

Process Migration for Lightweight Manycore Processors in a Distributed Operating System

João Vicente Souto

joao.vicente.souto@posgrad.ufsc.br

Parallel Computing - Computer Science
INE/UFSC, Florianópolis

September 28, 2020

Presentation Outline

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

1 Introduction

2 Virtualization

3 Migration

- Pre-Copy Migration
- Post-Copy Migration

4 Metrics

5 Architectures

- Kalray MPPA-256

6 Motivation And Justify

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

Introduction

Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Arising of **distributed systems**
- Solutions established:
 - MOSIX (1985)
 - V (1988)
 - OSF/1 AD TNC (1995)
- Transferring a process between machines
- Proposed for:
 - Load balancing
 - Fault tolerance
 - System administration
 - Data access locality



Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Arising of distributed systems
- Solutions established:
 - MOSIX (1985)
 - V (1988)
 - OSF/1 AD TNC (1995)
- **Transferring a process between machines**
- Proposed for:
 - Dynamic load distribution
 - Fault resilience
 - Improved system administration
 - Data access locality



Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Despite the research efforts, **migration has not achieved widespread use**
- Treating residual dependency is a difficult task:
 - Opened files stored in the source node
 - Opened communicators with other processes
 - Shared resources
 - Internal kernel state.
- Solution: Virtualization

Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Despite the research efforts, migration has not achieved widespread use
- **Treating residual dependency is a difficult task:**
 - Opened files stored in the source node
 - Opened communicators with other processes
 - Shared resources
 - Internal kernel state.
- Solution: Virtualization

Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Despite the research efforts, migration has not achieved widespread use
- Treating residual dependency is a difficult task:
 - Opened files stored in the source node
 - Opened communicators with other processes
 - Shared resources
 - Internal kernel state.
- Solution: **Virtualization**

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

Virtualization



Virtualization

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Create a **Virtual Version** of Computing Resources
 - CPU
 - Memory
 - Storage device
 - Network device
- Features
 - Improved previous features
 - Power management
 - Isolation
 - Security
- Two types
 - Virtual Machines (VM)
 - Containers



Virtualization

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Create a Virtual Version of Computing Resources

- CPU
- Memory
- Storage device
- Network device

- Features

- Improved previous features
- Power management
- Isolation
- Security

- Two types

- Virtual Machines (VM)
- Containers



Virtualization

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

- Create a Virtual Version of Computing Resources
 - CPU
 - Memory
 - Storage device
 - Network device
- Features
 - Improved previous features
 - Power management
 - Isolation
 - Security
- Two types
 - Virtual Machines (VM)
 - Containers



Virtual Machines (VM)

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

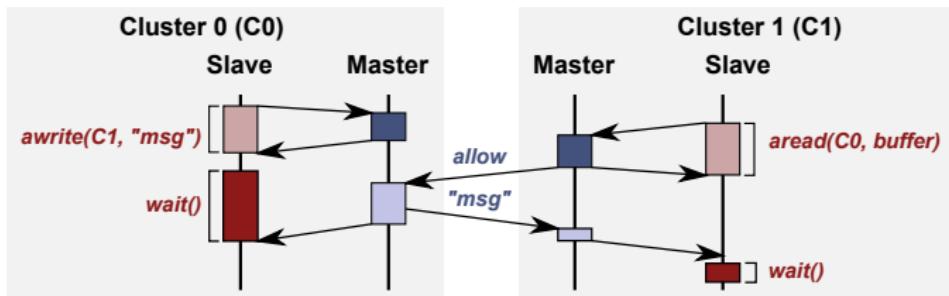
Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

- Encapsulates a whole operating system
- Depends on a Hypervisor to provide the hardware resources
 - Full virtualization
 - Paravirtualization



Migration of VMs

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

■ Memory content

- Hypervisor allocate memory
- Guest OS allocate memory
- Application requested memory

■ Disk content

- Hypervisor allocate blocks
- Guest OS and Application used blocks



Migration of VMs

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

■ Memory content

- Hypervisor allocate memory
- Guest OS allocate memory
- Application requested memory

■ Disk content

- Hypervisor allocate blocks
- Guest OS and Application used blocks



Containers

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

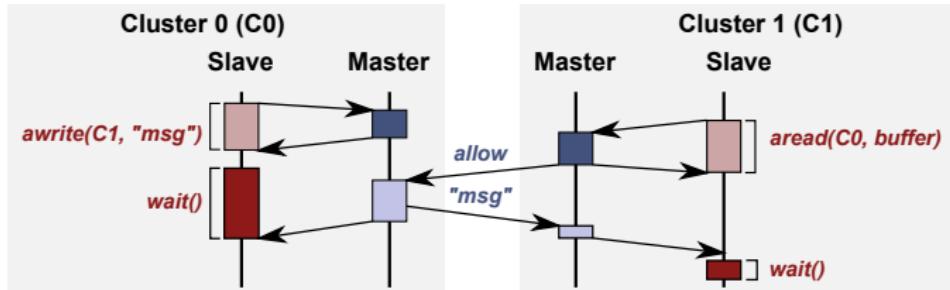
Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

- OS-level virtualization
- Multiple isolated user space instances (Shared OS)
- Lightweight compared to VMs
 - Linux Containers
 - Docker
- Example: Use *cgroups* and *chroot* to isolate a group of process and restrict their resource access.



Migration of Containers

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

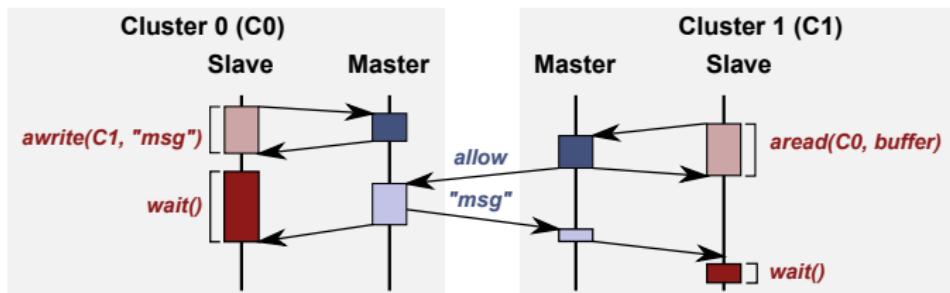
Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

- Transfer the file system of the container
- Save the state of the container into a file (all processes and their resources)
- Transfer the container file
- Restart the container



Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

Migration



Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

- Generally, there are **three stages of memory transfer:**
 - Push Copy (iteration copy) stage
 - Stop-and-Copy stage
 - Pull Copy (On-demand copy) stage
- Migration techniques
 - Stop-and-Copy
 - *Pre-Copy*
 - *Post-Copy*
 - Hybrid

Process Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

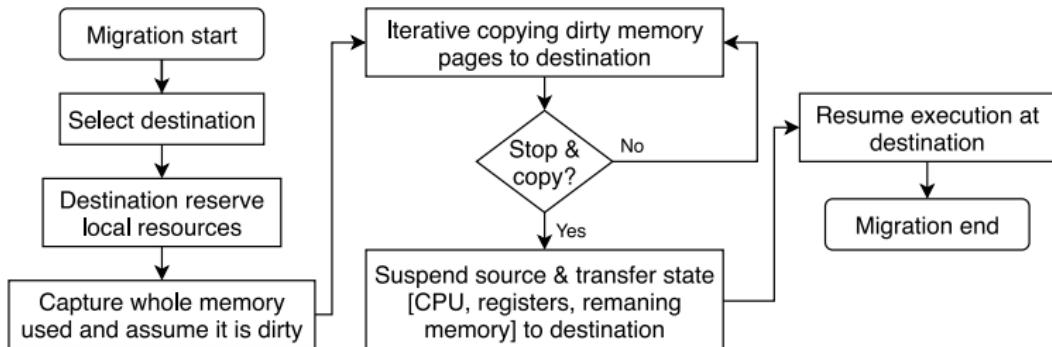
Kalray MPPA-256

Motivation And
Justify

- Generally, there are three stages of memory transfer:
 - Push Copy (iteration copy) stage
 - Stop-and-Copy stage
 - Pull Copy (On-demand copy) stage
- **Migration techniques:**
 - Stop-and-copy
 - *Pre-Copy*
 - *Post-Copy*
 - Hybrid

Pre-Copy Migration

Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Pre-Copy Scenario

Pre-Copy Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

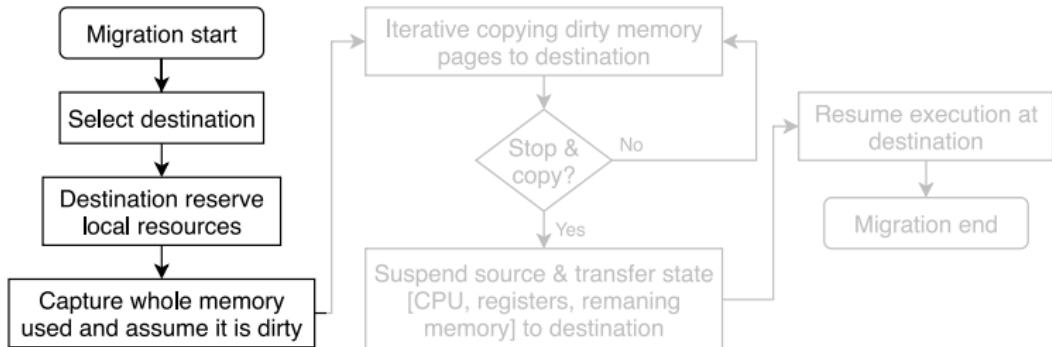
Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

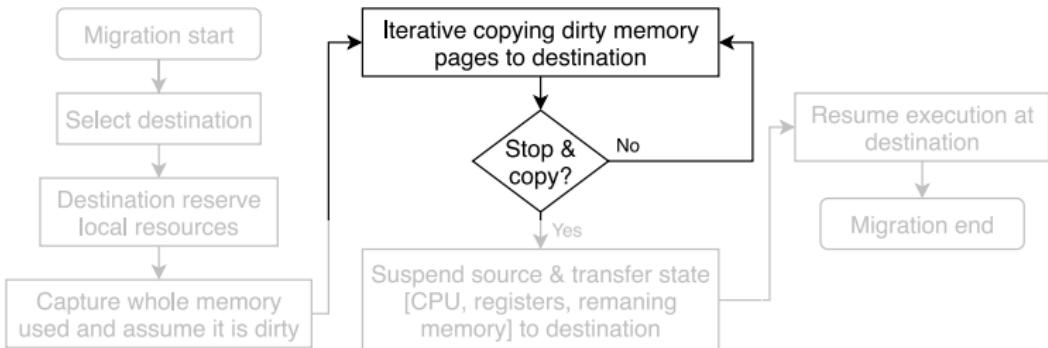
Motivation And Justify



Resource reservation and **transfer preparation**

Pre-Copy Migration

Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Sending the memory pages before the execution context

Push Copy Stage

Pre-Copy Migration

Introduction

Virtualization

Migration

Pre-Copy Migration

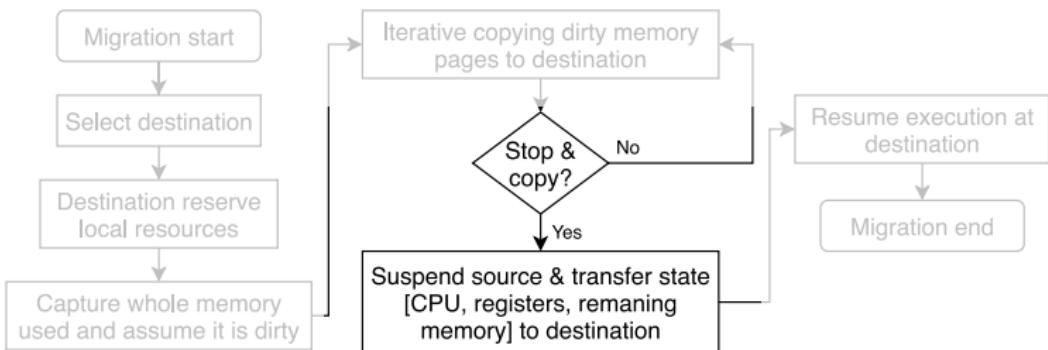
Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

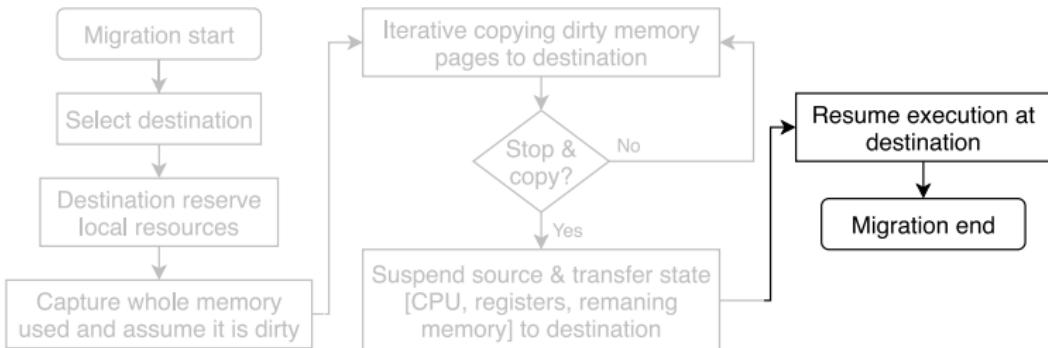
Motivation And Justify



When enough memory has been transferred, send the execution context
Stop-and-Copy Stage

Pre-Copy Migration

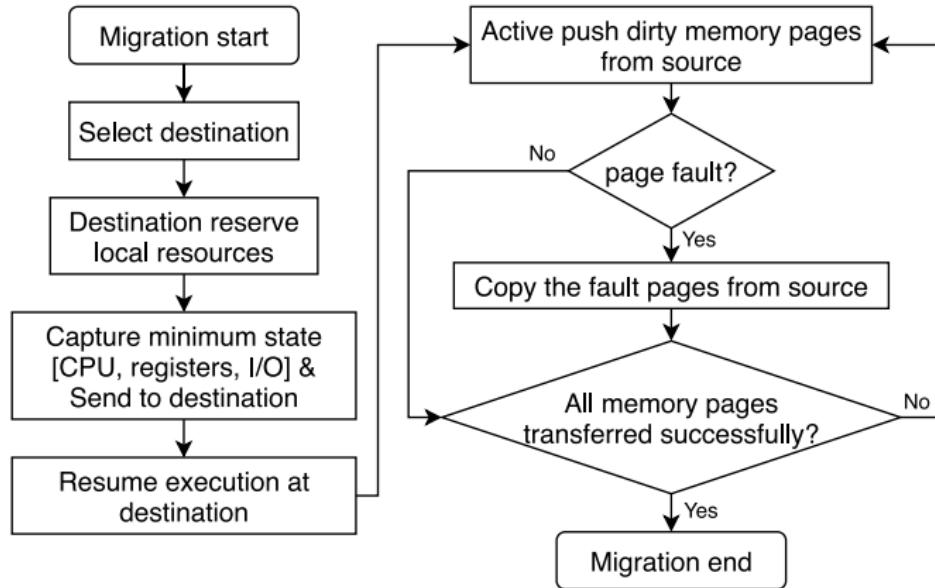
Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Resume stopped execution context in the destination

Post-Copy Migration

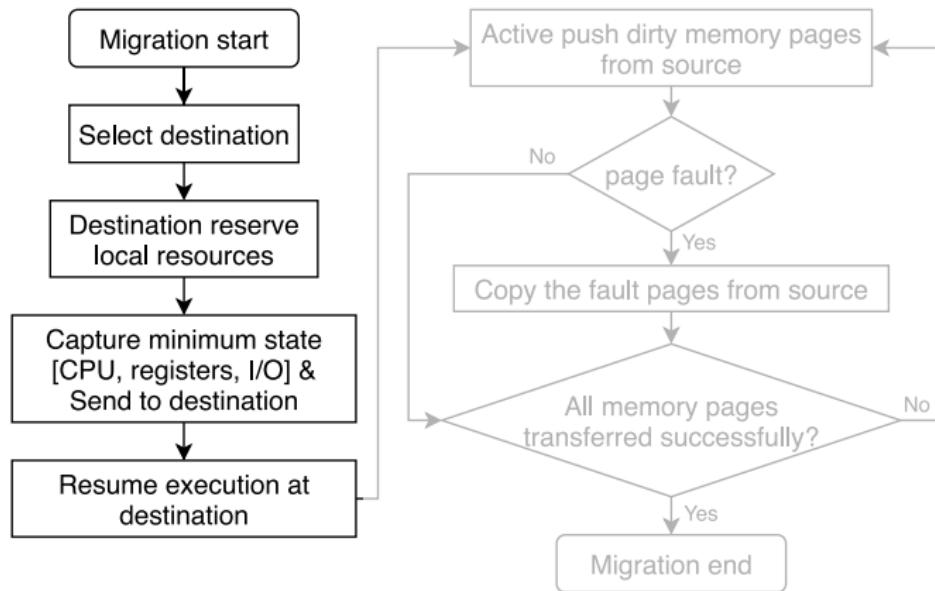
Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Post-Copy Scenario

Post-Copy Migration

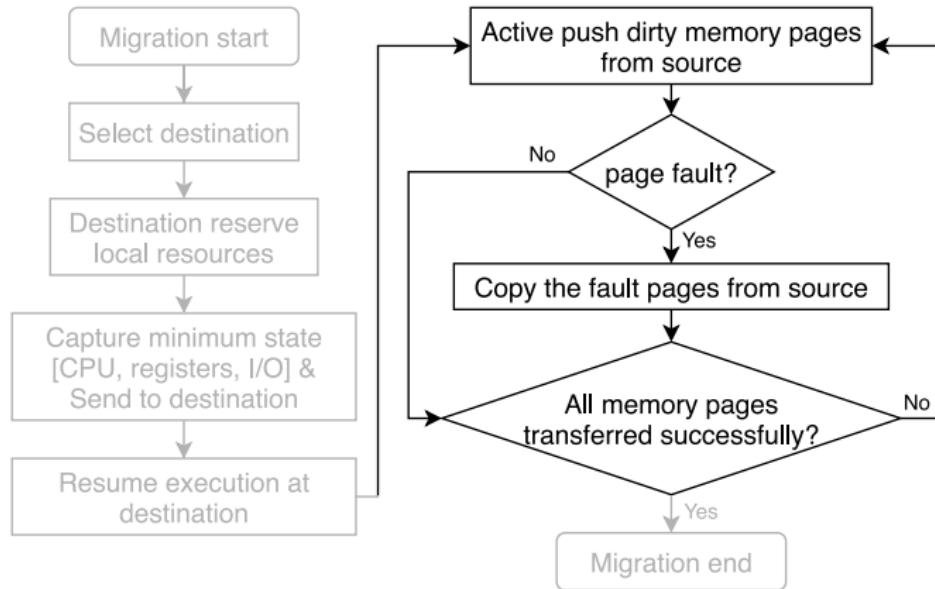
Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



**Sends execution context before memory pages
Stop-and-Copy Stage**

Post-Copy Migration

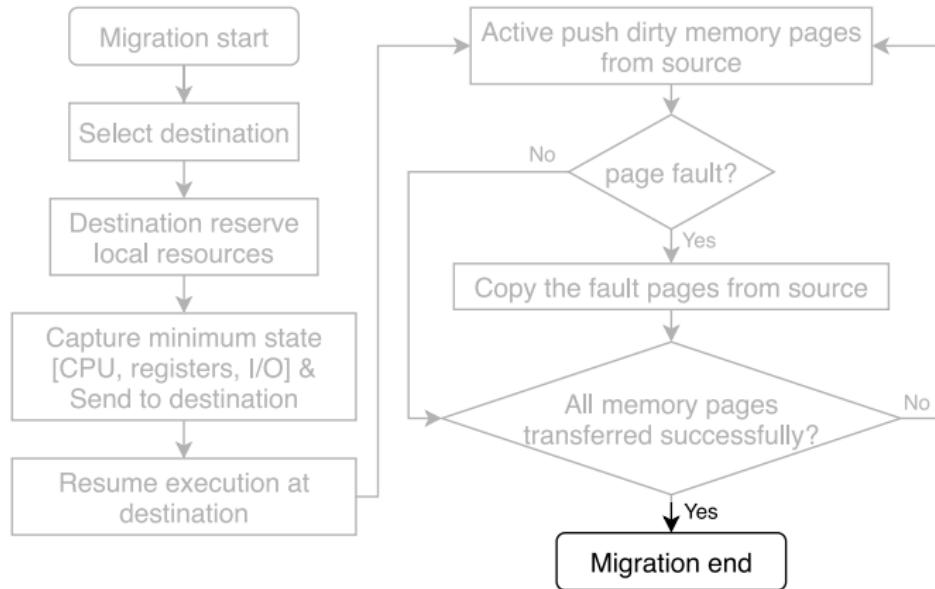
Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Merges **pull rest of memory pages** and any page that generate **page faults**
Pull Copy Stage

Post-Copy Migration

Introduction
Virtualization
Migration
Pre-Copy Migration
Post-Copy Migration
Metrics
Architectures
Kalray MPPA-256
Motivation And Justify



Continues to run normally

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

Metrics

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

Architectures



Lightweight Manycores Processors

Introduction

Virtualization

Migration

Pre-Copy Migration

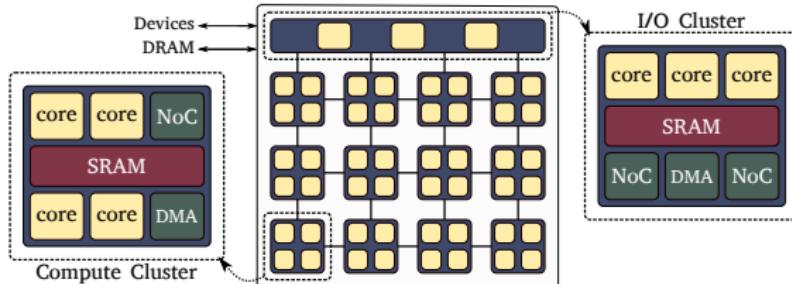
Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify



Overview of a Manycore

■ Hundreds of Lightweight Cores

- Expose Massive thread-level parallelism
- Feature low-power consumption
- Target MIMD workloads
- Distributed Memory Architecture
- On-Chip Heterogeneity

Lightweight Manycores Processors

Introduction

Virtualization

Migration

Pre-Copy Migration

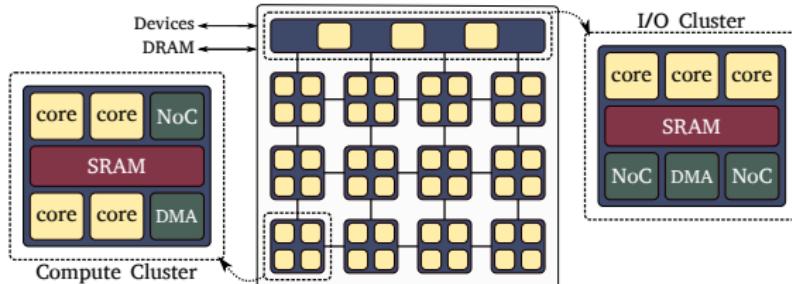
Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify



Overview of a Manycore

- Hundreds of Lightweight Cores
- **Distributed Memory Architecture**
 - Grants scalability
 - Relies on a Network-on-Chip (NoC)
 - Has constrained memory systems
- On-Chip Heterogeneity

Lightweight Manycores Processors

Introduction

Virtualization

Migration

Pre-Copy Migration

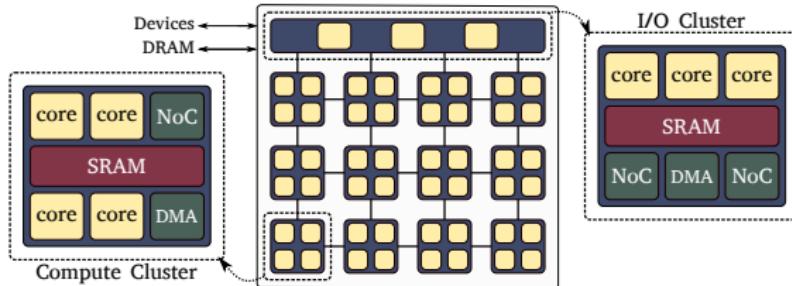
Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify



Overview of a Manycore

- Hundreds of Lightweight Cores
- Distributed Memory Architecture
- **On-Chip Heterogeneity**
 - Features different components

Kalray MPPA-256

A Lightweight Manycore Processor

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

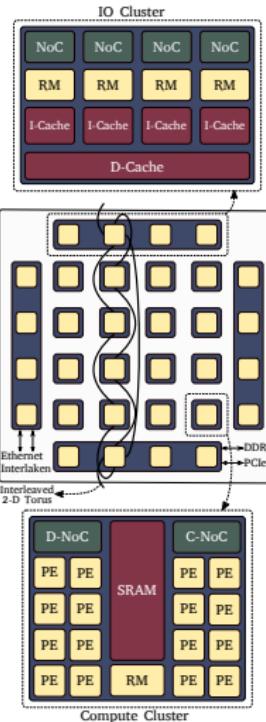
Kalray MPPA-256

Motivation And Justify

- **288 processing cores**
 - 16 Compute Cluster (CC)
 - 4 I/O Cluster (IO)

- **Data NoC (D-NoC)**
 - 256 RX slots
 - 8 TX channels
 - 8 μ threads for async TX

- **Control NoC (C-NoC)**
 - 128 RX slots
 - 4 TX channels



Motivation And Justify

Conclusions

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And Justify

■ Motivation

- Historical evolution from single-cores to Lightweight Manycores

■ Contribution

- A Inter-Cluster Communication Facility for LW Processors

■ Results

- **Optimal sizes** for large data transfers
- Well-known **distributed algorithms can be efficiently supported** by Nanvix OS

■ Future Works on Nanvix OS

- Remove limitation on asynchronous send
- MPI port (BSc dissertation)
- Shared Memory Service (MSc dissertation)
- **Distributed Process Scheduling (MSc dissertation)**



Thank you!
Questions?

João Vicente Souto

joao.vicente.souto@posgrad.ufsc.br

Parallel Computing - Computer Science
INE/UFSC, Florianópolis

September 28, 2020

References I

Introduction

Virtualization

Migration

Pre-Copy Migration

Post-Copy Migration

Metrics

Architectures

Kalray MPPA-256

Motivation And
Justify

