基于香山处理器的标签化体系结构设计与实现

蔡洛姗

中科院计算所

2022年8月25日

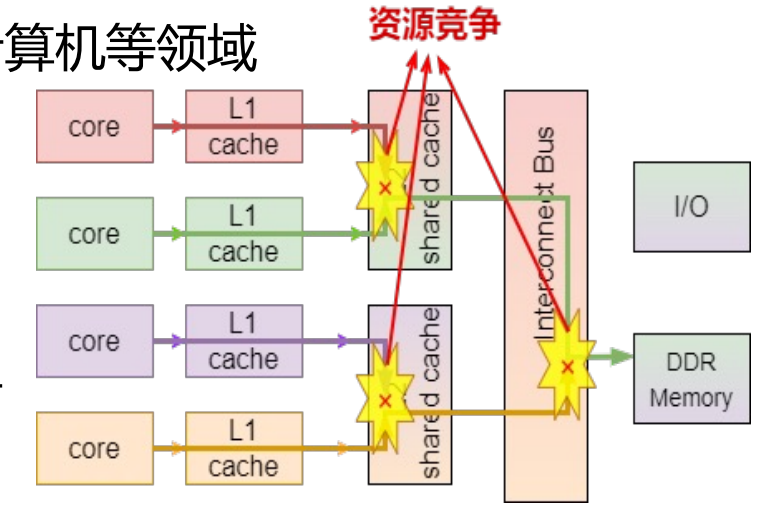
大纲

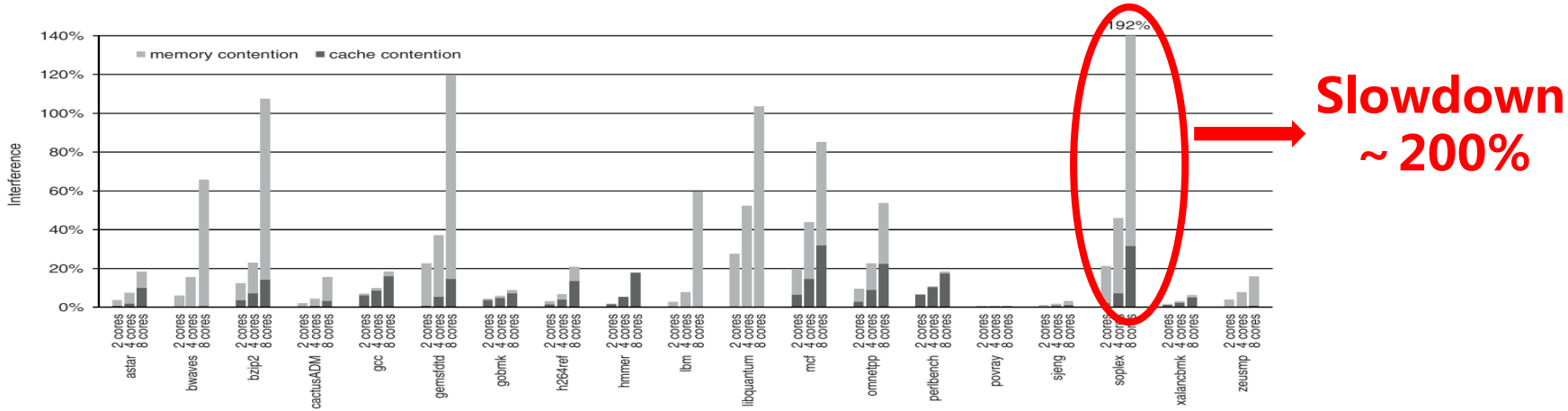
- 标签化冯·诺依曼体系结构
- 标签化香山整体架构
- 实现标签的传递
- 应用1: NoHype硬件虚拟化机制
- 应用2: 缓存容量划分机制
- 实验结果
- 总结与展望

研究背景 · 多核资源竞争与性能波动

- 多核处理器正广泛应用于个人移动设备、PC、服务器、高性能计算机等领域
  - ➢ 大幅提高系统的吞吐量
- 多核架构带来的挑战：

  引起对共享硬件资源（Cache、Bus、Memory 、IO …）的竞争
  - ➢ 导致性能下降，访问延迟增加，资源利用率和服务质量降低





Slowdown ~ 200%

研究背景 · 如何减少无序竞争带来的干扰

- 优先级控制+共享资源划分：Linux cgroup， DMHA [CF'09], 、 MCP[MICRO'11]
- 但传统计算机采用**分层模式**，底层硬件**无法区分**数据属性，上层软件**信息无法传递**到底层硬件，软件与硬件之间**协同优化难**

关键问题：

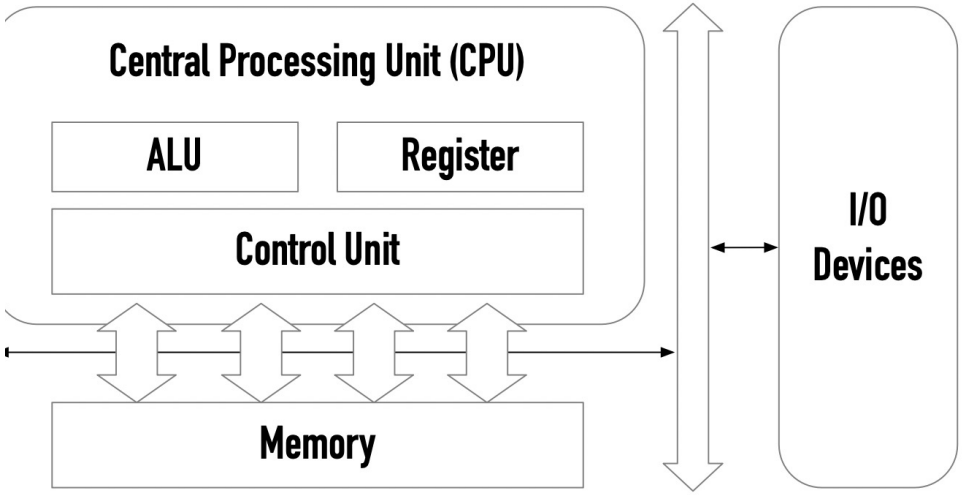
硬件精准获取上层软件的需求信息，并据此实现资源隔离与调度；

但传统体系结构缺少接口传递类似信息

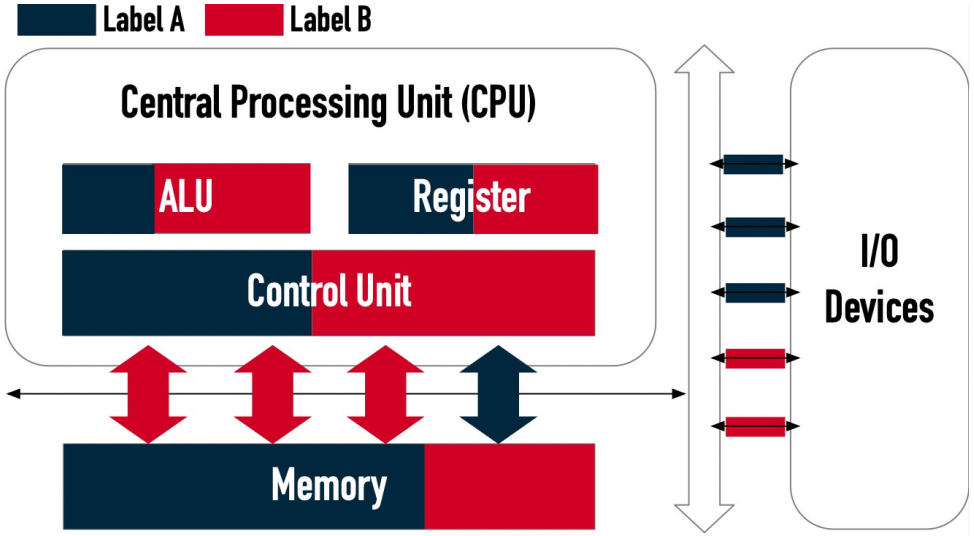
标签化冯·诺依曼体系结构

设计思路：

计算机中的每个组件（处理器核心，缓存，内存，IO设备等）之间的关系可以看作一个网络，组件之间的请求传递就相当于网络包通信。通过在经典冯·诺依曼结构基于地址访问数据的基础上，增加一套标签机制，打通软硬件全通路的信息传递，提高体系结构的控制能力，降低计算机系统内的竞争干扰。



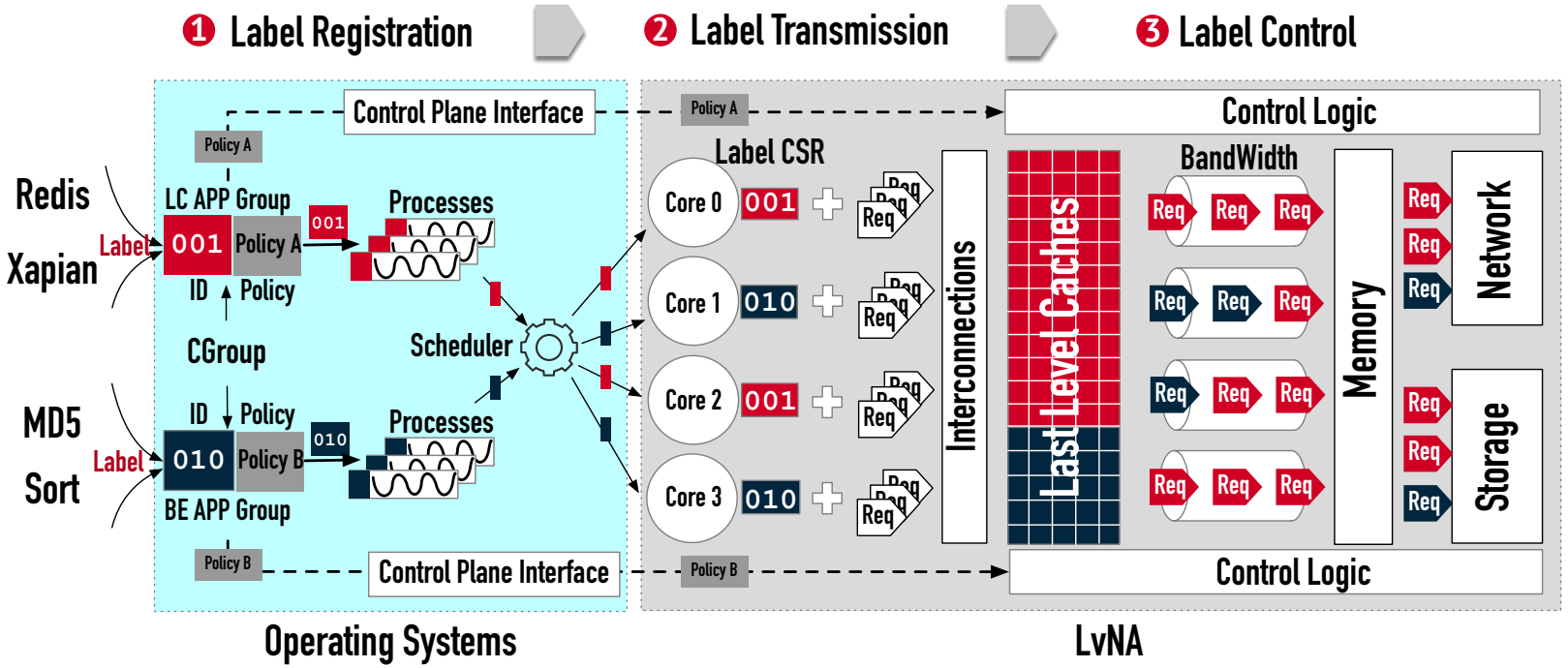
The von Neumann Architecture



The Labeled von Neumann Architecture

标签化体系结构

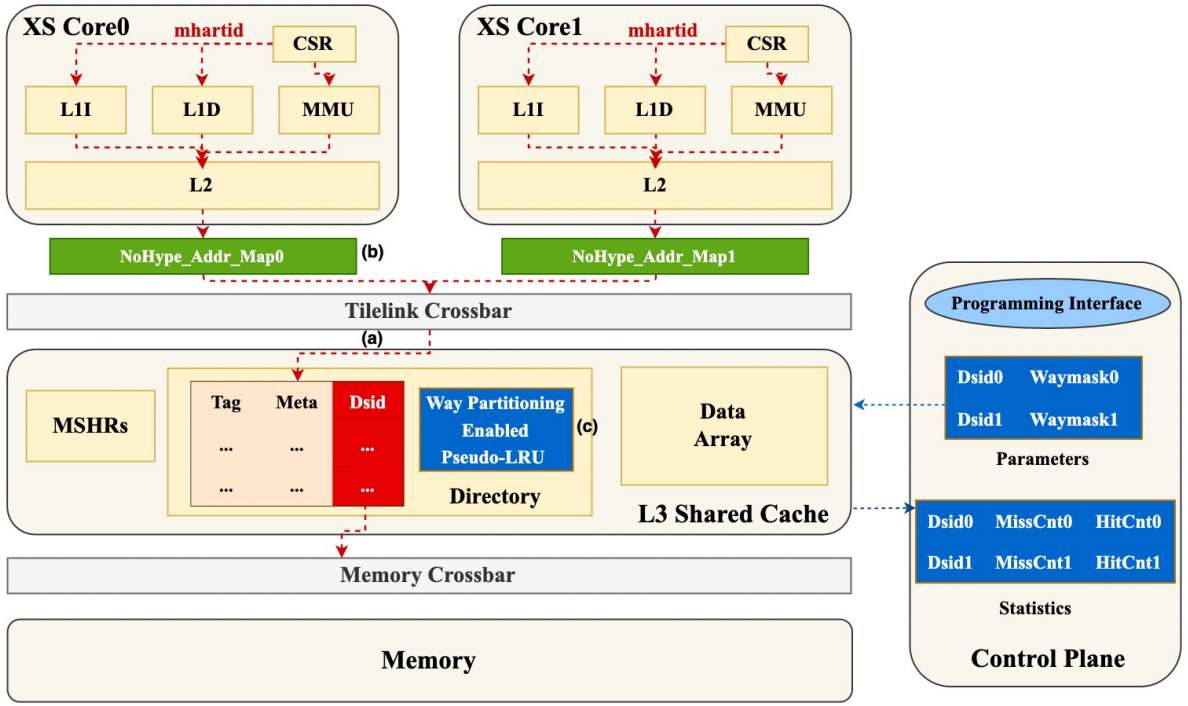
- **控制对象：**为每一个计算机的内部请求增加一个标签
- **关联语义：**标签值与上层高级实体（如虚拟机、线程、软件变量等）的关联
- **携带传播：**标签将在请求访问各个存储层次过程中全程携带
- **软件定义标签的控制逻辑：**软件创建基于标签的规则，硬件根据标签含义对请求差异处理



标签化香山整体架构

开发目标：通过在香山上引入标签机制，实现基于标签的资源隔离与划分，使得标签化体系结构在高性能开源处理器上得到进一步的应用，吸引更多开发者在香山上探索标签化的应用场景

开源地址：https://github.com/OpenXiangShan/XiangShan/tree/labeled-xs



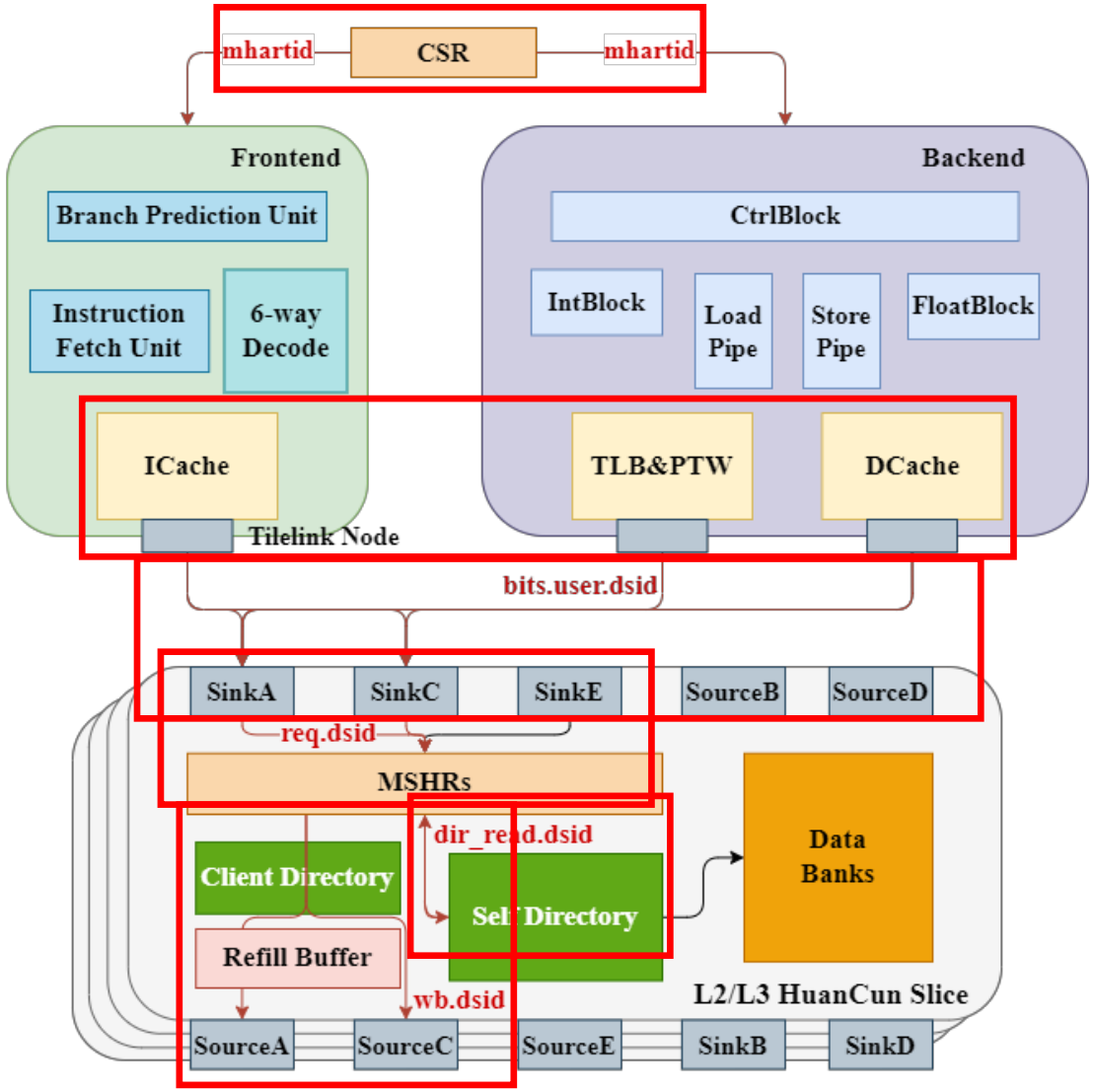
① 标签传递机制

② 基于标签的 NoHype 机制

③基于标签的可编程缓存容量划分机制

标签的传递


核心标签CSR.mhartid
ICache
DCache
PTW
Tilelink总线自定义bit
L2/L3 HuanCun内部请求
SelfDir目录读请求
SelfDir目录meta数组
MSHR
Refill, Writeback
Tilelink


mhartid
CSR
mhartid
Frontend
Branch Prediction Unit
Instruction Fetch Unit
6-way Decode
ICache
Backend
CtrlBlock
IntBlock
Load Pipe
Store Pipe
FloatBlock
TLB&PTW
DCache
Tilelink Node
bits.user.dsid
SinkA
SinkC
SinkE
SourceB
SourceD
req.dsid
MSHRs
dir_read.dsid
Client Directory
Self Directory
Refill Buffer
wb.dsid
Data Banks
L2/L3 HuanCun Slice
SourceA
SourceC
SourceE
SinkB
SinkD

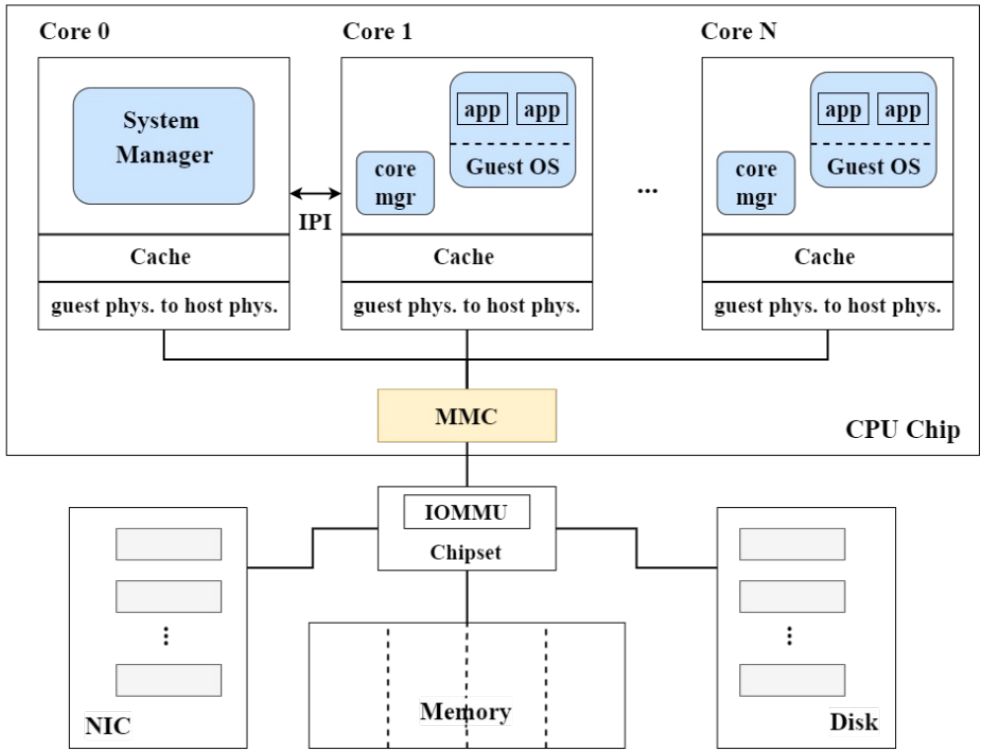
应用1: NoHype硬件虚拟化机制

NoHype 概念:

在虚拟化平台中，将Hypervisor完全移除，同时由硬件负责实现虚拟化层的管理功能（对 CPU、内存和 IO 设备的仲裁访问，网口转发、管理虚拟机的启动关闭等）

作用：

- **物理隔离硬件资源，防御安全攻击**
- **消除Hypervisor层开销**
- **为多系统的实验场景提供支持**



NoHype原始架构图

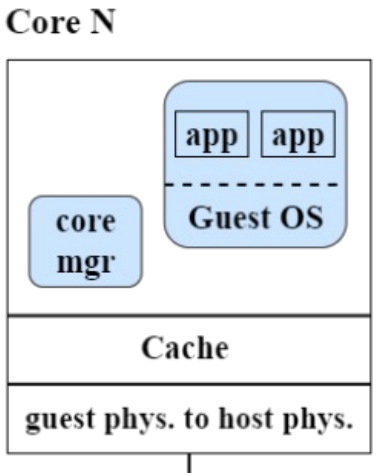
应用1: 基于标签的NoHype硬件虚拟化机制

实现方案：

- 每个处理器核心专用于一个虚拟机
- 内存分区：
  - OS被分配的物理内存对应到真实主机中是内存空间的一部分
  - 处理器负责将虚拟机（核心）物理内存地址映射到主机物理内存地址

    在处理器内部（私有L2 与共享 L3 之间）添加重映射模块，

    将每个核心的物理访存空间基于核心标签进行线性地分配
- I/O设备分配：为核心指定专用的I/O设备
  - 设定虚拟机启动操作系统时需要一个串口，基于核心标签分配对应串口的地址空间
  - 在操作系统初始化阶段指定不同虚拟机/核心发往中断控制器的中断地址



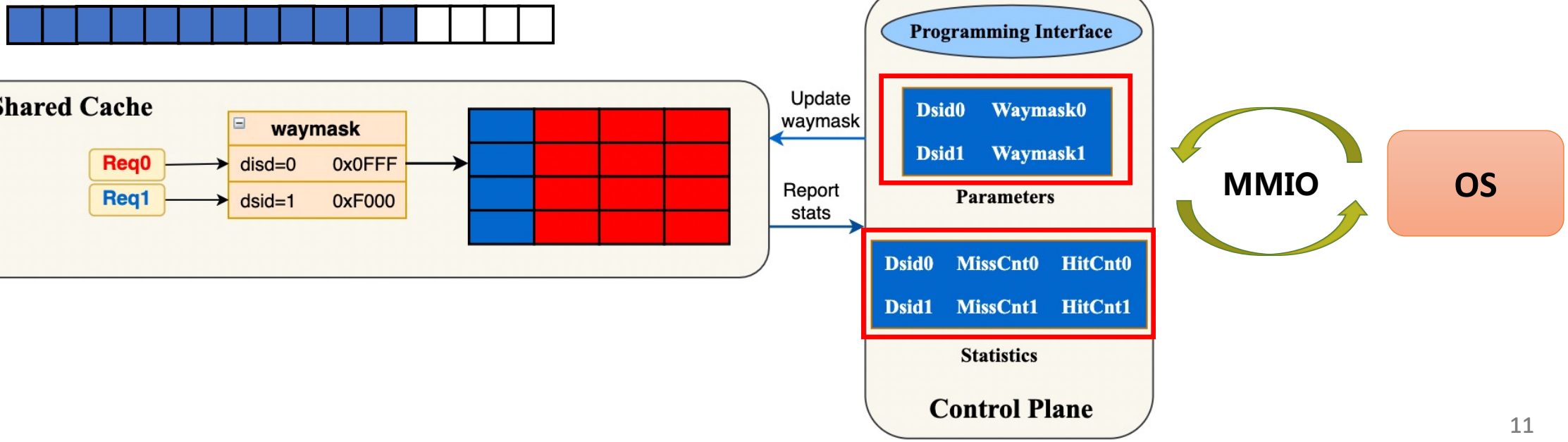
应用2: 基于标签的缓存容量划分机制

- 共享LLC的控制平面与操作系统交互
- 变量 waymask 表示划分方式，作用于路替换算法：

  L3从控制平面读取请求dsid对应的waymask，在waymask(i)=1对应的路中选择替换路

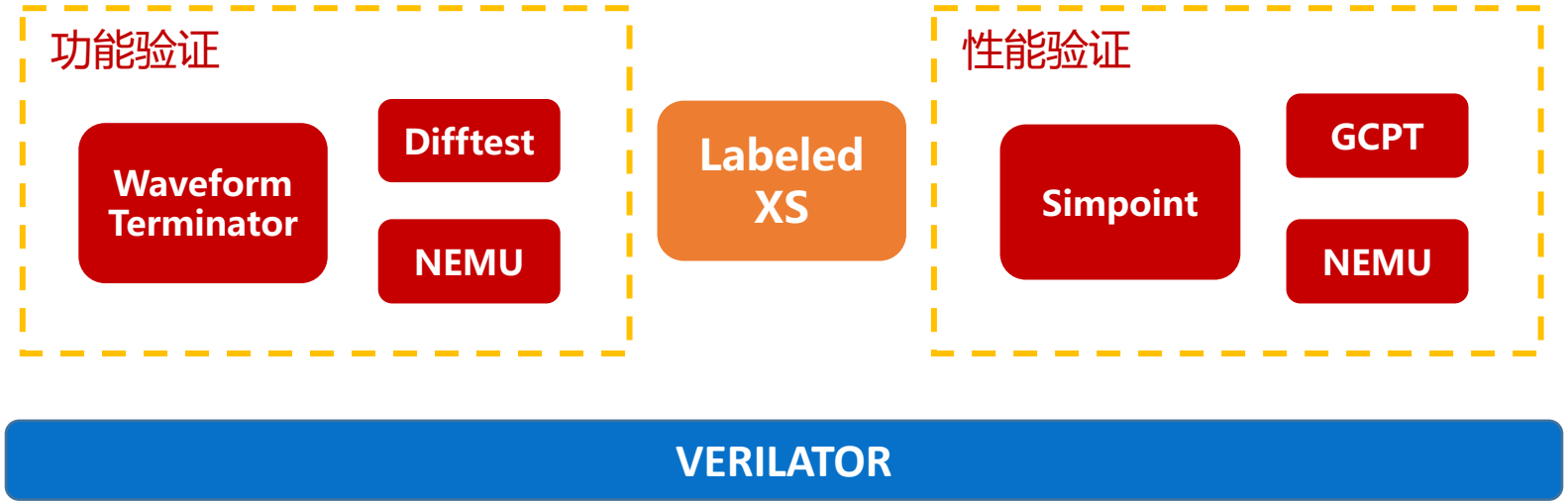
➢ 被替换的缓存块被当前核心占有，保证了该核心占有的缓存空间在waymask分配的容量内

例：waymask=FFF0



实验结果 · 环境配置

- 仿真实验平台



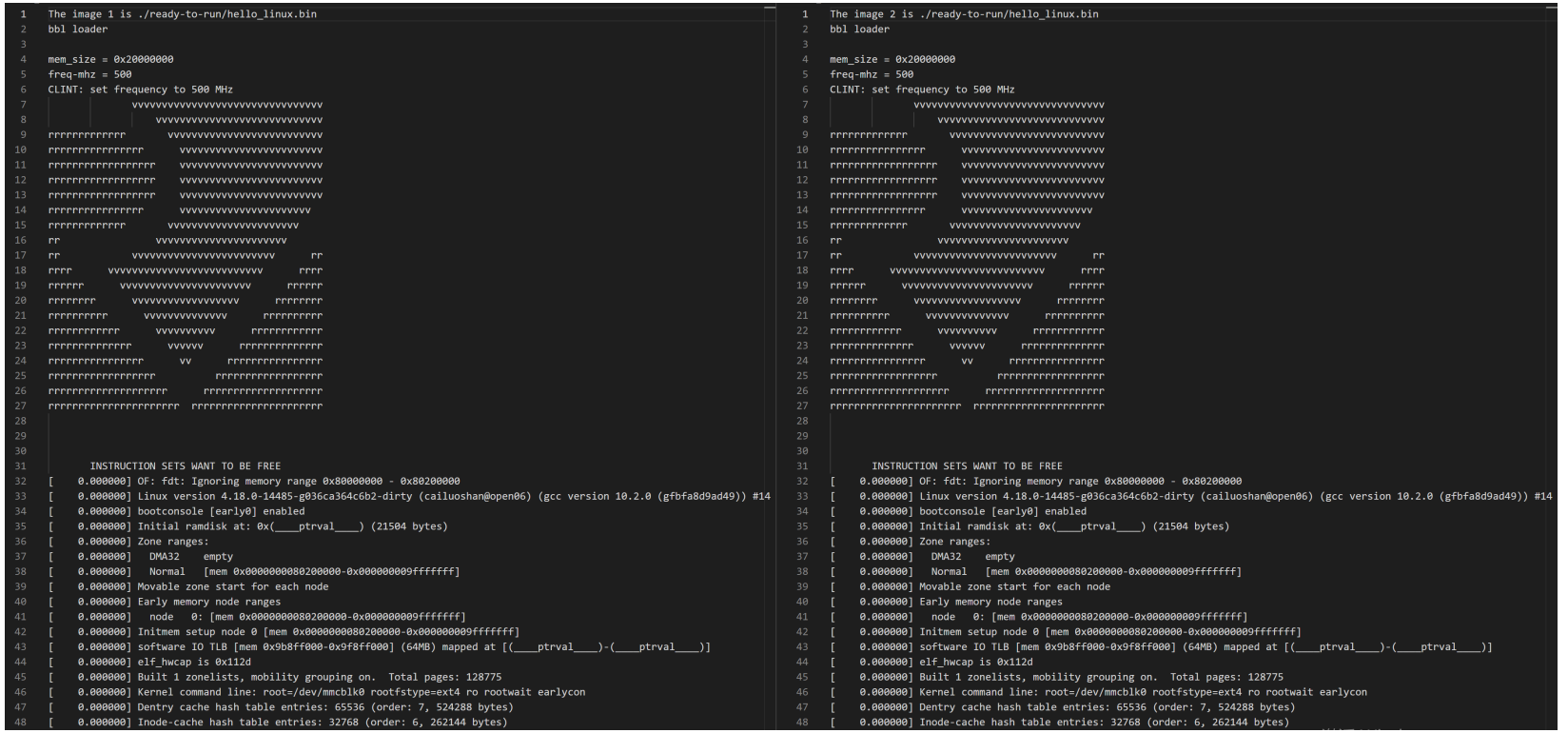
- 评估内容

① 在 Nohype 机制下每个香山核心是否可以独立启动Linux操作系统

② 共享缓存路划分机制是否能减少系统干扰，保障目标应用的性能

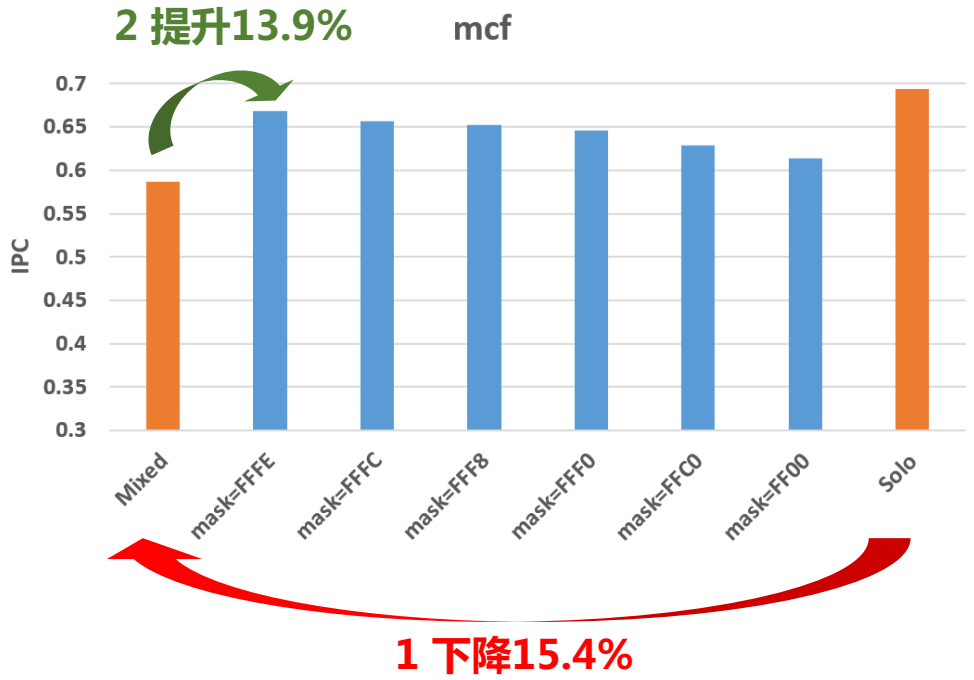
实验结果 · 硬件虚拟化

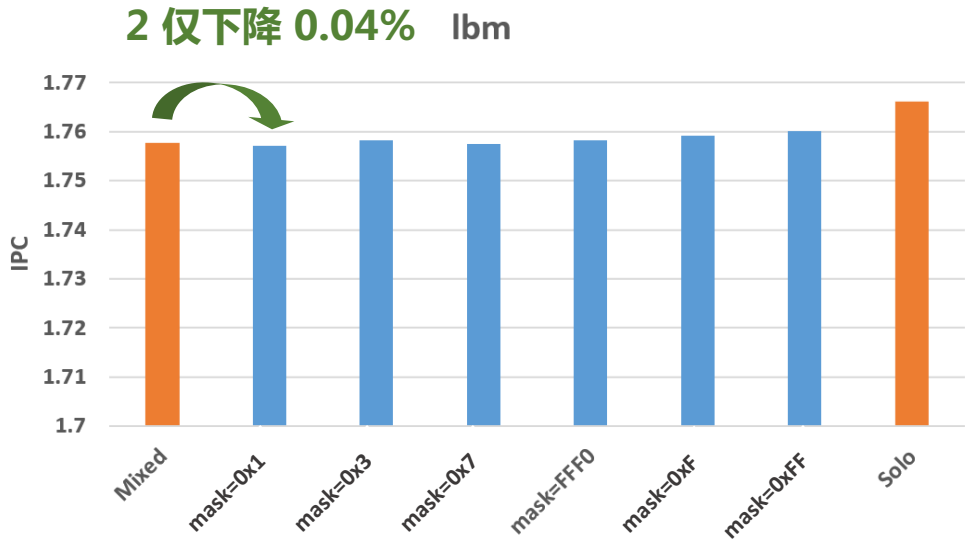
- 在双核处理器上启动两个 Linux 操作系统



实验结果 · 关键应用性能保障

- 测试程序：SPEC mcf (cache sensitive) 关键应用、lbm (memory intensive) 干扰应用
- 60,000,000 InstCnt；运行时固定 waymask





总结与展望

总结

基于香山高性能开源处理器，设计并实现了一个标签化体系结构的处理器原型，为上层软件提供与硬件交互的接口；并在此基础上实现了无需软件虚拟化层的NoHype机制和可编程缓存容量划分机制。

展望

- 目前只进行了概念性的验证，后续需要在FPGA开发板上进一步测试并评估硬件开销；
- 建立更系统的标签化香山功能验证与性能测试流程，供更多开发者参考和使用；
- 扩展标签的内涵，使香山支持进程级别的差异化处理，并可以传递除了实体标识之外的其它类型信息。

谢谢！

敬请批评指正！

蔡洛姗

2022年8月25日