UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

HiPeRT Lab
High-Performance Real-Time Lab

Setup of a multi-OS platform based on the Xen hypervisor

An industral case study

Paolo Burgio
paolo.burgio@unimore.it

Roberto Cavicchioli
Ignacio Sanudo Olmedo
Marco Solieri

Who are we?

High-Performance Real-Time Lab

High-Performance Real-Time Lab

- ✓ Since 2014 @UniMoRe
- ✓ 4 profs, 7(+1) post-docs, 4(+2) PhDs, 1 admin + undergrads
- ✓ http://hipert.unimore.it

HERCULES

OPEN-NEXT

IMECH

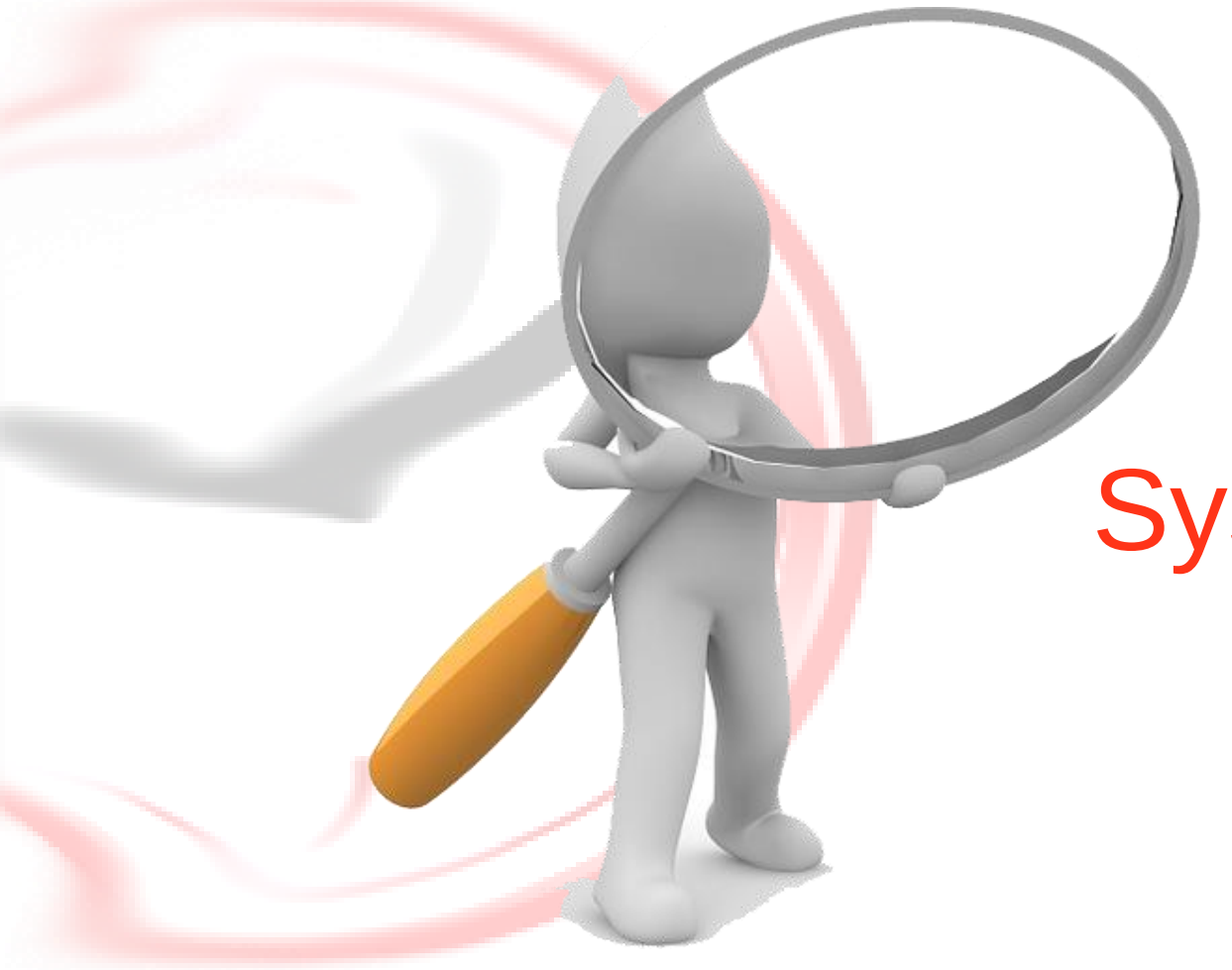
ENABLE-S3

OMISSIS

What we will see today

An OPEN-NEXT use-case

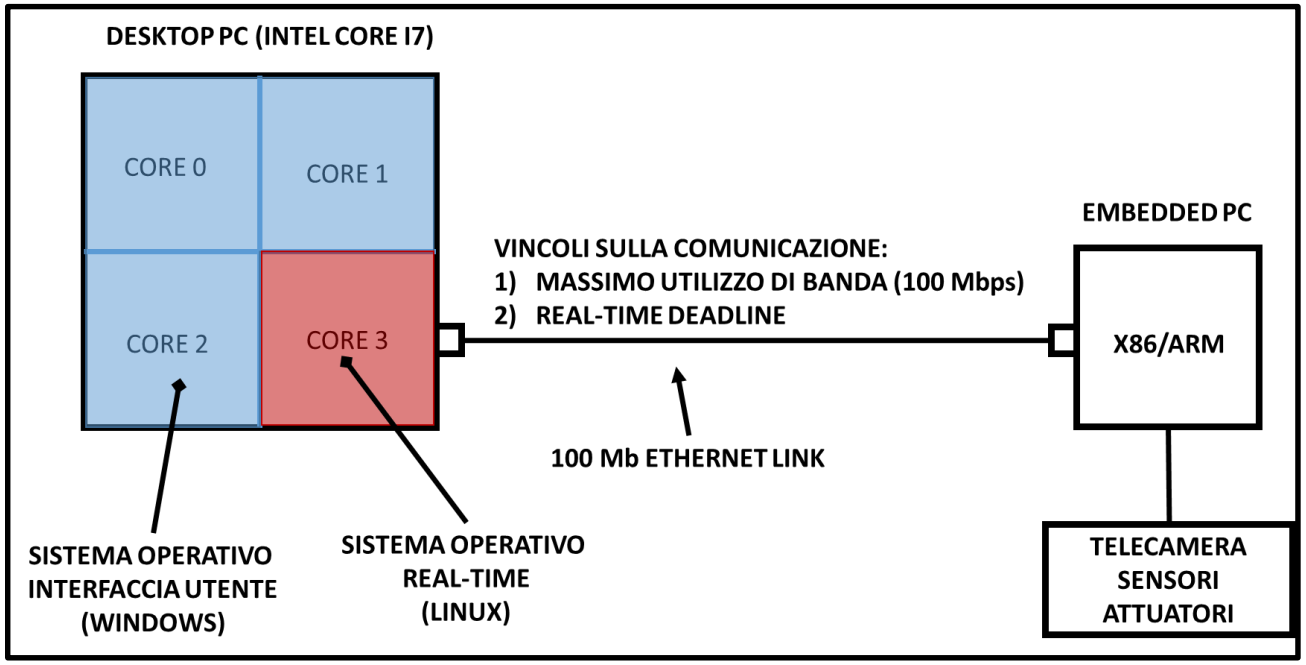
- ✓ System and problem description
- ✓ Current system benchmarking / profiling
- ✓ Proposed solution & benchmarking
  - Synthetic workloads
  - Real application



System description

The case study

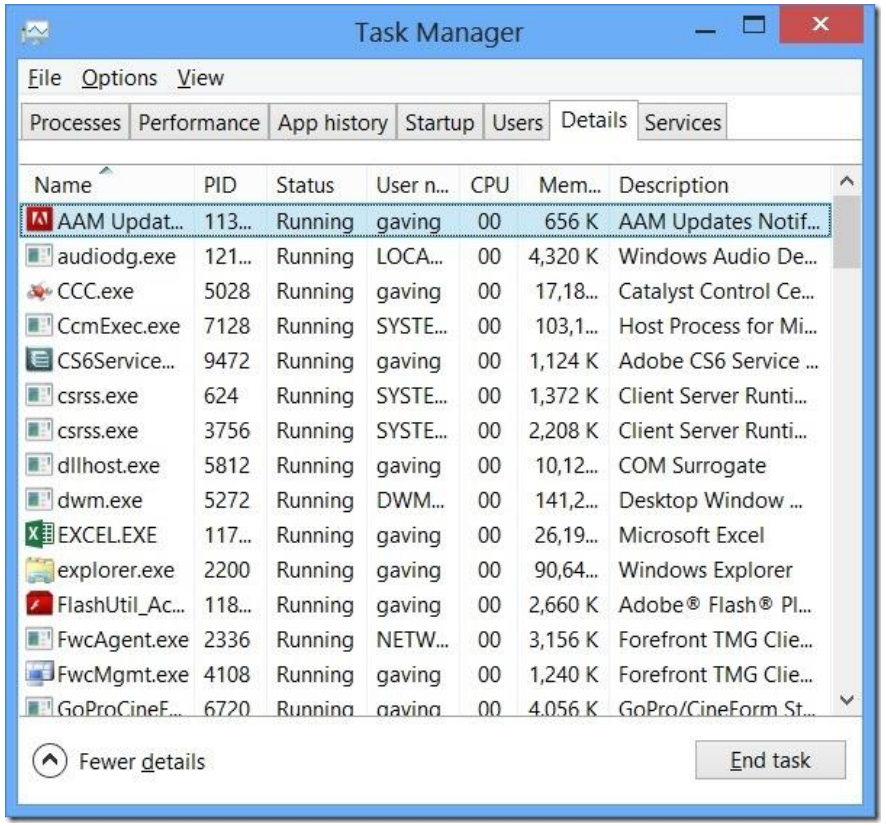
- ✓ Industrial automation system for packaging
- ✓ x86 host pc with Windows for HMI + ARM embedded PC for image processing in prod. line
- ✓ Goal: meet maximum comm. BW (100Mbps) under RT constraints, @TRL7



Background

What is a real-time computing system?

- ✓ Reacts within precise time constraints to events in the environment
- ✓ Deterministic behavior
- ~~✓ Windows determinism~~



Background

Real-time classification

- ✓ Hard: missing a deadline is a total system failure.
- ✓ Firm: infrequent deadline misses are tolerable, but may degrade the system's quality of service. The usefulness of a result is zero after its deadline.
- ✓ Soft: the usefulness of a result degrades after its deadline, thereby degrading the system's quality of service.

This case study

The target machine

- ✓ CPU:
  - 4 core Intel(R) Core(TM) i7-4770T CPU @ 2.50GHz
  - Windows 7
  - Soon, multi-OS (Win + Linux)
- ✓ Memory:
  - 32GB
- ✓ Network:
  - x8 I350 Gigabit Network Connection
  - 82574L Gigabit Network Connection
- ✓ Storage:
  - [0:0:0:0] disk ATA Samsung SSD 850 2B6Q /dev/sda (128GB)
  - [1:0:0:0] disk ATA Samsung SSD 850 2B6Q /dev/sdb (128GB)
  - [2:0:0:0] disk ATA ST2000DM001-1ER1 CC25 /dev/sdc (1,8T)

©2017 University of Modena and Reggio Emilia

System profiling

Synthetic benchmarks

CPU

- ✓ Setup
  1. High priority task that calculates a prime number
  2. Two low priority tasks that calculate π

Network

- ✓ Setup
  1. High priority IPERF3 task (Network computation)
  2. CPU prime95 interference (CPU computation)

Current (production) system

- ✓ Mono-OS (Win)

Benchmark suites

PRIME95

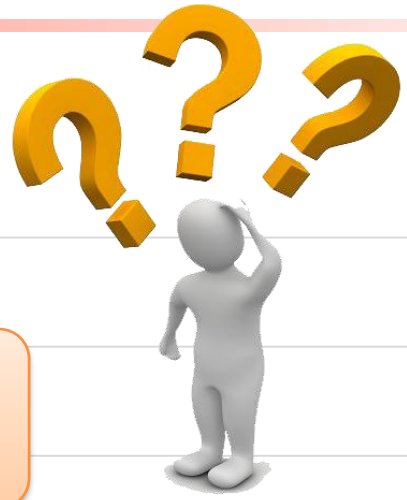
- ✓ Computes prime number(s)
- ✓ FFT

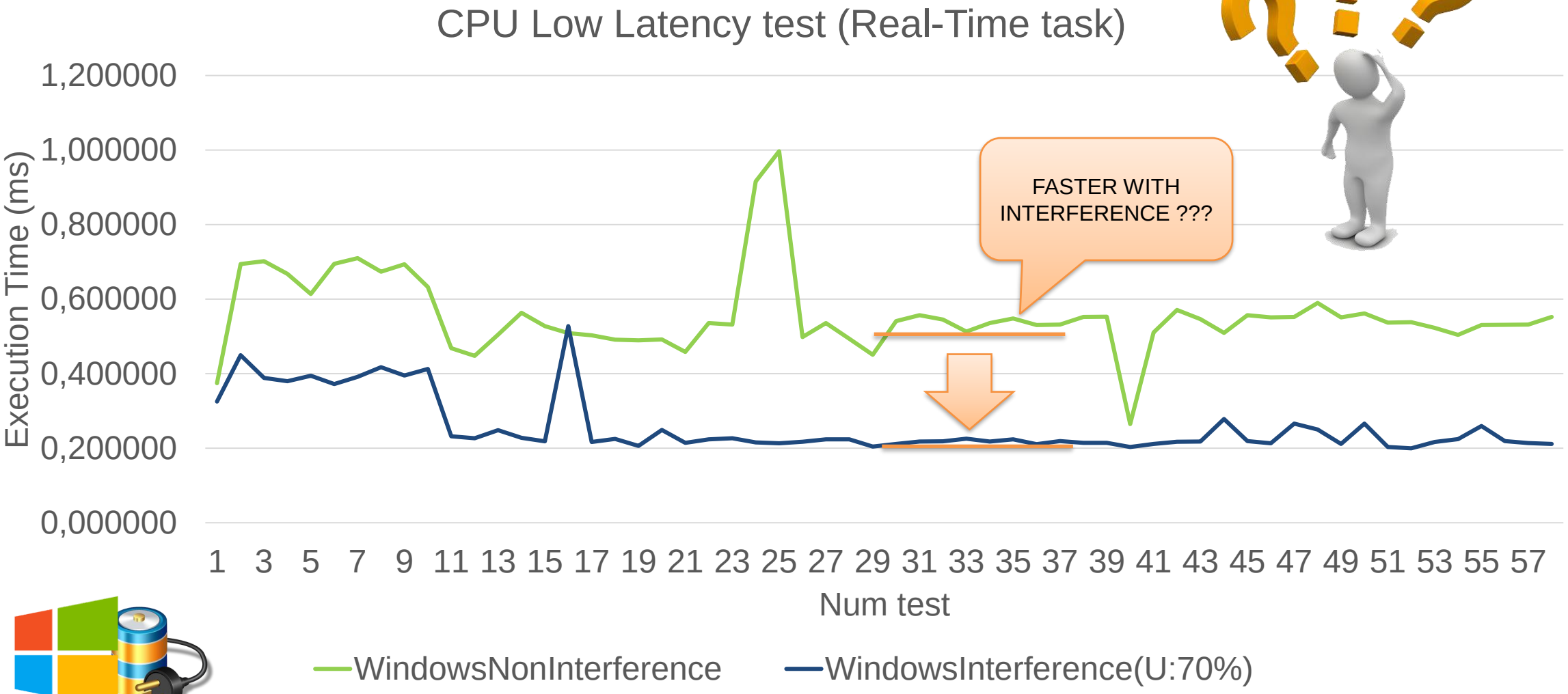
iPERF3

- ✓ Tool for active measurements of the maximum achievable bandwidth on IP networks
  - It supports tuning of various parameters related to timing, buffers and protocols (TCP, UDP, SCTP with IPv4 and IPv6)
  - For each test it reports the bandwidth, loss, and other parameters
  - Three-clause BSD license

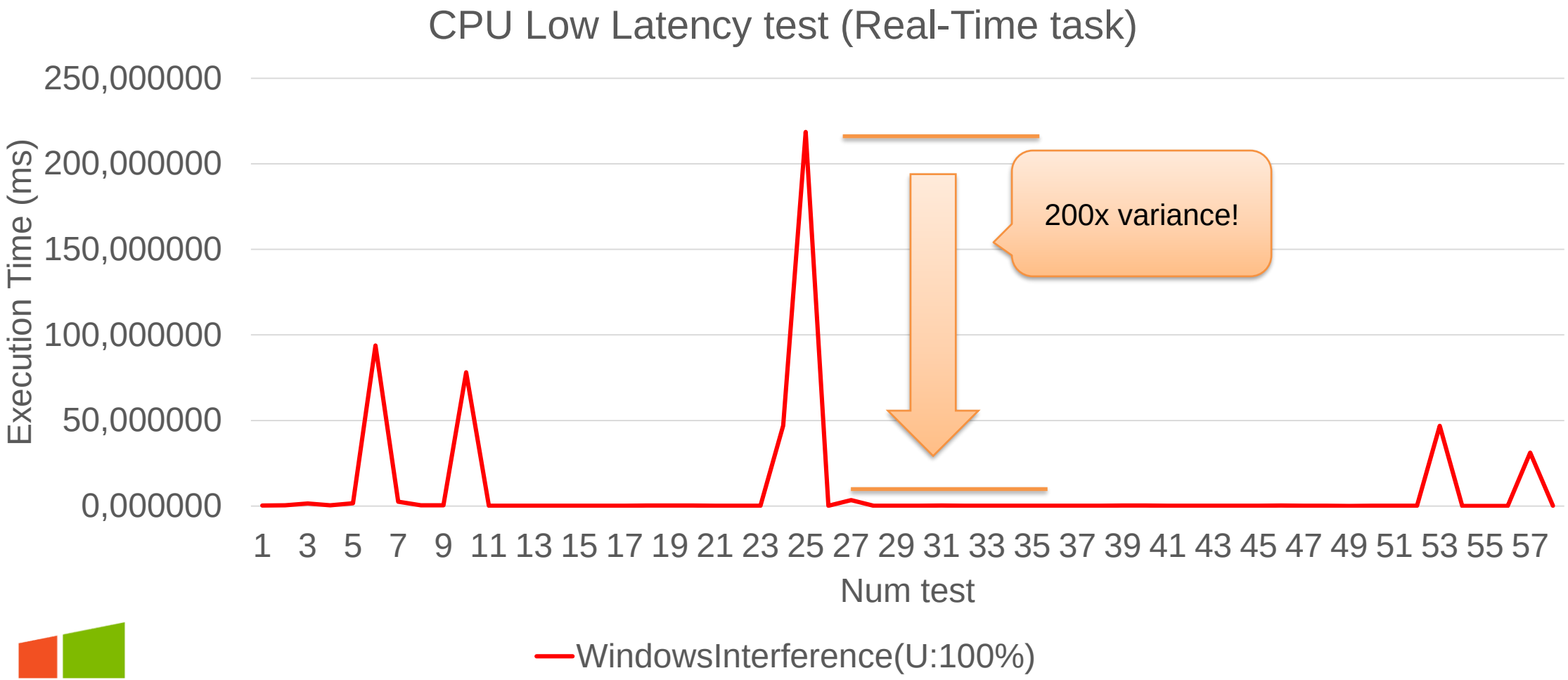
©2017 University of Modena and Reggio Emilia

CPU - A problem: OS power mgmt

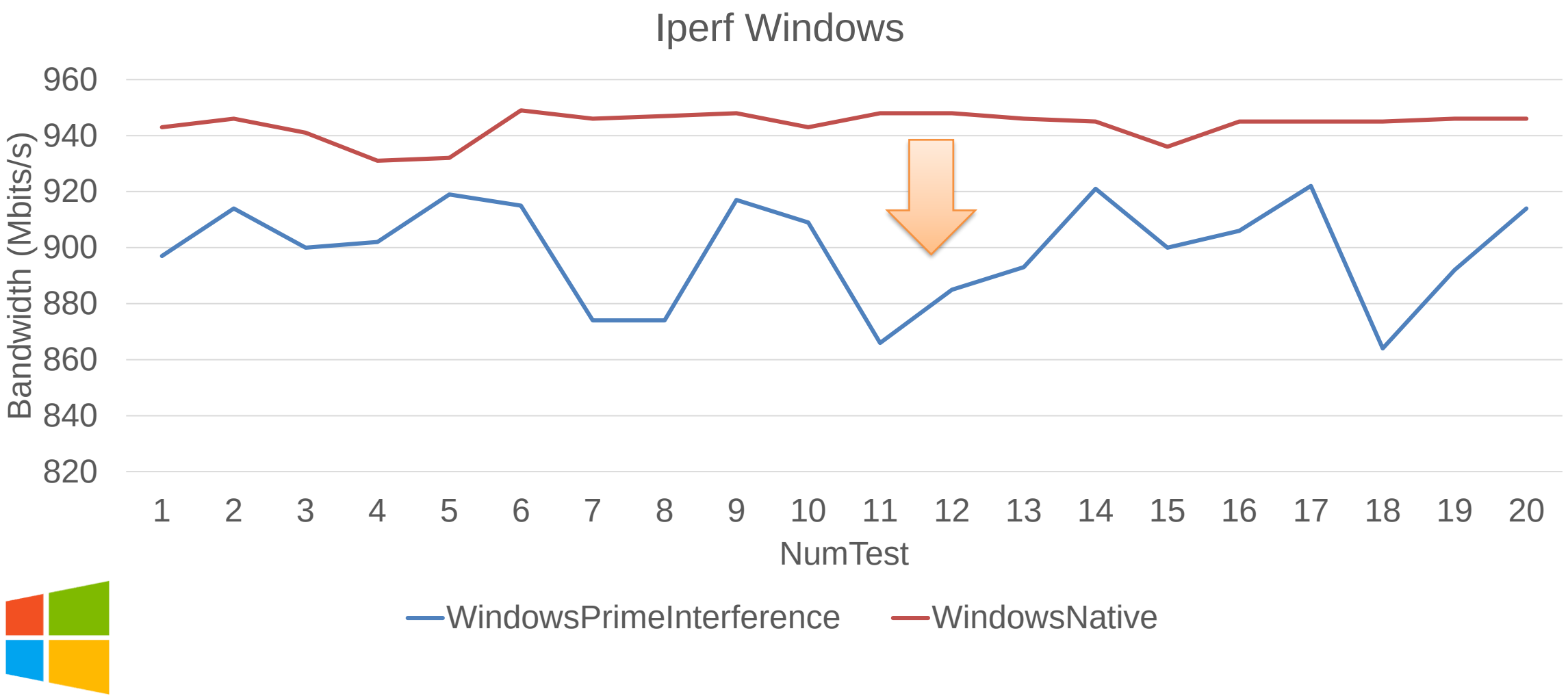




CPU – Exec time unpredictability



Network – sensitiveness to CPU interference



©2017 University of Modena and Reggio Emilia

February 20th, 2017 - Report #1

Main issues reported

- ✓ Problem with the power management
- ✓ We can not guarantee real-time constraints using Windows
- ✓ CPU computation messes up the (independent) network computation!

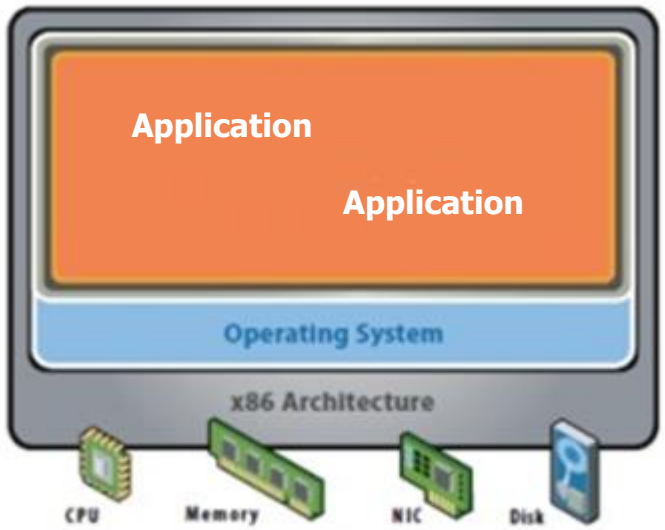
We must guarantee component isolation

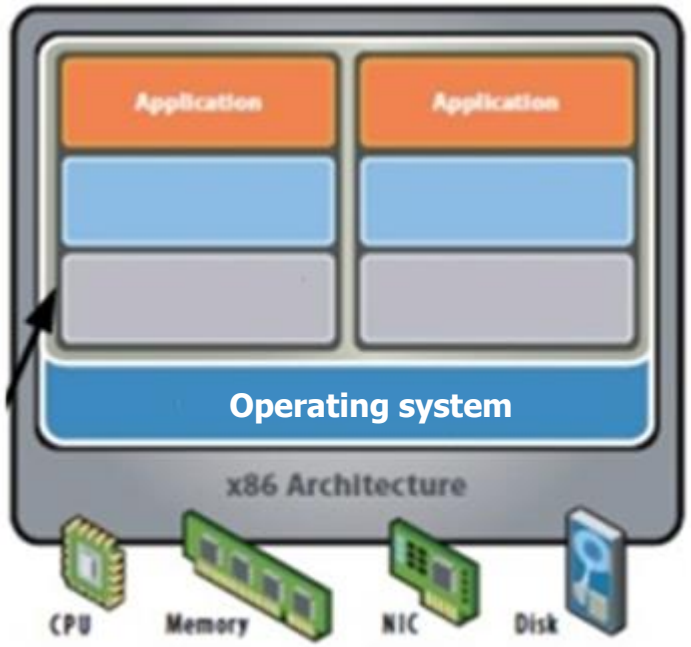
- ✓ Spatial
- ✓ Timing

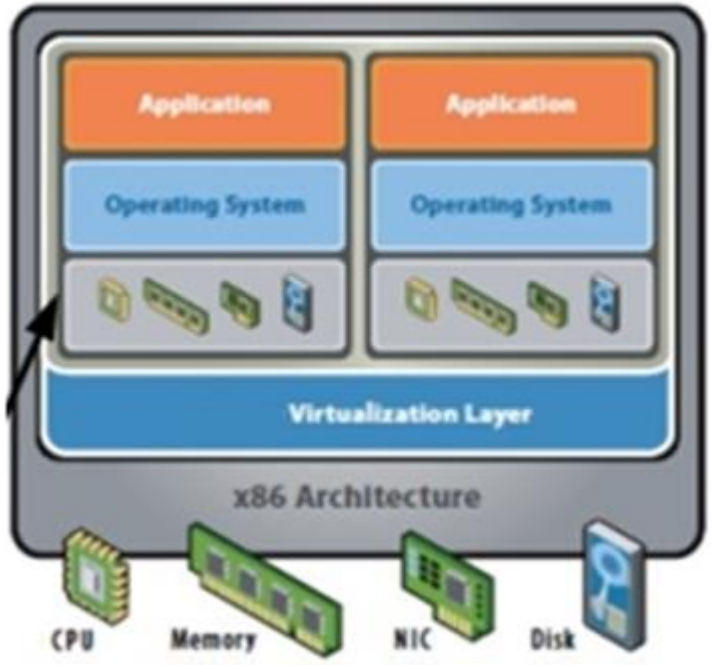
Proposed solution

…virtualization!

Multi-application: traditional approaches





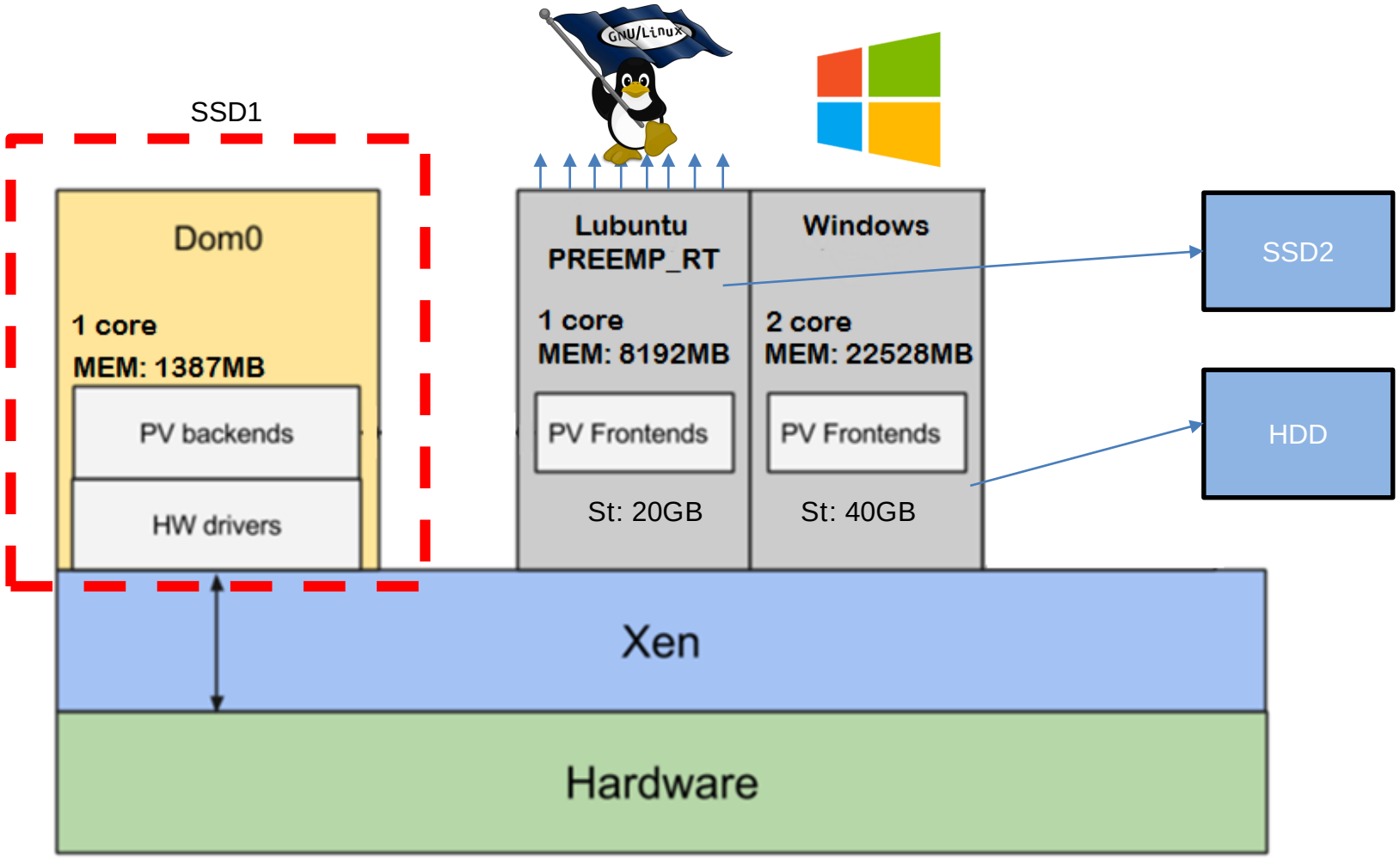


Performance...?

Determinism

Proposed system configuration

- ✓ Xen Hypervisor
- ✓ New platform generation
  - Mixed Linux/Win



Measure the hypervisor overhead

Goal: show the (non-)latencies introduced by the hypervisor:

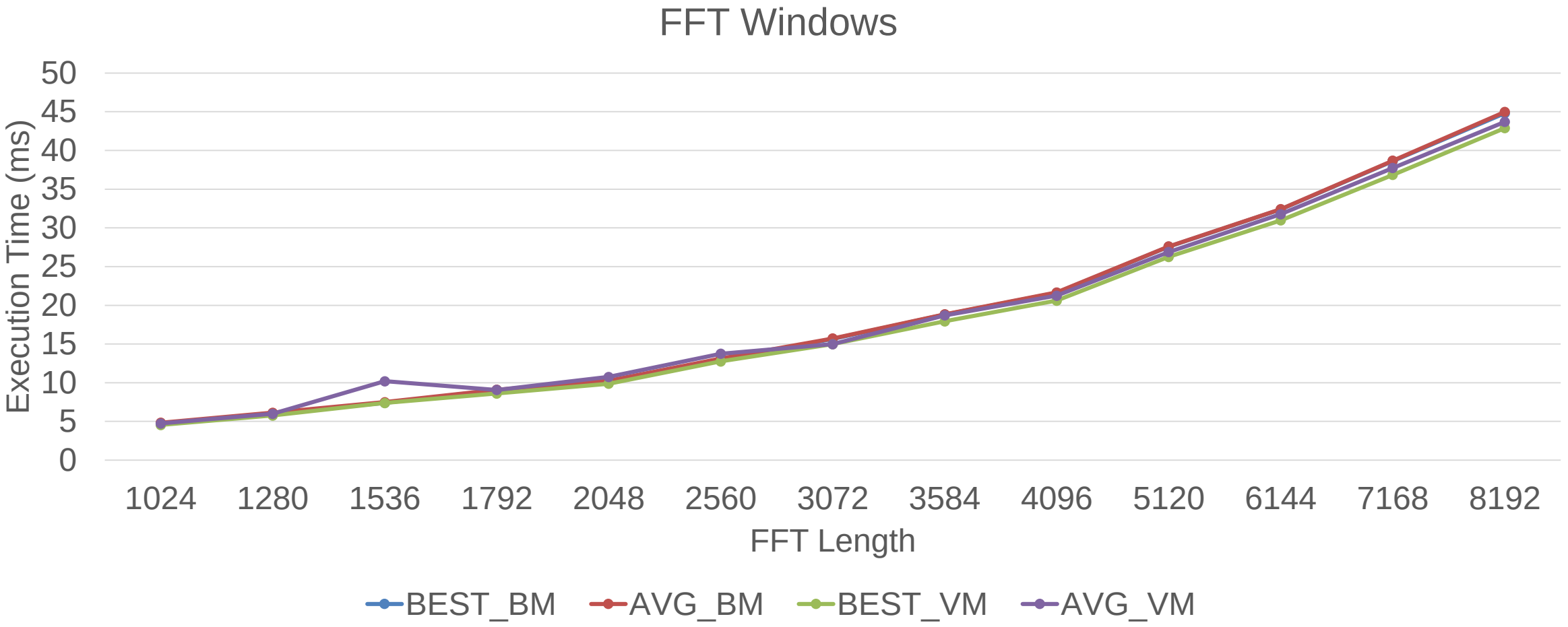
- ✓ Windows Virtual Machine (VM) vs Windows Bare Metal (BM)
- ✓ Linux VM vs Linux BM
- ✓ Benchmark single-OS vs multi-OS interference

Two tests

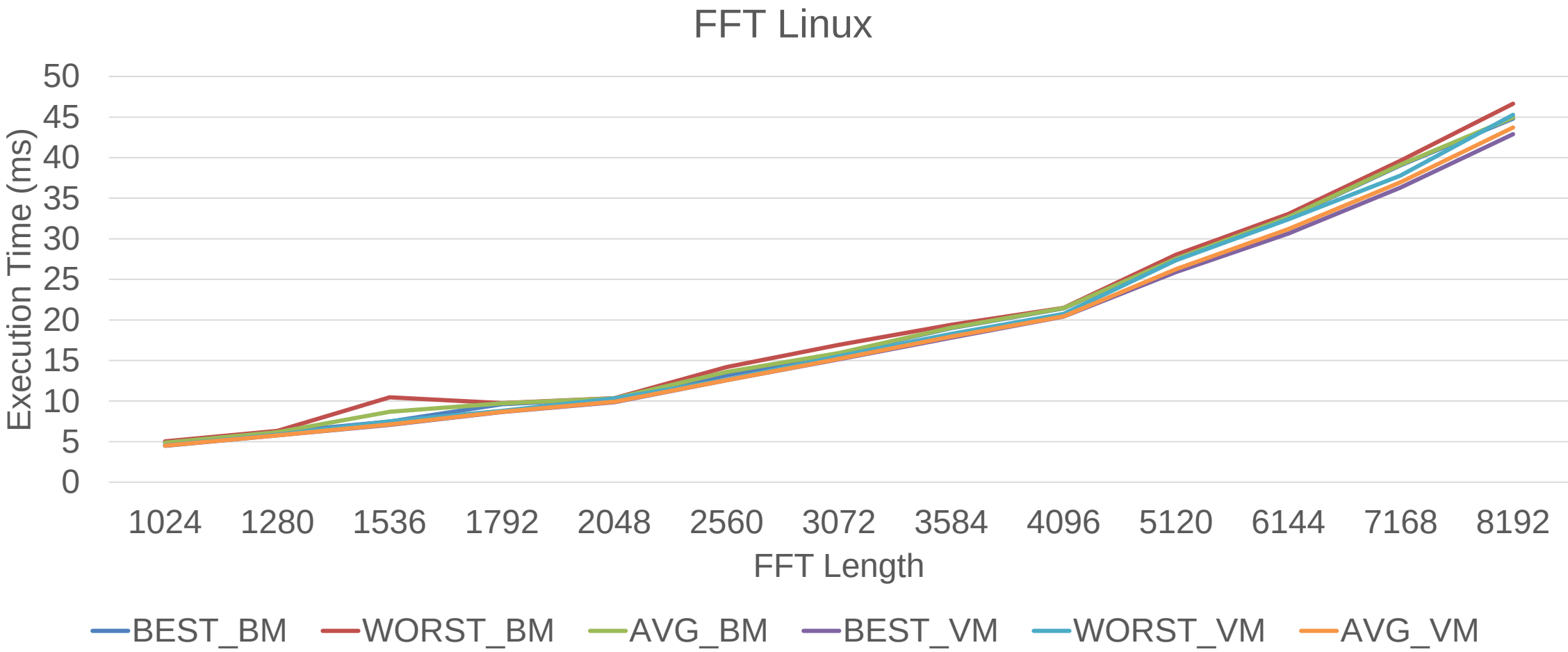
- ✓ CPU - intensive (FFT)
- ✓ Memory - intensive (R/W)
  - http://alasir.com/software/ramspeed/

No interference (single-OS)

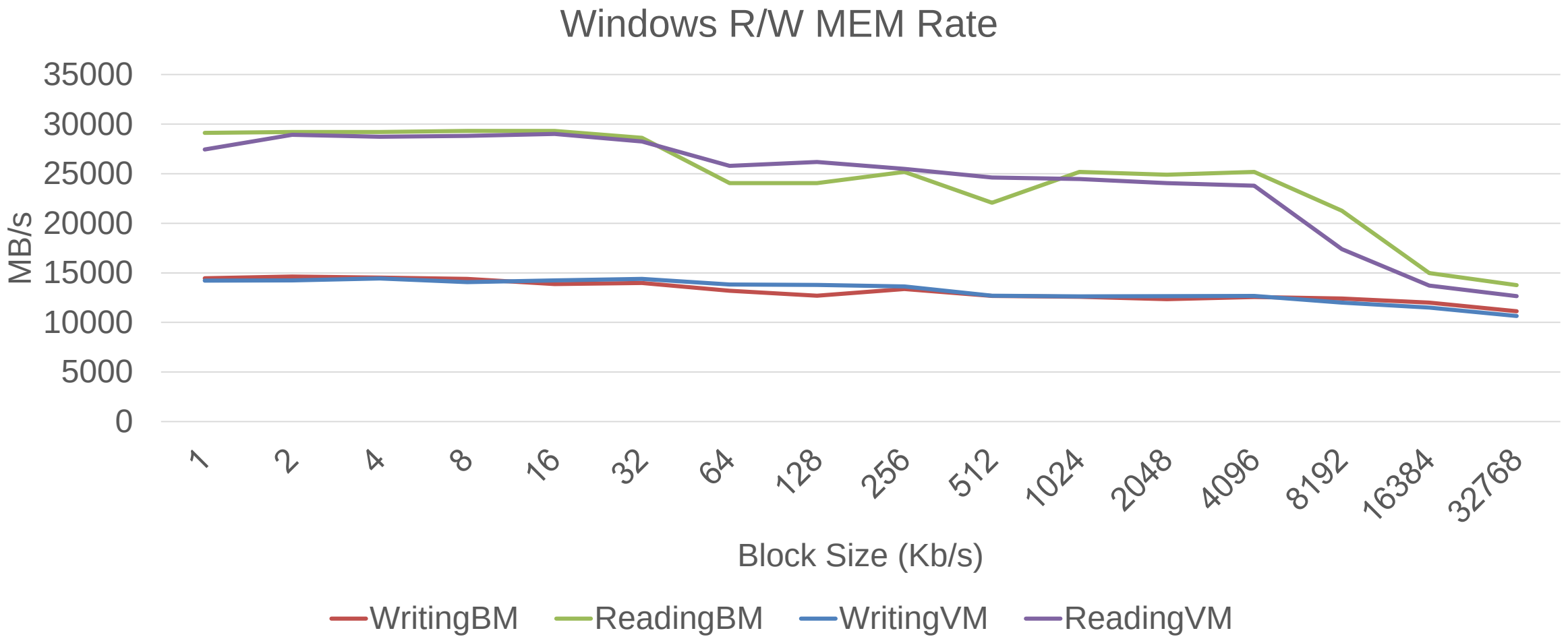
CPU - Xen overhead vs. BM (Win)



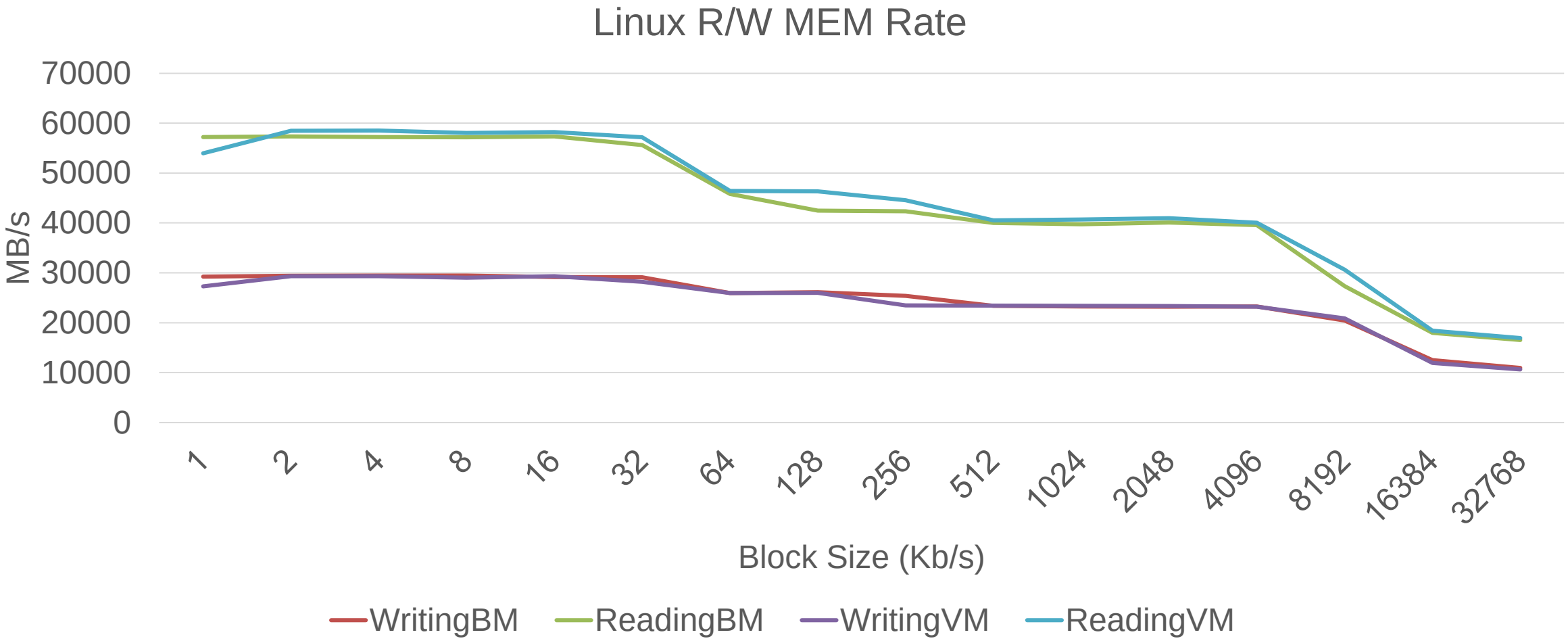
CPU - Xen overhead vs. BM (Linux)



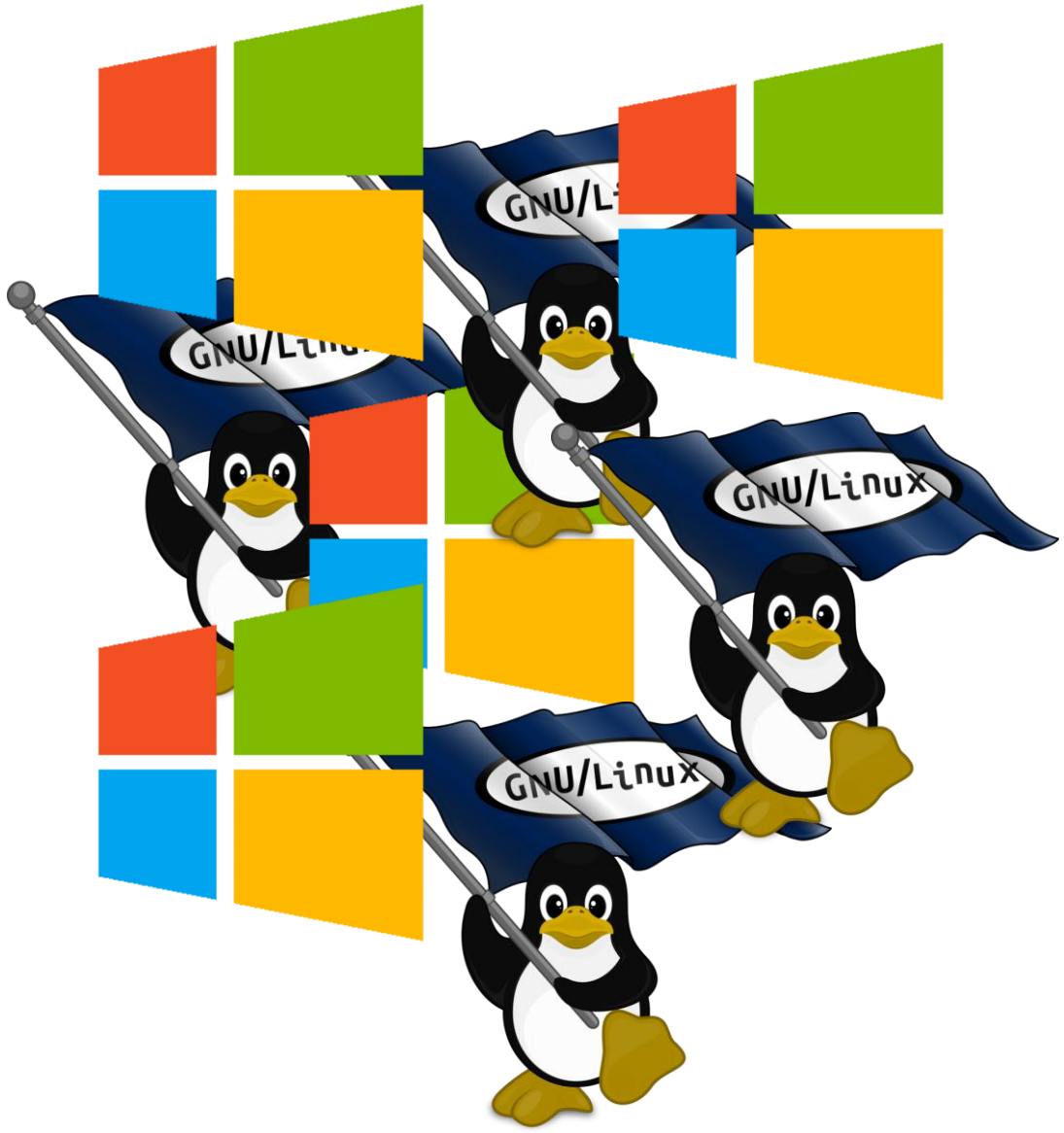
Mem - Xen overhead vs. BM (Win)



Mem - Xen overhead vs. BM (Linux)



Inter-OS interference



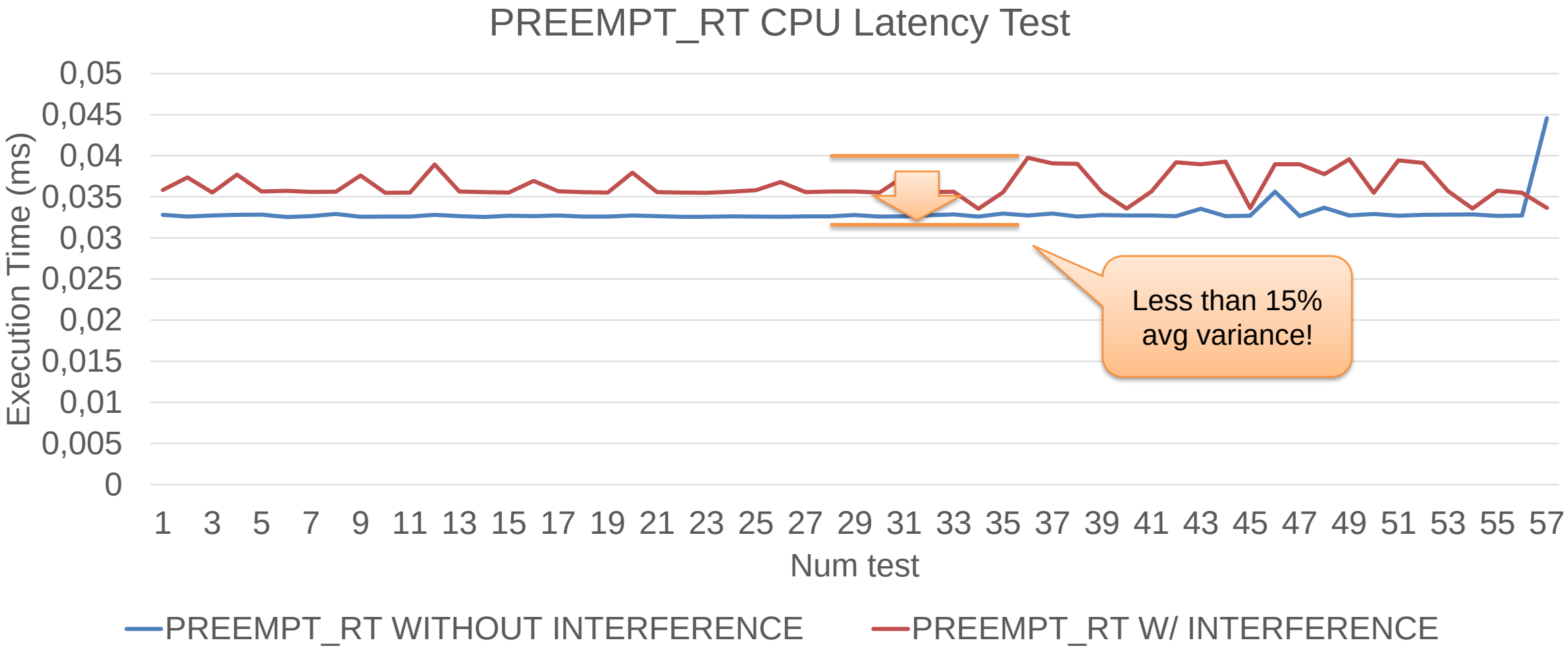
CPU - Test outline

- ✓ High priority task running in the Linux PREEMPT_RT partition that calculates a prime number
- ✓ Two low priority tasks that calculate a pi number running in the WINDOWS partition

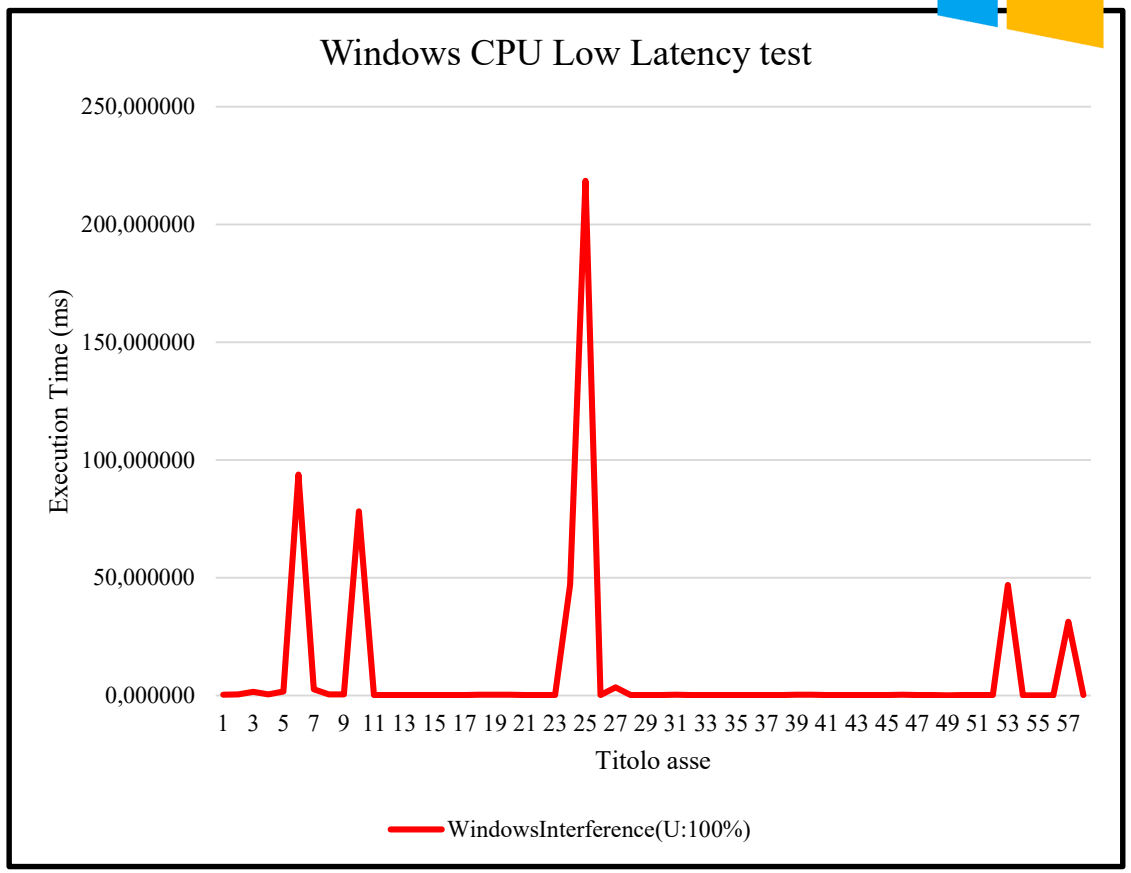
Btw…few linux RT extensions

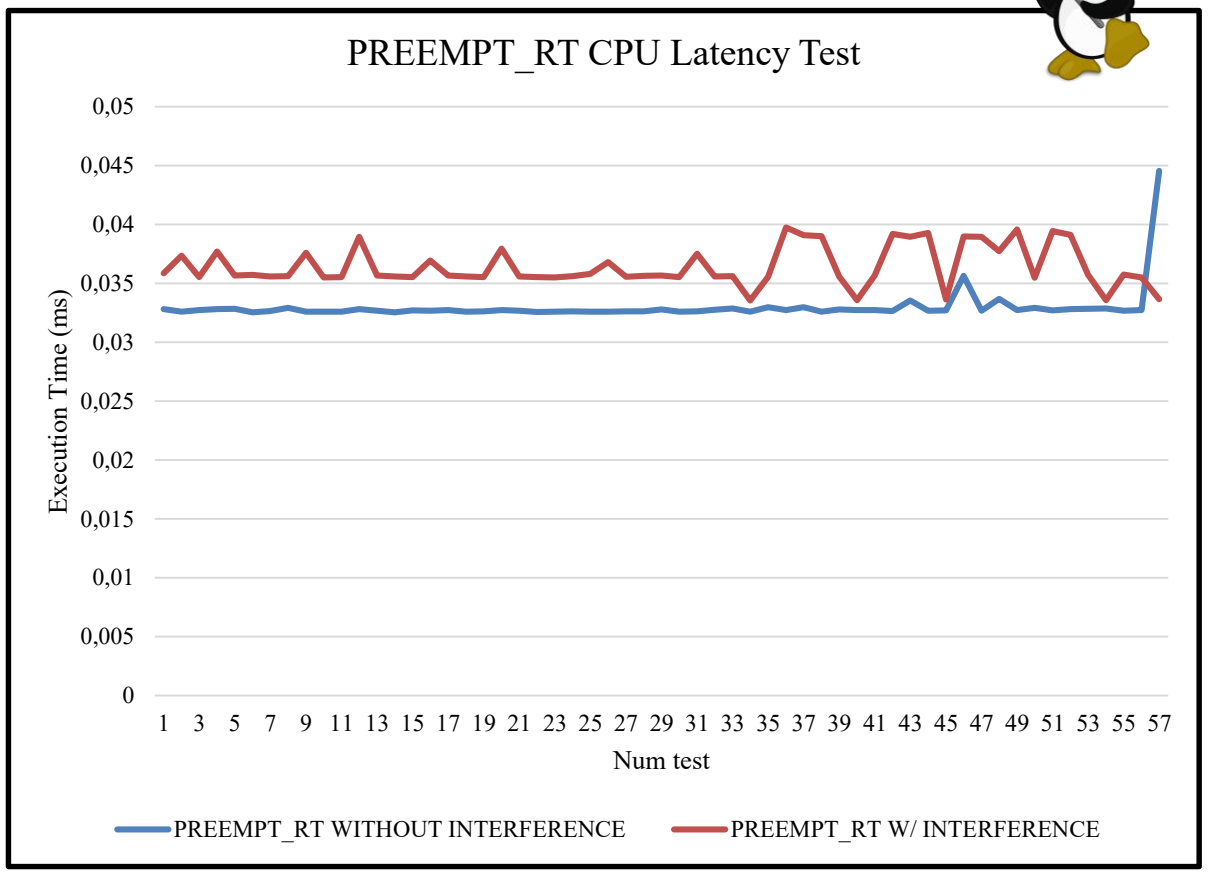
1. PREEMPT_RT patch (not mainline yet)
2. SCHED_DEADLINE is mainline
3. GRUB is (recently) mainline

CPU - cross-OS interference



CPU - comparison

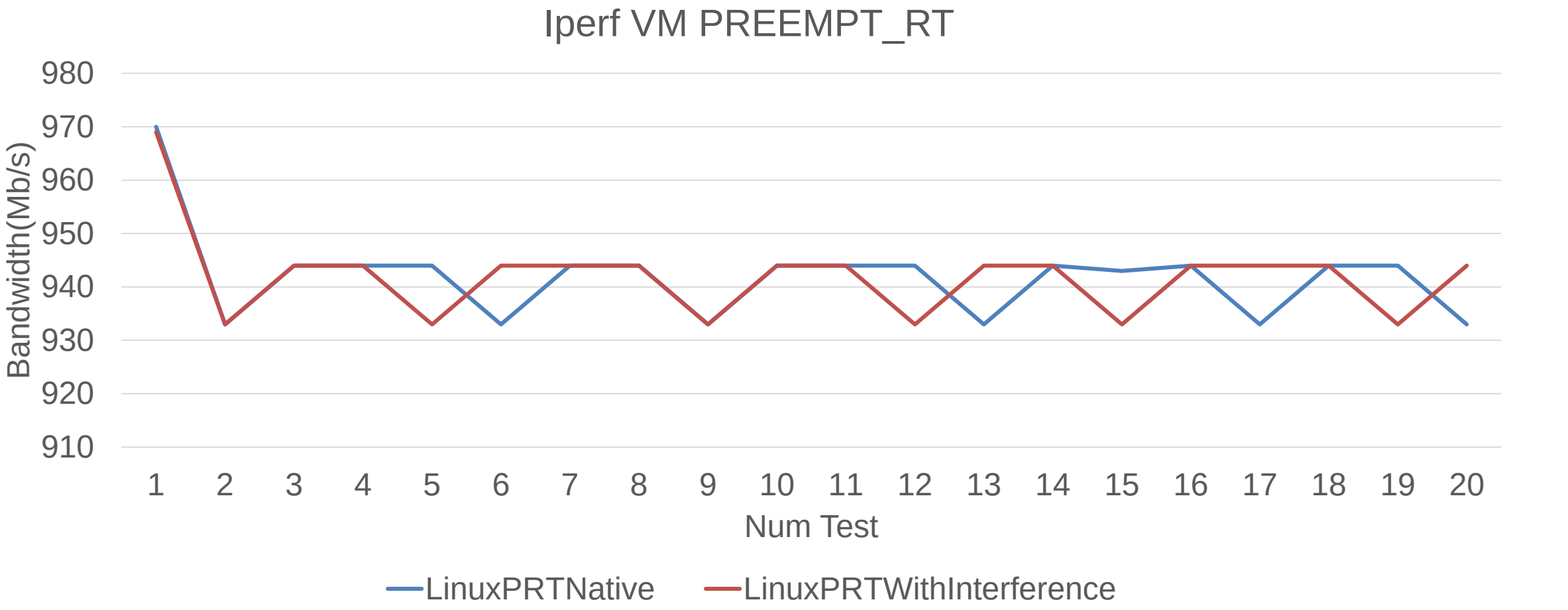




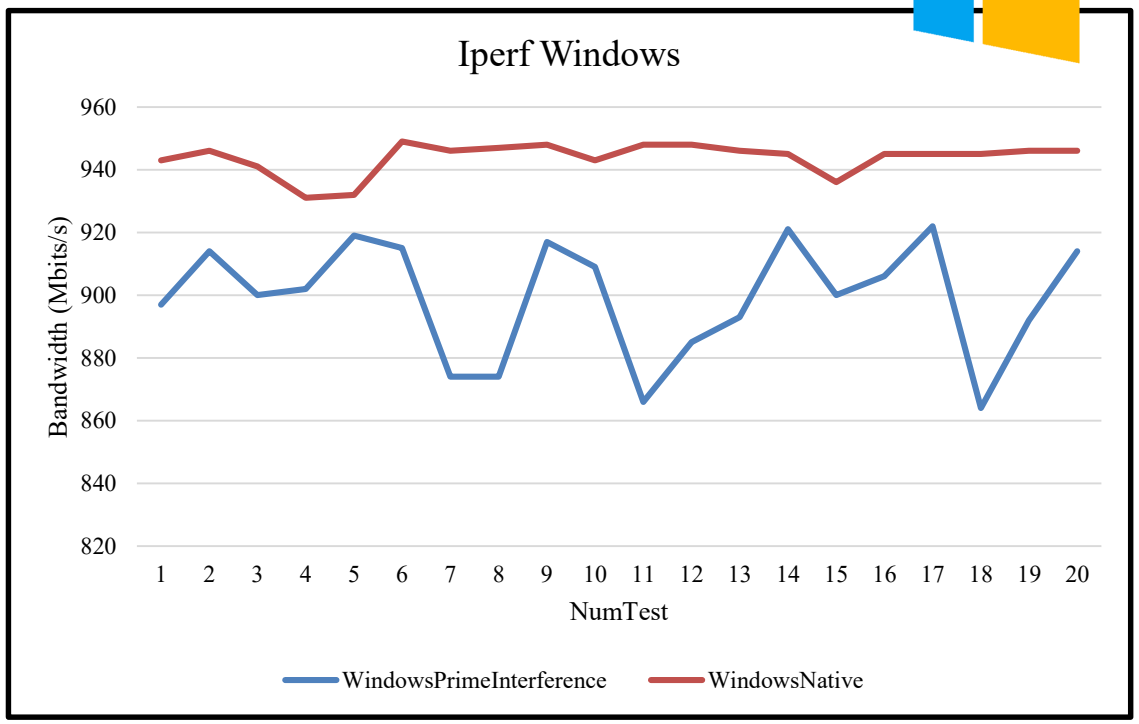
Network – Test outline

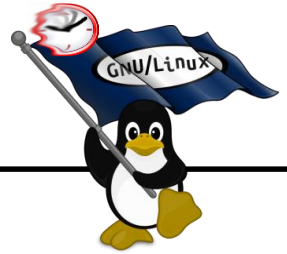
- ✓ Latency (Roundtrip time) test
  - Like a TCP version of "ping": roundtrip network communication
- ✓ PRIME95 as interfering

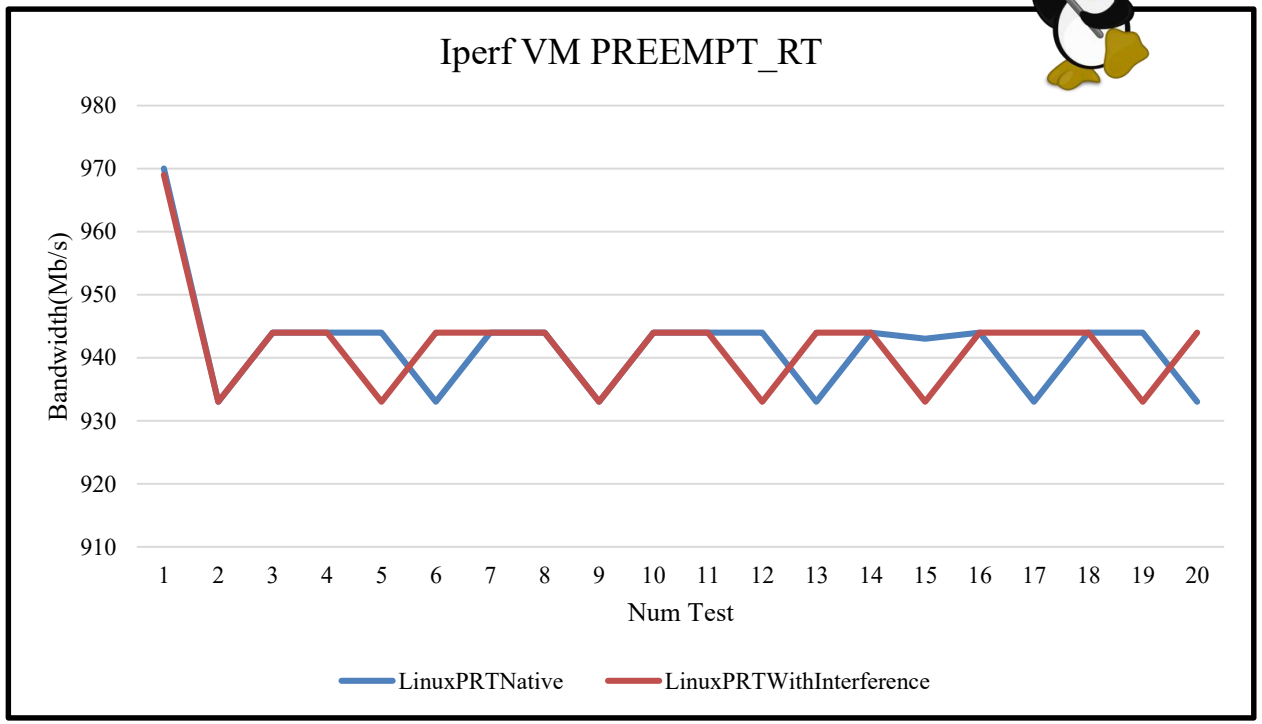
Network – OS interference



Network - comparison

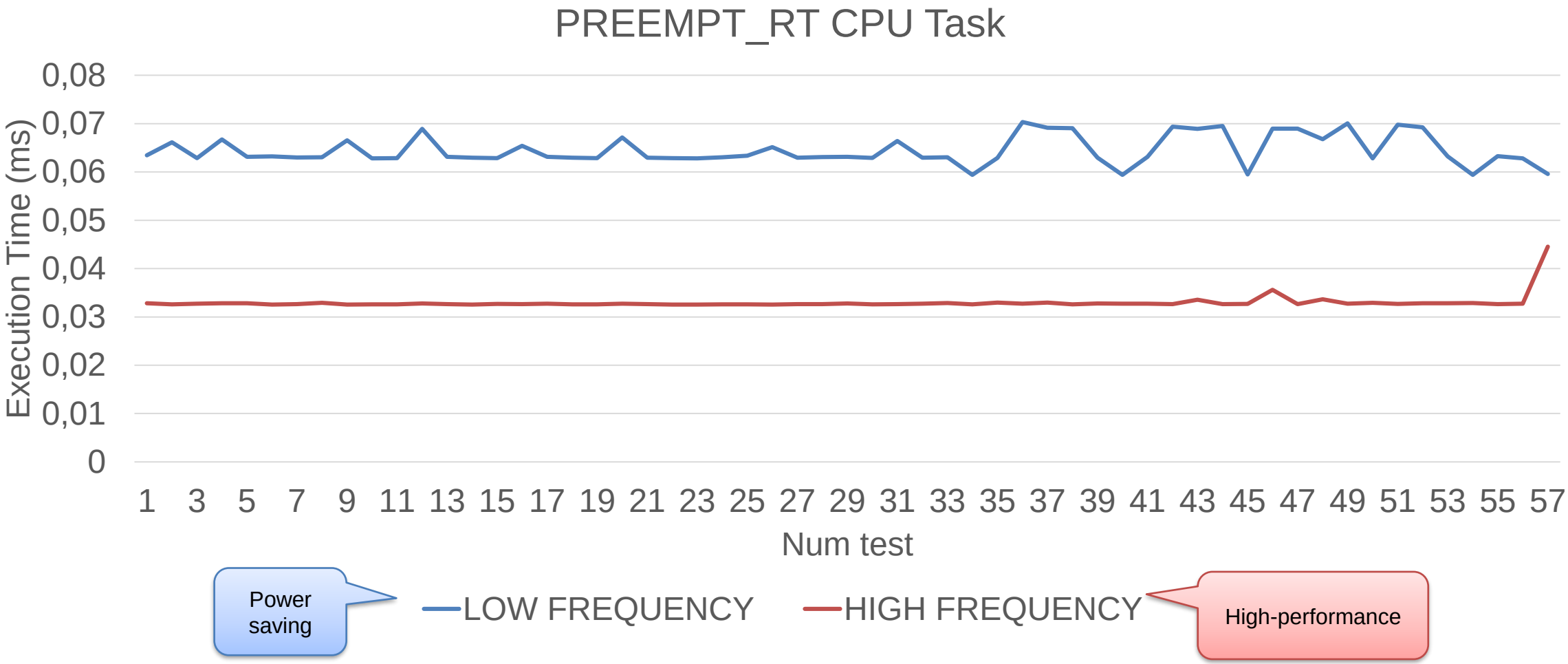






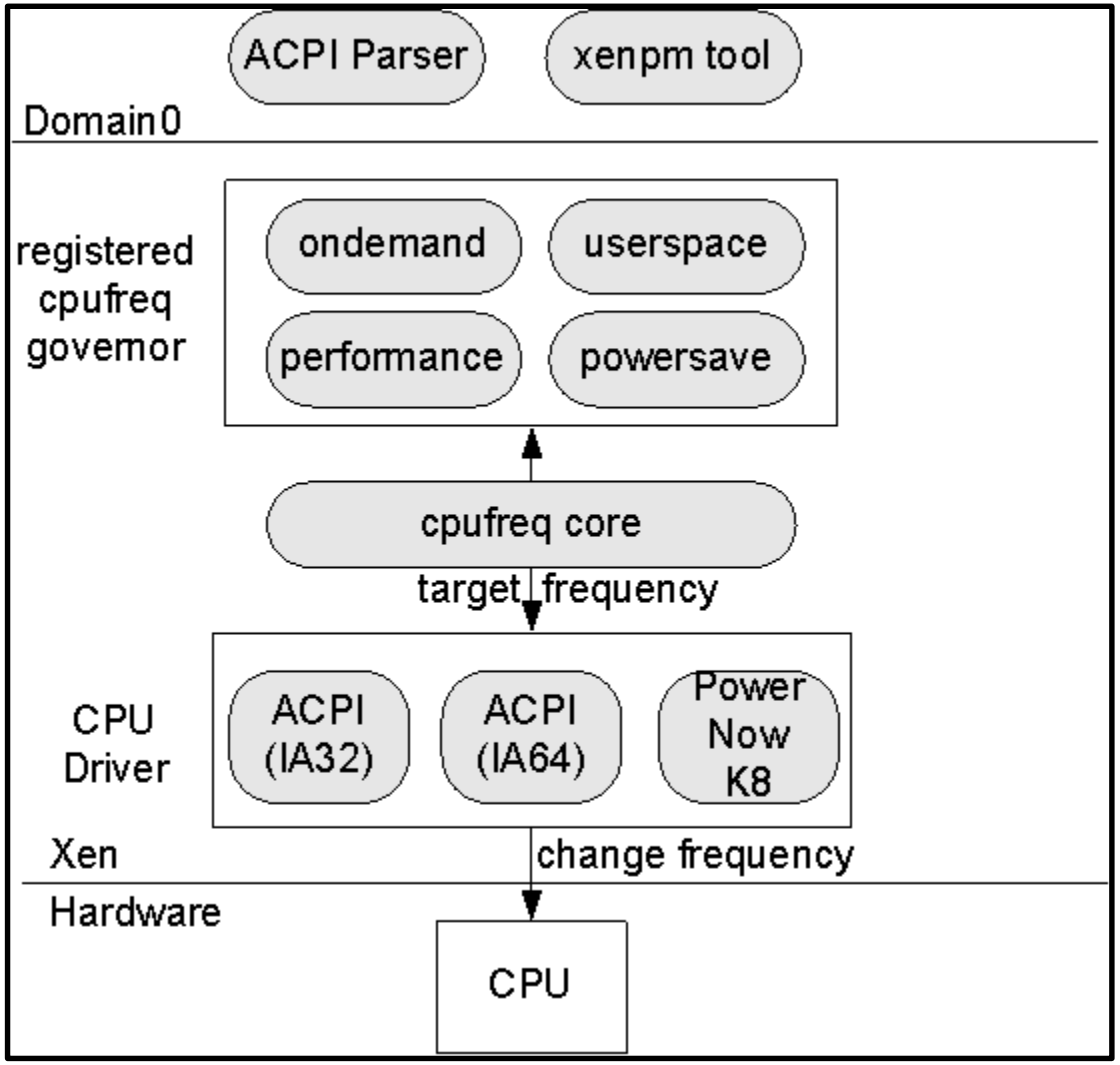
…and what about
the power manager?

Power management



Xen-based power mgmt

- Xen provides a simple way to configure the per-core frequency
  - `$ xenpm set-scaling-governor ondemand|performance|powersave`



Take-away lessons

Identified problems

- ✓ Power management still unresolved
  - Mixed Linux/Xen power management, Win?

- ✓ Windows ≠ real-time (while Linux has RT- extensions)
  - Work synergystically with Xen?

- ✓ CPU computation interference network computation on shared resource access
  - Xen-level policies

Real application (OPEN-NEXT use-case)

OPEN-NEXT

Use-case application

RT partition

- ✓ Read a file from SSD and send it multiple time over ETH/TCP-IP. We want to saturate the 1Gbps BW of each of the 8 ETH adapters (8 independent files)

Non-RT partition

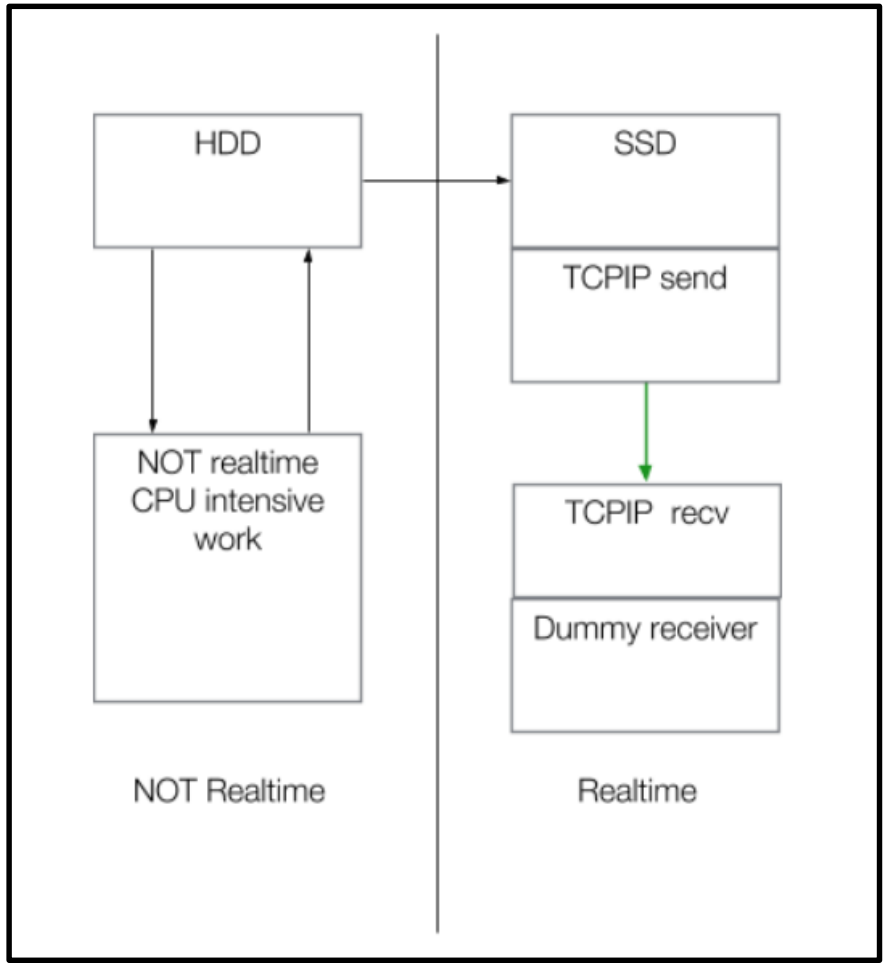
- ✓ Test 1 – HMI-like (graphical) workload. Up to 90% CPU utilization + write on HDD
- ✓ Test 2 – Also, transfer data from HDD to SSD, and sent to (dummy) printer

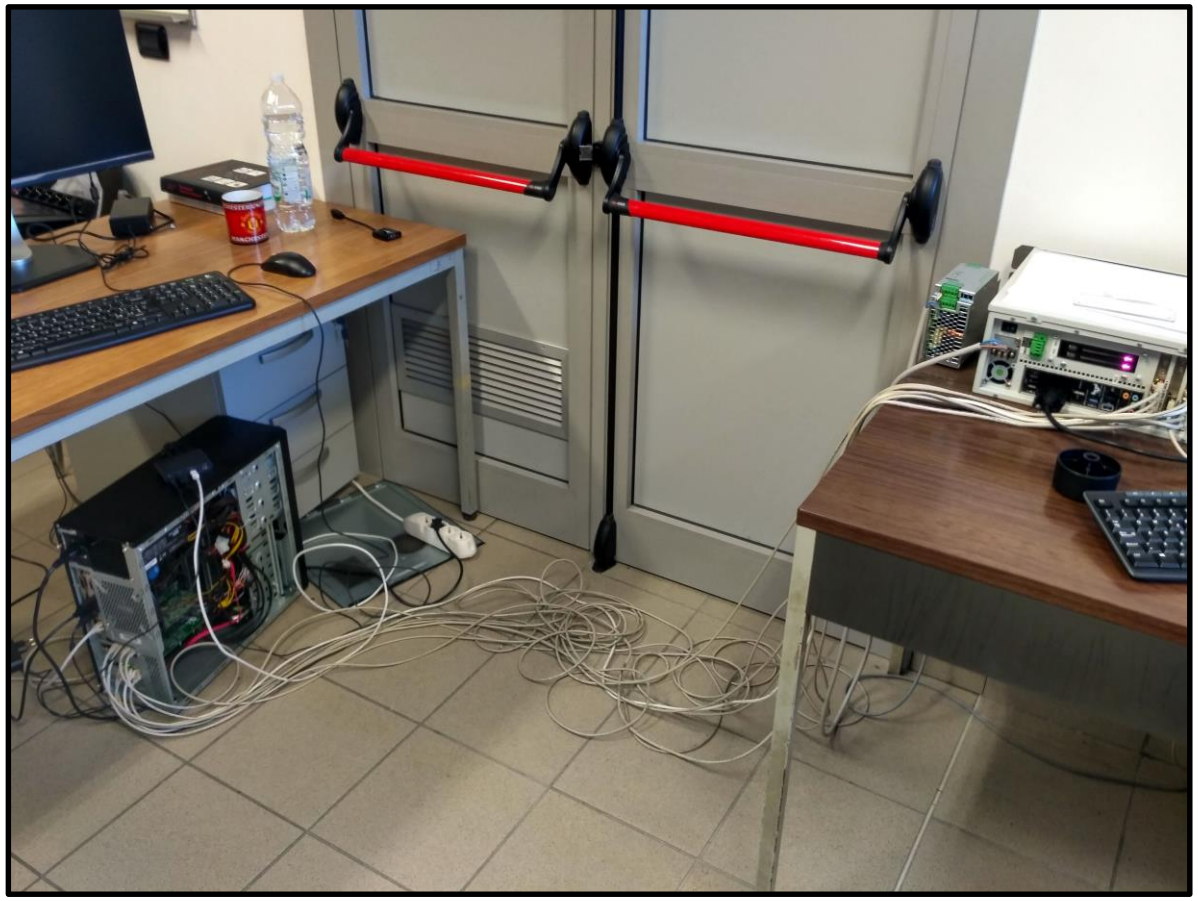
Examples

- ✓ Tile 60x60 -> 10k x 10k pixel -> 25MB file per channel; 30 tiles/min minuto; Period: 255 s
- ✓ Tile 1800x3600 -> 29k x 57k pixel -> 384MB file per channel; 12 mt/min al minuto; Period: 1.4 s
- ✓ Very small image -> 200 KB per channel -> 1k images/min; Period: 0.06 s

©2017 University of Modena and Reggio Emilia

Application scheme





Real-time OS-to-core mapping

OPEN-NEXT

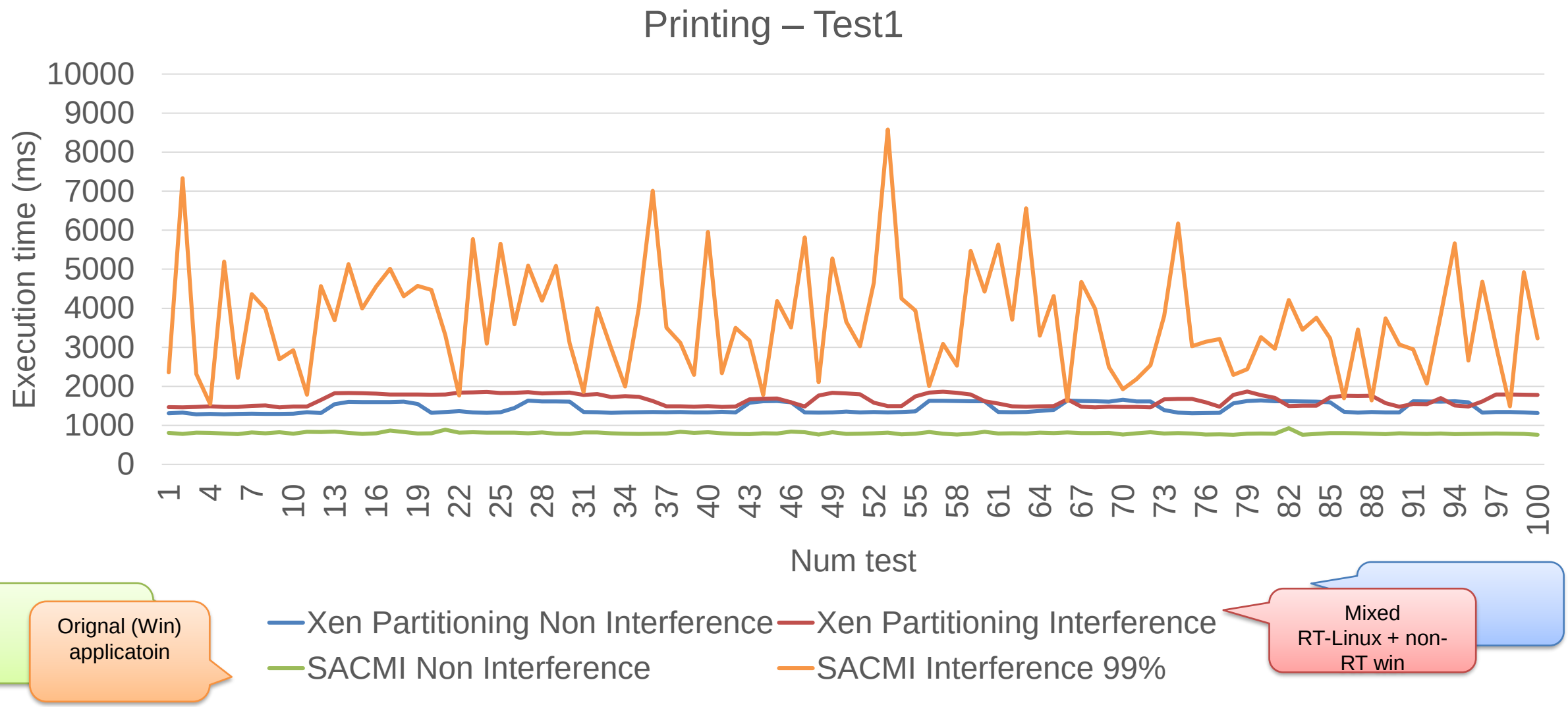
Partitioning

- ✓ OSs pinned to cores (Core 0 => Xen Dom0; Core 1 => Linux; Core 2,3 => Win)

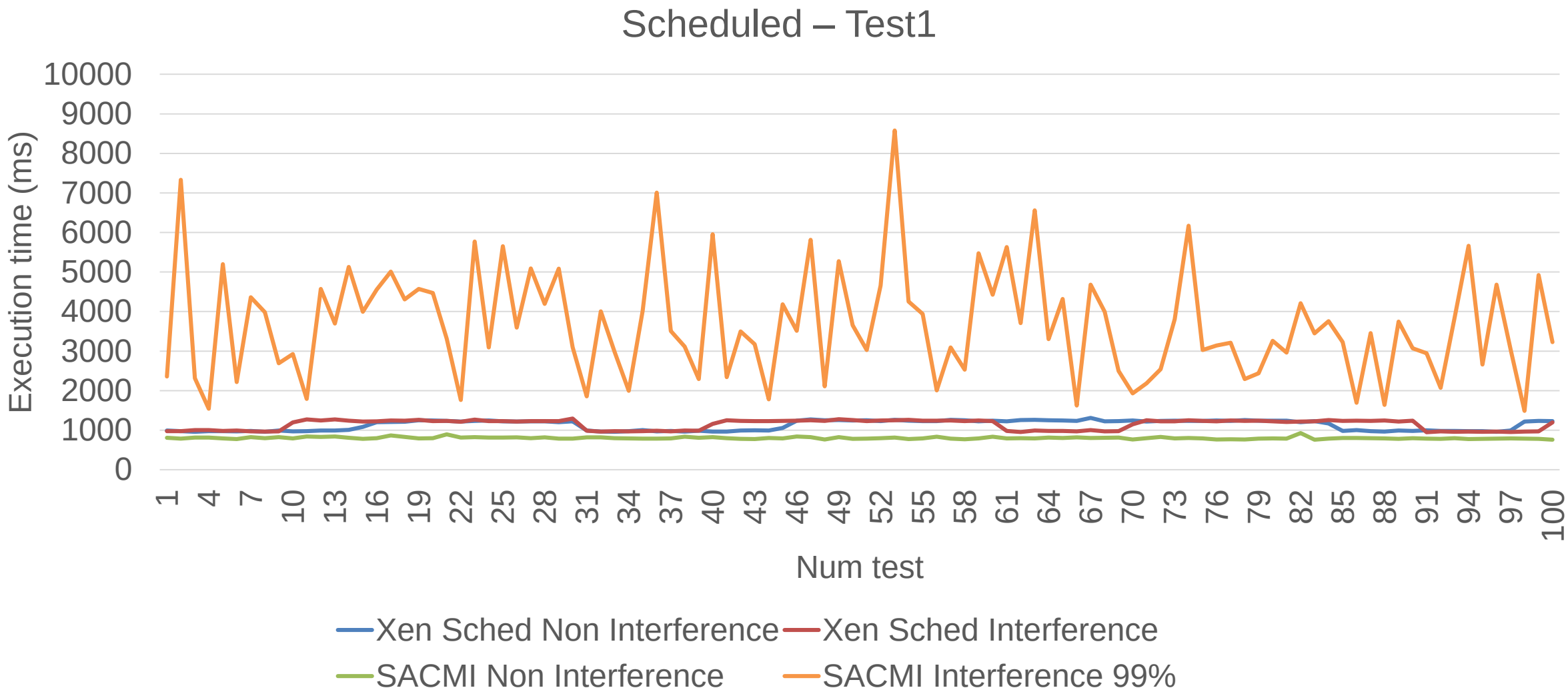
Scheduling

- ✓ OSs not pinned to cores: Xen performs allocation

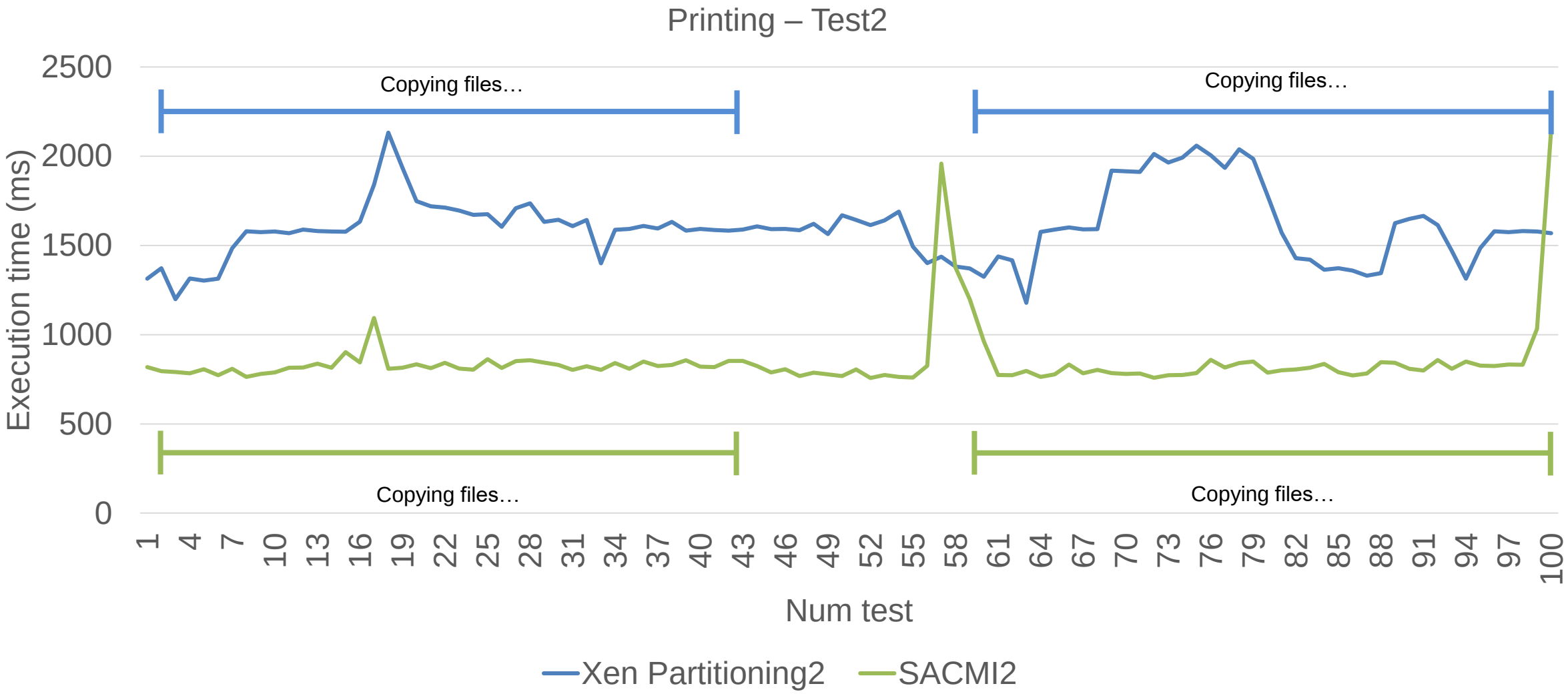
Test 1 – Partitioning approach



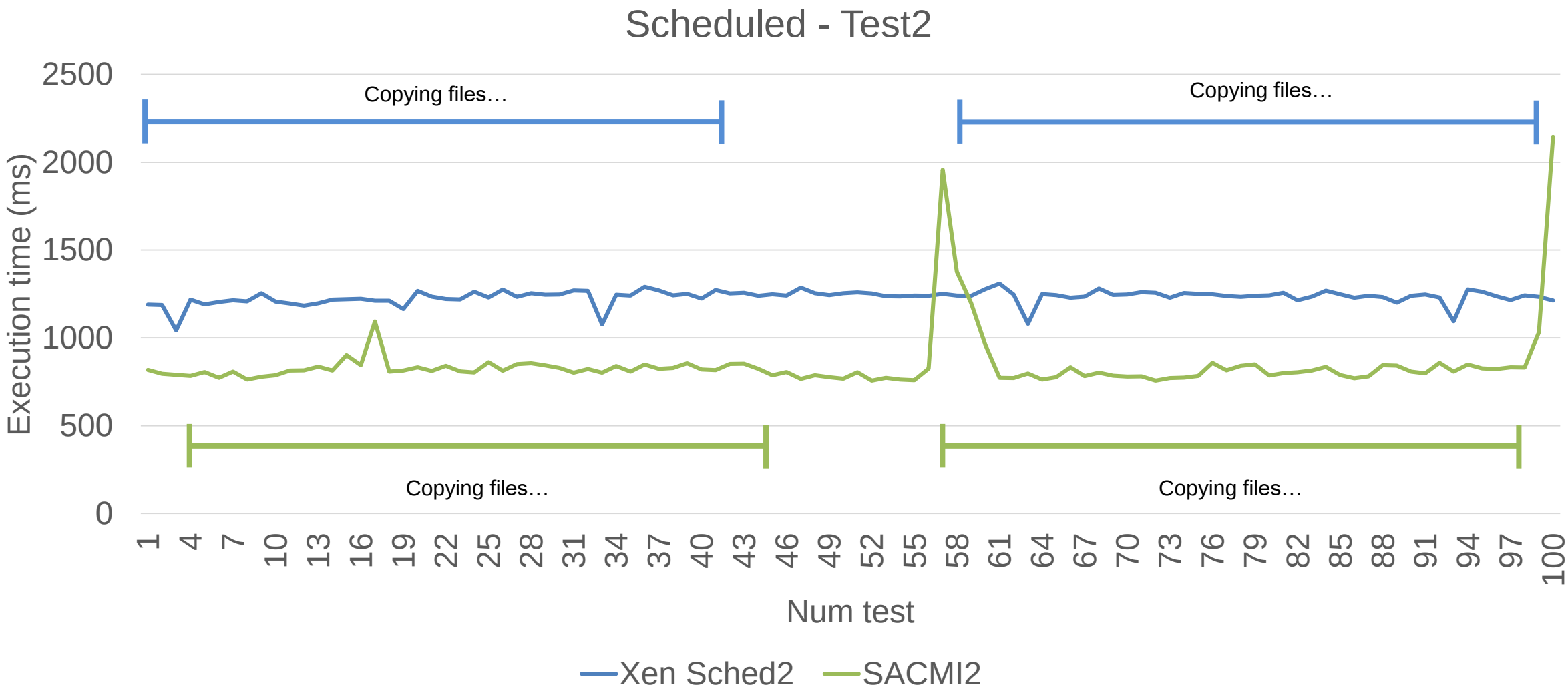
Test 1 – Scheduled approach



Test 2 - Partitioning

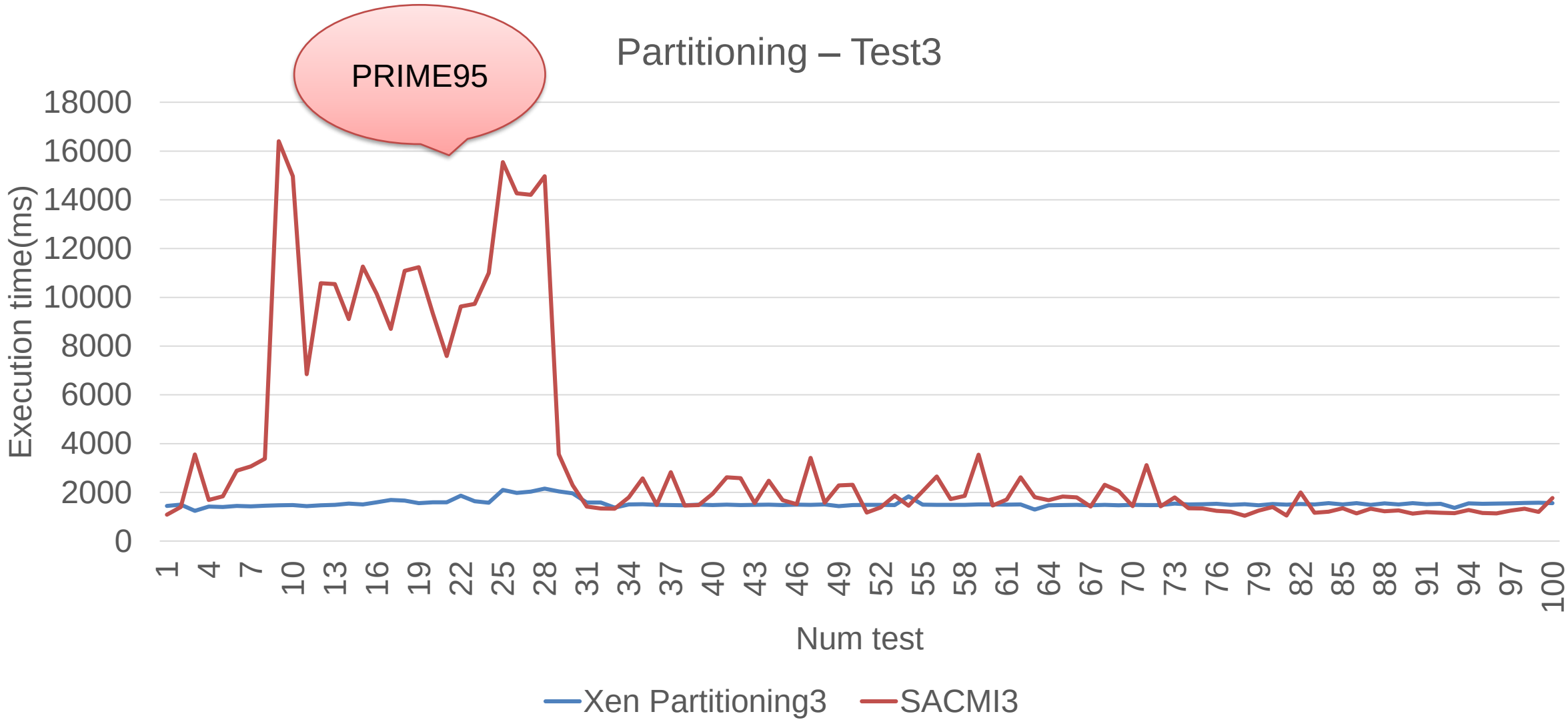


Test 2 - Scheduling

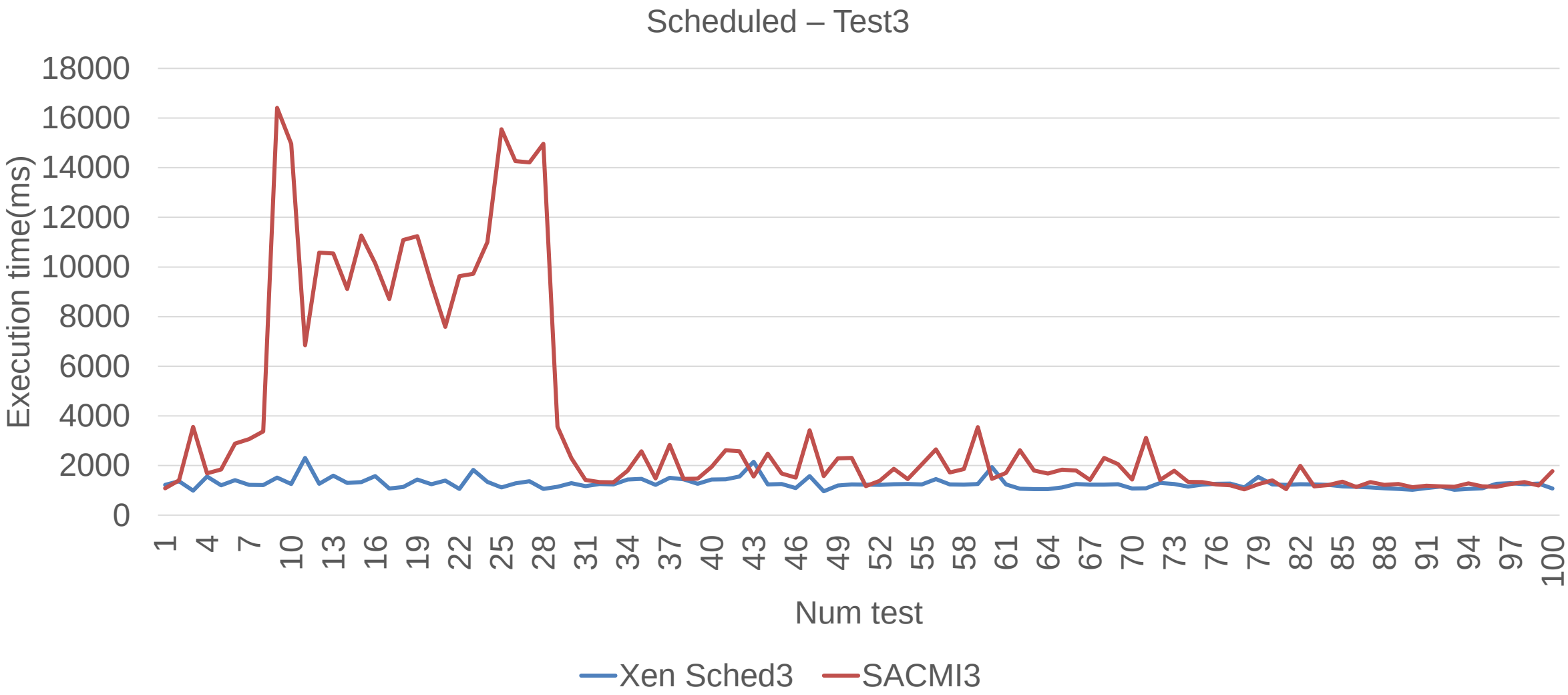


….even more interference

Test 3 - Partitioning



Test 3 - Scheduled



Take-away lessons

- ✓ Xen hypervisor adds no overhead
  - Can be effectively used to increase isolation/predictability for soft-RT systems
  - Open-source, can be modified / further improved
- ✓ Windows uneffective in RT: Linux w/RT extensions provide quite good results
  - Some tuning is beneficial
  - No modifications (yet) to the source code
- ✓ Still, power management is an issue
  - Platform-specific solutions?

Thank you!

OPEN-NEXT

Paolo Burgio

Roberto Cavicchioli
Ignacio Sanudo Olmedo
Marco Solieri