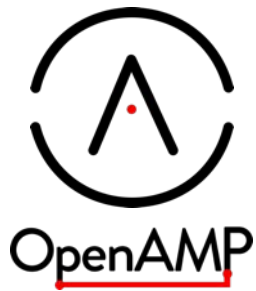# OpenAMP: "**Open A**symmetric **M**ulti-**P**rocessing" Project

Runtime coexistence and collaboration

Runtime hardware resource assignment

Resource sharing and IPC between runtimes

Control mechanisms to start and stop runtimes

Typical system: Linux + RTOS on one system-on-chip
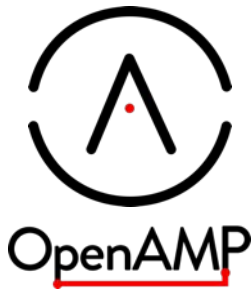
www.openampproject.org.
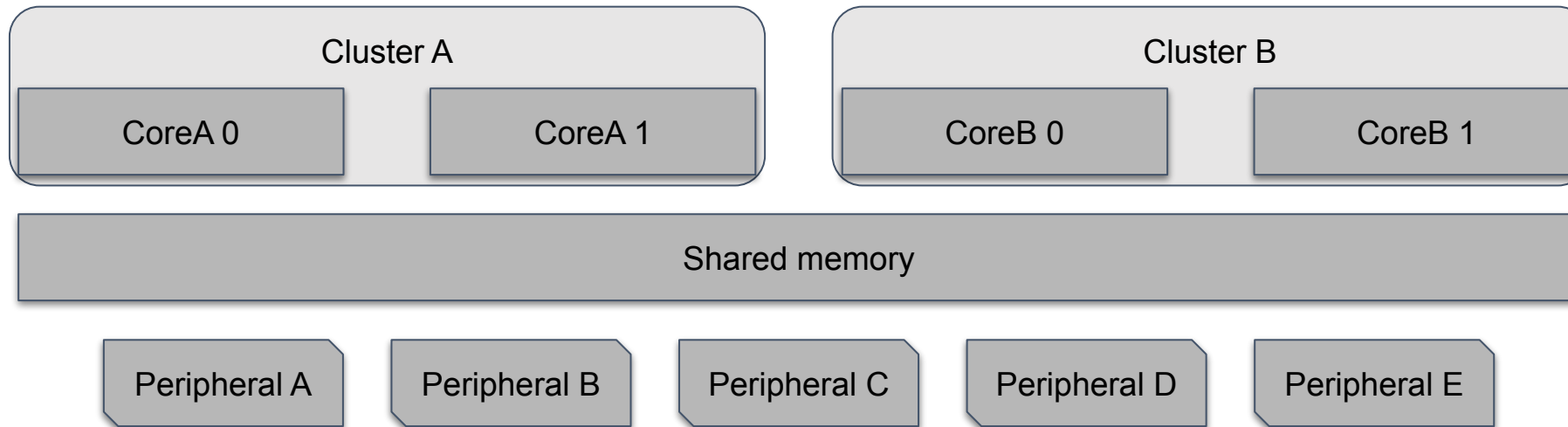
OpenAMP Project Intro
Standardizing Asymmetric Runtime Integration

OpenAMP Embedded Targets

Modern Embedded Targets integrate multiple HW resources, e.g. multiple core clusters, shared memory and peripherals

Cluster A		Cluster B	
CoreA 0	CoreA 1	CoreB 0	CoreB 1

Shared memory

Peripheral A	Peripheral B	Peripheral C	Peripheral D	Peripheral E

OpenAMP Embedded Runtimes

Embedded Targets have multiple Runtimes that need to collaborate

| Linux + Apps	RTOS App	Bare Metal App

OpenAMP HW Assignment

The HW resources need to be assigned into Runtime Domains

Linux + Apps	RTOS App	Bare Metal App

CoreA 0	CoreA 1	CoreB 0	CoreB 1

Memory Linux	Memory Shared	Memory RTOS	Memory Shared	Memory Bare Metal

Linux Domain

RTOS Domain

BM Domain

Peripheral A
Peripheral B
Peripheral C
Peripheral D
Peripheral E
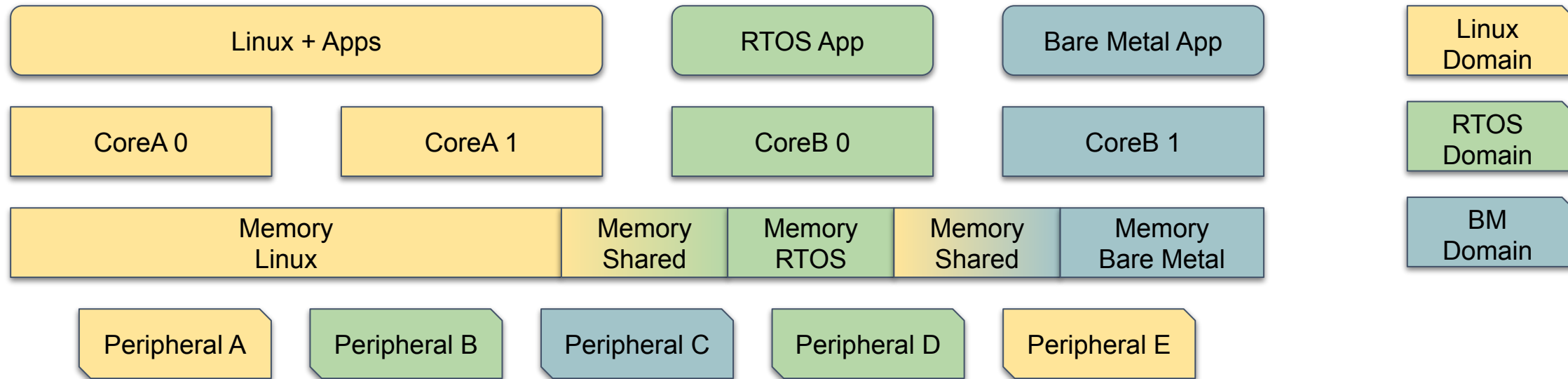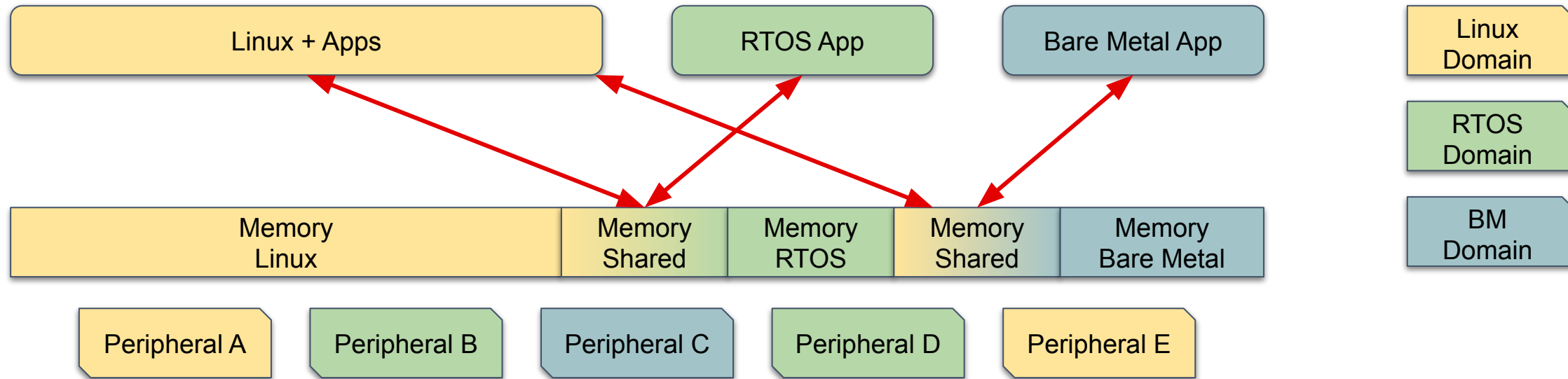
OpenAMP Runtime Control

The Runtimes need to be managed,
e.g. loaded into memory and started
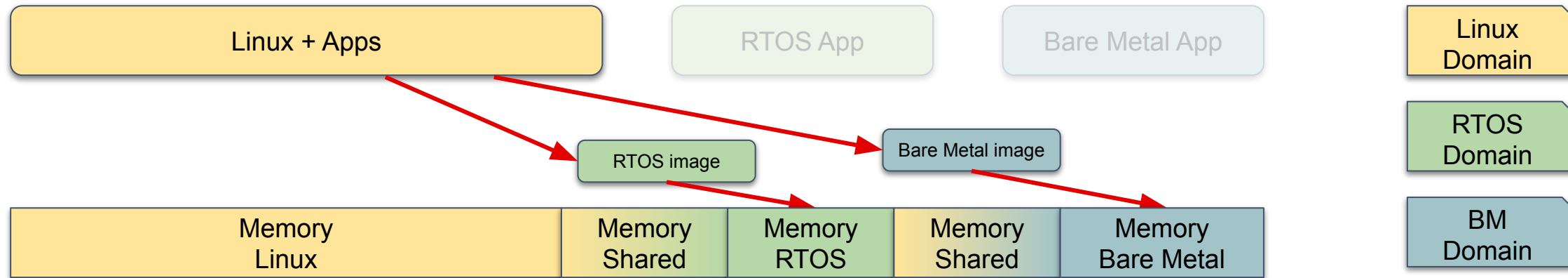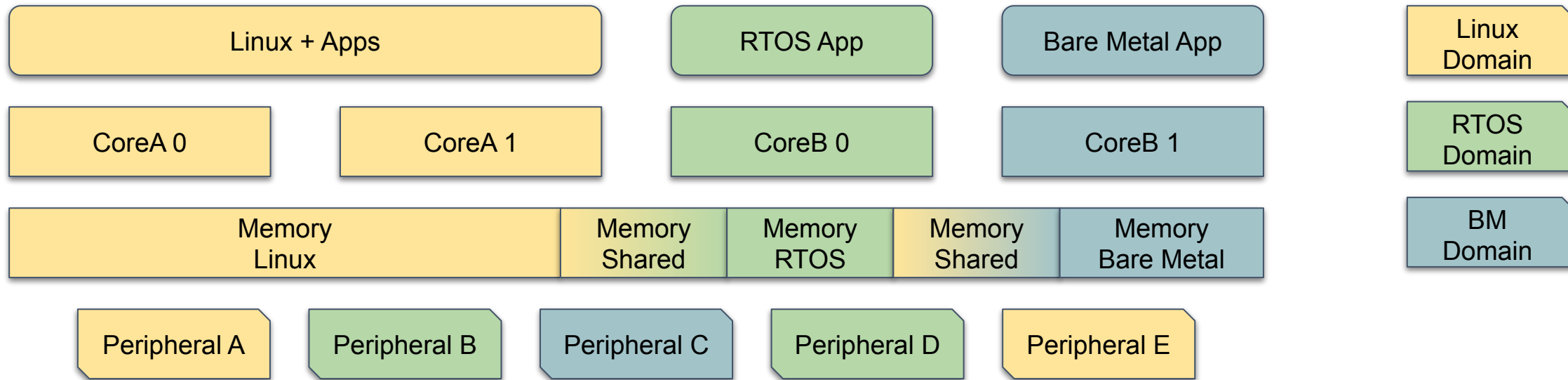
OpenAMP Resource Sharing and IPC

The Runtimes need to share data and services

OpenAMP Mission

OpenAMP provides standards, runtime libraries and tooling built on top of existing open source projects to simplify runtime collaboration

Linux + Apps	RTOS App	Bare Metal App	Linux Domain

CoreA 0	CoreA 1	CoreB 0	CoreB 1	RTOS Domain

Memory Linux	Memory Shared	Memory RTOS	Memory Shared	Memory Bare Metal	BM Domain

Peripheral A	Peripheral B	Peripheral C	Peripheral D	Peripheral E

Check it out and get involved!

- Community Project Website
  - www.openampproject.org
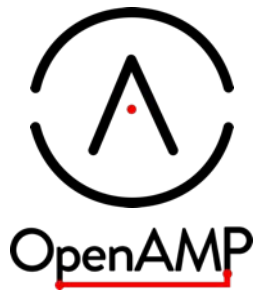
-

- Contributing companies:

Thank You

More details

- Consider adding a slide with 1 bullet per other use case, such as baremetal-baremetal, RTOS-RTOS, etc.

Standardizing SW for Heterogeneous Environments

- Today, most heterogeneous environments are cobbled together ad-hoc
  - E.g. Everybody coming up with their own shared memory scheme for passing messages
  - E.g. The address and configuration of device addresses are hard coded in many places
- There is a need to standardize how runtimes interact
  - Configuring the runtimes
  - Managing the runtimes
  - Passing messages between runtimes
  - Share resources between runtimes
- Open source implementation is fastest way to standardization
  - Based on already existing open source projects

OpenAMP is a Linaro Community Project solving these kinds of problems

OpenAMP Philosophy:
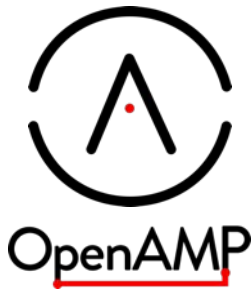# Focus on Enhancing Existing Standards and Code

Prefer to avoid reinventing the wheel. Leverage & extend where possible.

- Configuration
  - Device Trees -> System Device Trees
- Low-level data sharing
  - VirtIO and rpmsg
- Life-cycle management
  - remoteproc
- High-level data sharing
  - fileIo and sockets

Audience
Who is Using OpenAMP?

- Embedded application developers
  - Need an OS and SoC independent way of interacting between applications running on multiple cores

- SoC vendors
  - Need common interfaces to allow multiple operating systems to work together

- OS and hypervisor vendors/projects
 - Need a common infrastructure across multiple SoC vendors

Examples of OpenAMP in Industry
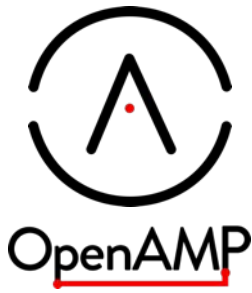
- Arm
- Active role in System Device Tree discussion
- Kalray
- Standard message passing solution within homogeneous Manycore architecture on MPPA®3 processor
- MPPA®3 as accelerator: virtio over PCIe
- Within MPPA®3 processor: virtio with shared memory
- Linaro
- Hosting OpenAMP project through Community Projects division
- Involvement in Zephyr, openamp-rp, LAVA testing
- Nordic Semiconductor
  - Bluetooth Host Controller Interface (HCI) based on OpenAMP in Zephyr
  - RPC of Bluetooth host
  - IEEE 802.15.4 radio driver serialization
- NuttX OS
  - OpenAMP integrated & available.
-

Examples of OpenAMP in Industry
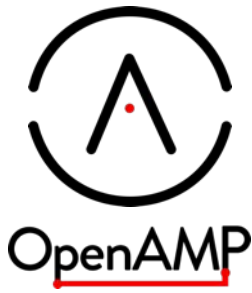
- NXP
  - Remoteproc & RPMSG used in i.MX SoC family with managing Cortex-M[X] and DSP.
  - Reviewing openAMP libraries and integrating in i.MX SW release.
  - Remoteproc to control Cortex-A core on multiple core processor (Cortex-A core running Linux and other cores running U-Boot-based bare metal code)
- Siemens Embedded
- Core for the Mentor Embedded Multicore Framework and Multicore Framework Cert product offerings.
- Expands on OpenAMP: Linux as a Remote, Large Buffer, Zero Copy, Proxy support for Ethernet
- Communication between the safe and non-safe domains in Mixed Safety-Criticality systems
- STMicroelectronics
  - Remote proc firmware loading via Linux, U-Boot or OP-TEE (signed firmware)
  - Inter processor communication protocol for muti-core and multi-Soc STM32 solutions (relying on OpenAMP library and Linux kernel)
 - OpenAMP library used with baremetal, FreeRTOS, Zephyr.
- TI
- Enhancing the Linux kernel implementation of RemoteProc & RPMsg
- Defining the wire protocol between processors
- Contributed a limited scope version of remoteproc loader into U-Boot

Examples of OpenAMP in Industry
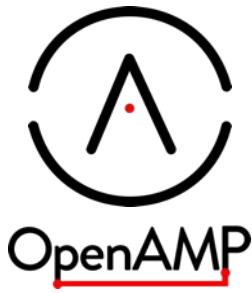
- Wind River
- To accelerate the ability of developers to create edge compute applications
- Xilinx
  - Default AMP solution for Zynq-7000, Zynq UltraScale+ MPSoC, and Versal devices
  - Cortex-A application processor units (APUs), Cortex-R real-time processor units (RPUs). Microblaze "soft" processors in programmable logic. Either APU or RPU can act as the master.
- Zephyr
  - OpenAMP integrated & available
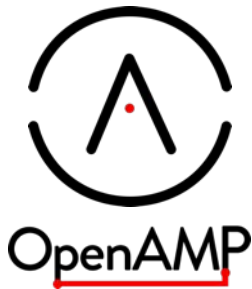 - Rpmsg service abstraction layer for multi rpmsg services support

OpenAMP Embedded Targets
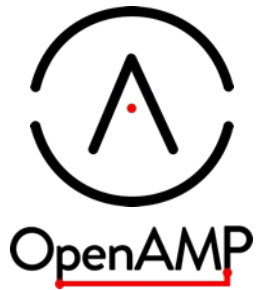
OpenAMP enables multiple runtimes to collaborate and share resources
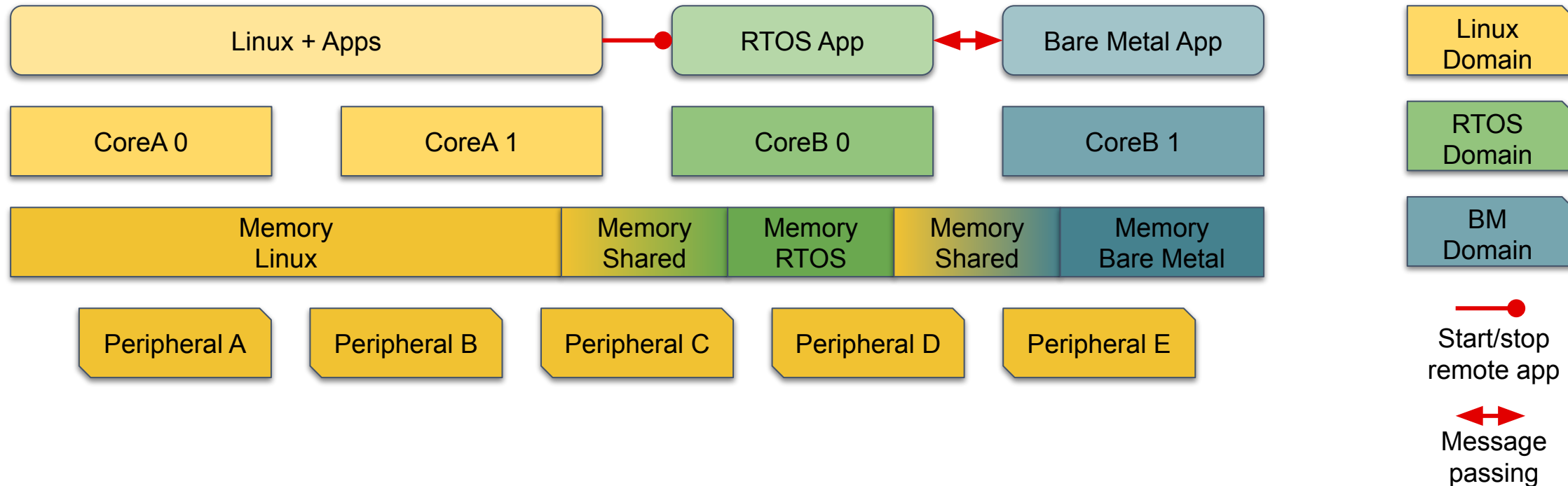
OpenAMP Overview

OpenAMP (Open Asymmetric Multi-Processing) is an open source project to enable multiple runtimes to coexist and collaborate in the same computer. OpenAMP enables runtimes to be assigned hardware resources and to be started and shutdown in an orderly fashion. OpenAMP also enables applications to access resources owned by other runtimes. To do this, OpenAMP provides high-level POSIX APIs such as file descriptors and sockets to applications although it also has low-level APIs for firmware.

Typical OpenAMP systems have a native Linux OS plus an RTOS or bare metal runtime running inside one system-on-chip. However, other scenarios are possible such as boards that have more than one microprocessor chip that share a memory system, or virtualized systems. To learn more, please visit our project website at www.openampproject.org.

OpenAMP is brought to you by our community contributors and is also sponsored by these companies: