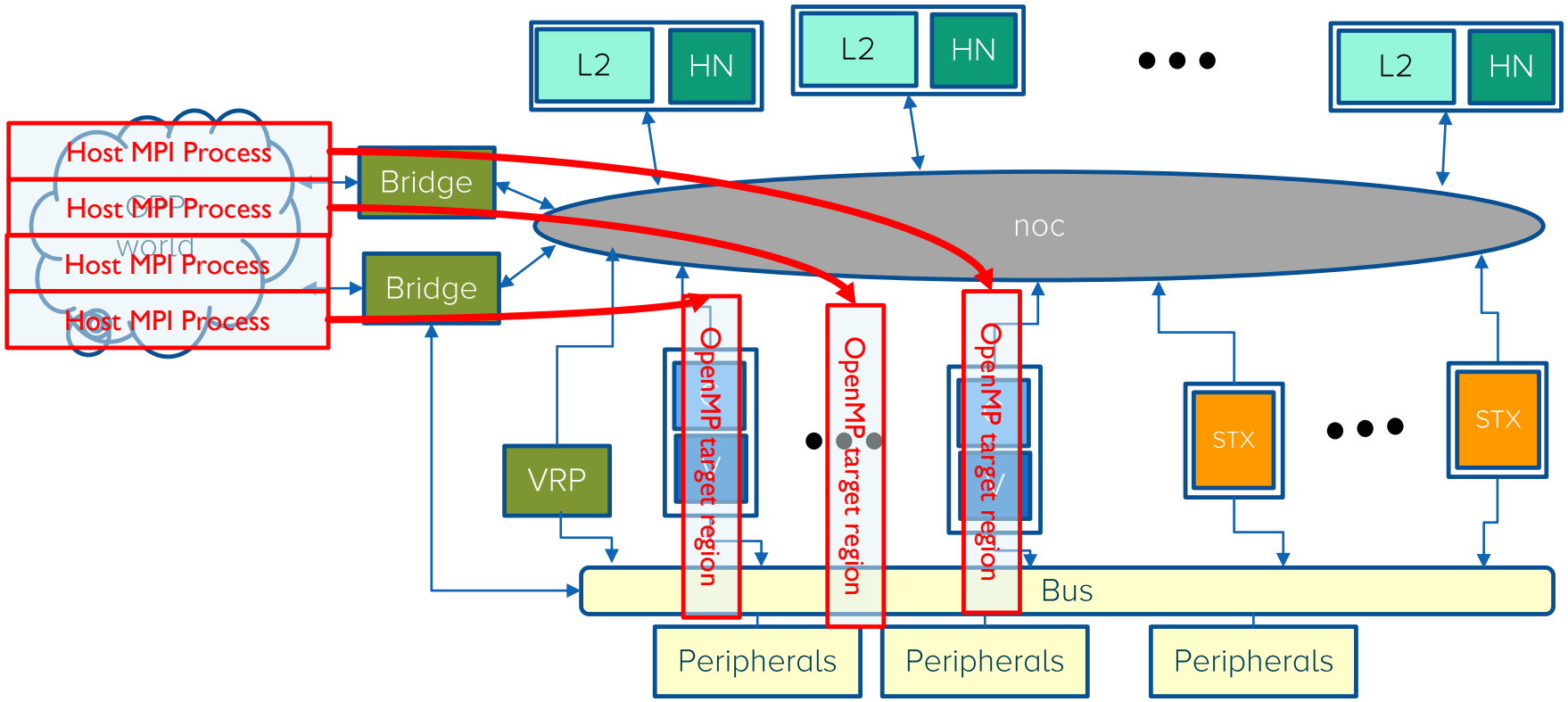
EPAC TILE ARCHITECTURE (3)

- Goal is to fully support OpenMP offloading (“target”) for all accelerators
- Also, support zero-copy acceleration
  - Sharing application’s virtual space between the Arm host and the RISC-V accelerators



EPAC TILE ARCHITECTURE (4)

- Programming model includes for instance Host MPI + individual OpenMP offloading (illustrated)
- Accelerators can be seen as one large accelerator, or multiple smaller accelerators
- Also working on MPI for Linux-capable accelerators for heterogenous MPI



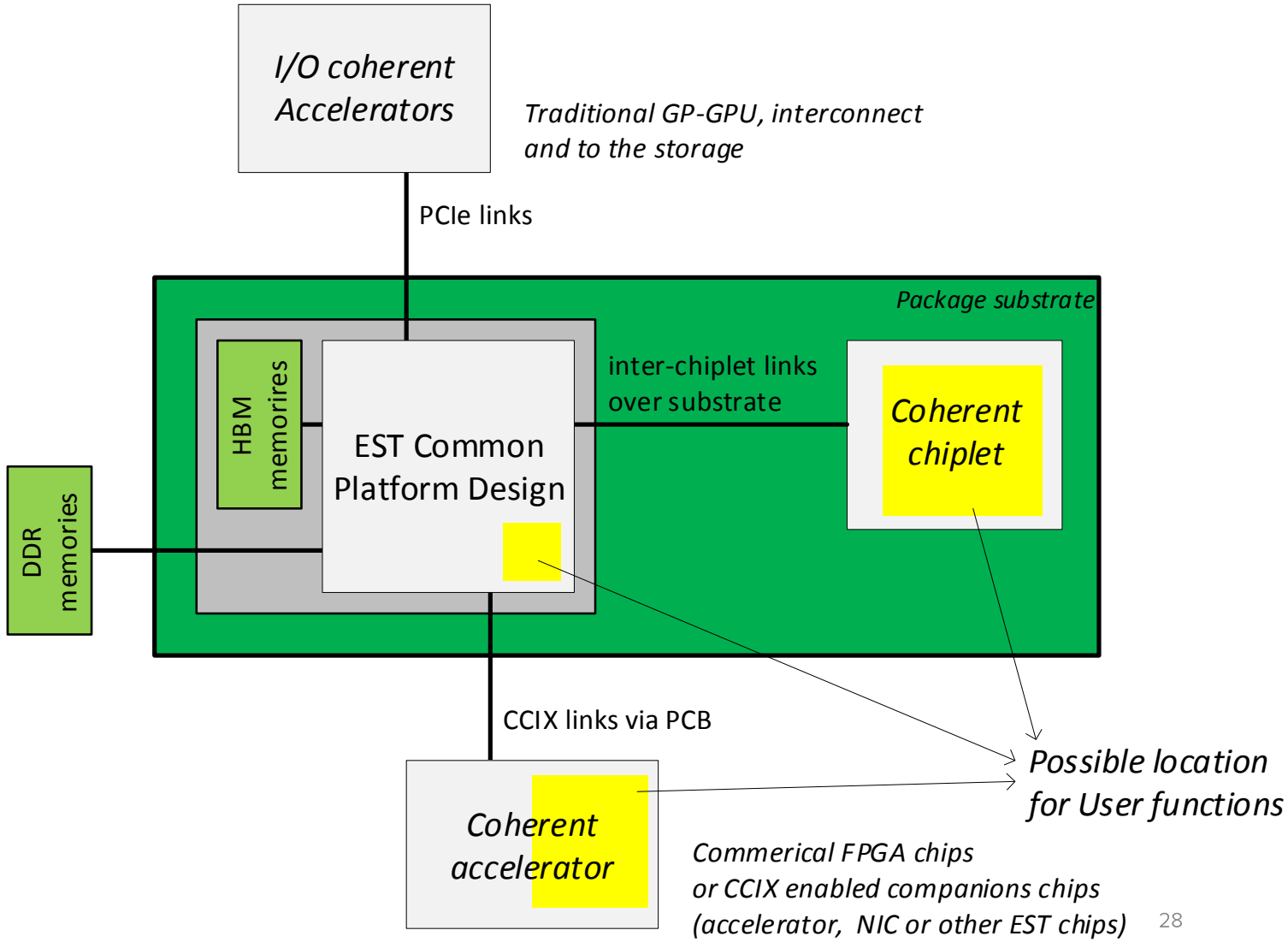
RISC-V IN EPI

- RISC-V already boots on its own
  - HiFive Unleashed, QEMU
- Needs to port some basic infrastructure
  - Self-hosted OpenMP, MPI, etc.
  - Vector, STX, VBR compiler(s) & tools, etc.
- Needs to adapt some existing infrastructure to the EPI architecture
  - Booting/initialization (OpenSBI)
  - Kernel, bare-metal runtimes
  - OpenMP offloading
  - Heterogeneous MPI
- Needs to figure out how to efficiently implement some aspects of our ambitions
  - Booting, securing, avoiding conflicts
  - Bridging NoCs between Arm and RISC-V domains
    - Physical addressing
    - Communications mechanisms
  - Enabling virtual memory sharing between Arm and RISC-V domains
    - MMUs with different properties
    - I/OMMU
  - Unforeseen issues …
- Also, more independent RISC-V such as PowerManagement

COMMON PLATFORM

HETEROGENEOUS INTEGRATION

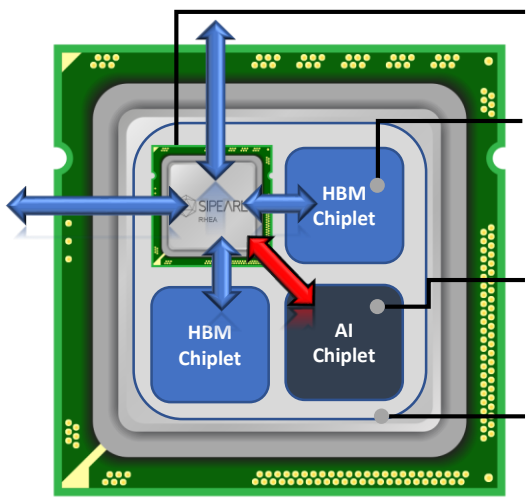
- Integrating customized functions at different levels
  - EPI accelerator IPs today is integrated in RheaR1 design
- Leverage the HW infrastructure for other accelerators or implementations
  - Coherent chiplet (in-package) via D2D interface
  - Coherent external accelerators via CCIX
- Leverage the software work to facilitate exploitation
  - OpenMP offloading



SIPEARL – Common Platform

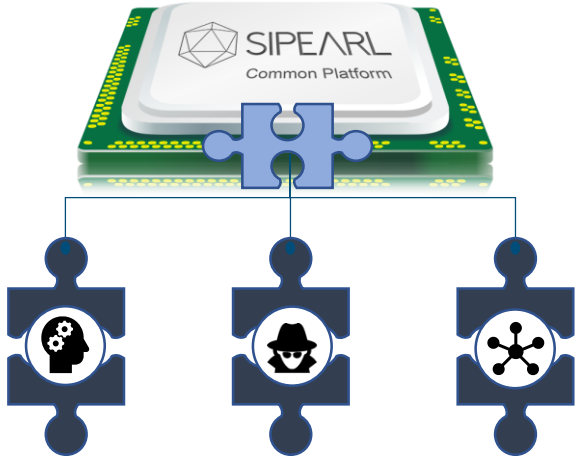
The Common Platform will leverage our investment due to limited time-to-market and costs to develop our own platform, enabling AI start-ups to make huge savings on licences

Interposer function



- **SIPEARL RHEA chip**
- **Embedded memory (like HBM)**
- **Low cost chip from AI start-ups**
- **Interposer for integration**

Value proposition



AI Start-up accelerator | **Crypto Accelerator** | **Quantum or photonics accelerator**

+60 potential customers

GRAPHCORE, LIGHTMATTER, thinci, PEZY Computing, cerebras, Esperanto Technologies, adapteva, DEEPSCALE, WAVE COMPUTING, Think Force, SiFive, AMPERE, TENSTORRENT, KONIKU, groq, BITMAIN, antmicro, GREENWAVE SYSTEMS, KALRAY, Tachyum, SambaNova SYSTEMS, krtkl, TERADEEP, ChipIntelli MAKE AI REAL, KNOWM, AIMOTIVE, DEEP VISION, Kneron, LIGHTELLIGENCE, GREENWAVES TECHNOLOGY, MYTHIC, Cambricon, gyrfalcon technology, another brain, NOVUMIND, brainchip

WE ACCELERATE ACCELERATORS !!!!

SIPEARL

THANK YOU

ROMAIN.DOLBEAU@EUROPEAN-PROCESSOR-INITIATIVE.EU
ROMAIN.DOLBEAU@ATOS.NET