ARM Architecture-based System Virtualization: Xen ARM open source software project

Sang-bum Suh Vice President, Jae-min Ryu Research Staff
{sbuk.suh, jm77.ryu}@samsung.com
S/W Platform Team
DMC Research Center
SAMSUNG Electronics Co.

2011.08.

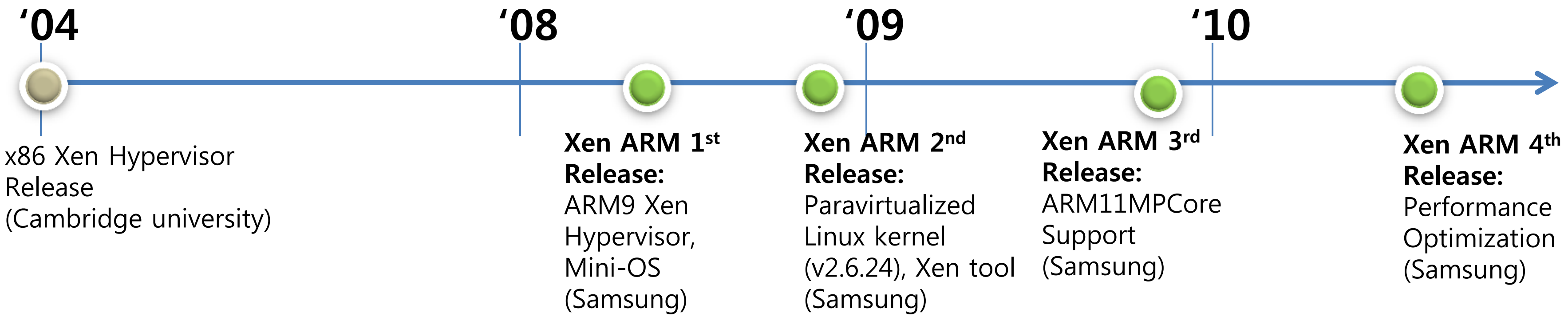
Xen Summit North America 2011

Agenda

- Overview
  - History of Xen ARM
  - Use Cases
- Xen ARM: Core
  - Xen ARM Virtualization
  - Performance Comparison
- Xen ARM Application: Security
  - Mobile Malware
  - Access Control
- Xen ARM Application: Real-time
  - Xen ARM: Preemption
  - Real-time Performance

Overview

History of Xen ARM



'04

x86 Xen Hypervisor Release (Cambridge university)

'08

Xen ARM 1st Release: ARM9 Xen Hypervisor, Mini-OS (Samsung)

'09

Xen ARM 2nd Release: Paravirtualized Linux kernel (v2.6.24), Xen tool (Samsung)

Xen ARM 3rd Release: ARM11MPCore Support (Samsung)

'10

Xen ARM 4th Release: Performance Optimization (Samsung)

Xen-ARM Open Source Community

- Samsung leads the Xen ARM project
- http://wiki.xensource.com/xenwiki/XenARM

Supported Hardware & Guest OS

- **ARM926EJ-S (i.MX21, OMAP5912)**
- **Xscale 3rd Generation Architecture (PXA310, Samsung SGH- i780)**
- **ARM1136/ARM1176(Core Only)**
- **Goldfish (QEMU Emulator)**
- **Versatile Platform Board**
- **ARM11MPCore (Realview PB11MP)**
- **Cortex-A9 (Tegra250)**

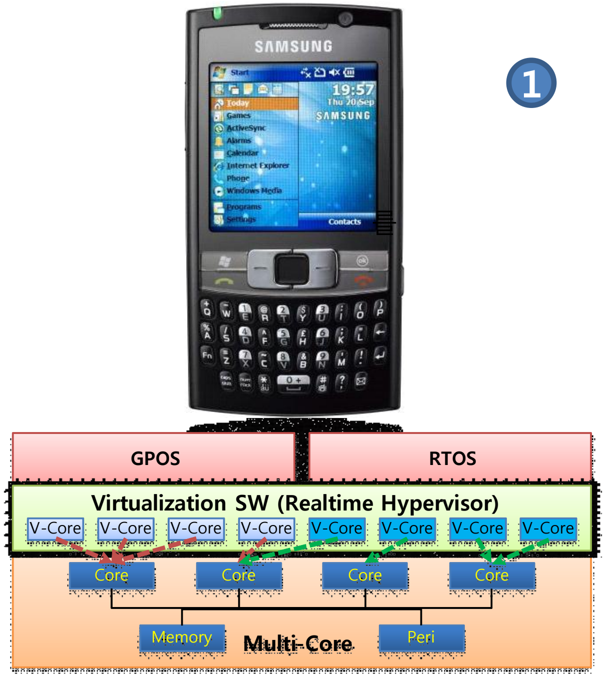
- **Linux v2.6.11, v2.6.18, v2.6.21, v2.6.24, v2.6.27 (multicore supported)**
- **uC/OS-II**

Use Cases

1 – **HW Consolidation:** AP(Application Processor) and BP(Baseband Processor) can share multicore ARM CPU SoC in order to run both Linux and Real-time OS efficiently.

2 – **OS Isolation:** important call services can be effectively separated from downloaded third party applications by Xen ARM combined with access control.

3 – Rich User Experience: multiple OS domains can run concurrently on a single smartphone.



1

GPOS

RTOS

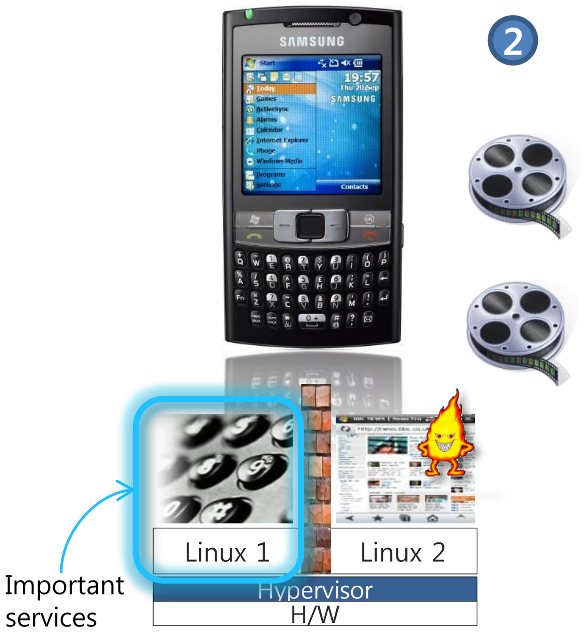
Virtualization SW (Realtime Hypervisor)

V-Core V-Core V-Core V-Core V-Core V-Core V-Core V-Core

Core Core Core Core

Memory Multi-Core Peri

AP SoC +BP SoC -> Consolidated Multicore SoC



2

Important services

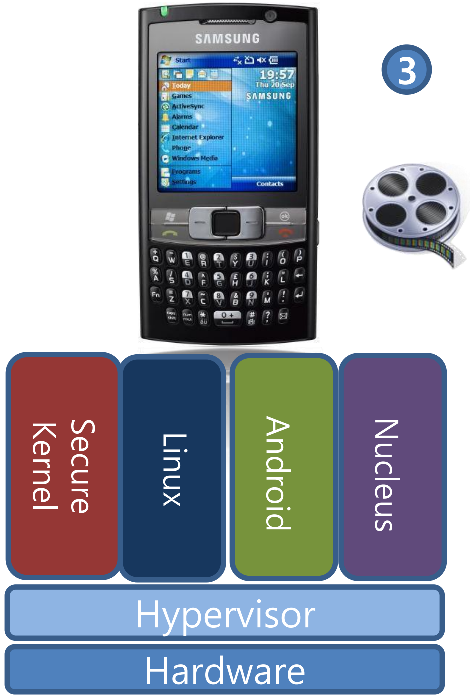
Linux 1

Linux 2

Hypervisor

H/W

Secure Smartphone



3

Secure Kernel

Linux

Android

Nucleus

Hypervisor

Hardware

Rich Applications from Multiple OS

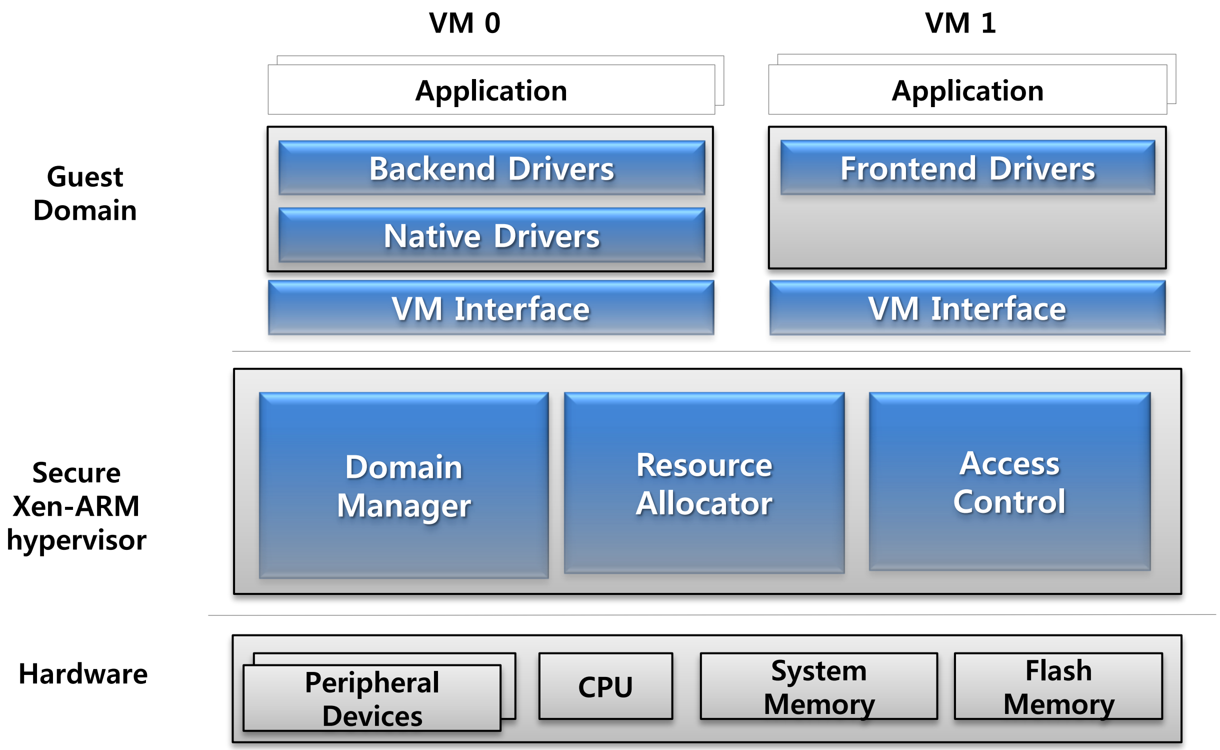
Xen ARM: Core

Xen ARM Virtualization

Goals

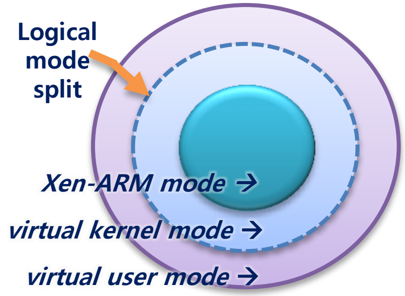
- Light weight virtualization for secure 3G/4G mobile devices
  - High performance hypervisor based on ARM processor
  - Fine-grained access control fitted to mobile devices

Architecture of Xen-ARM

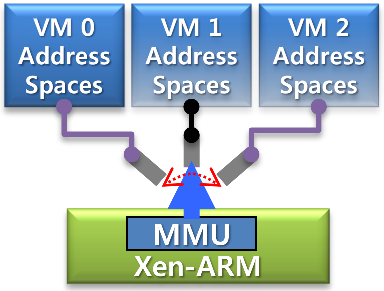


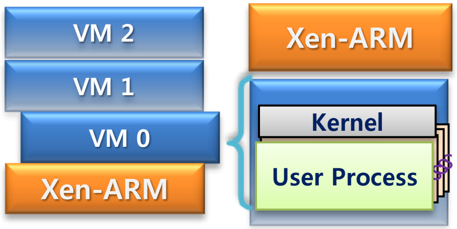
Xen ARM Virtualization

Overview



- **CPU virtualization**
  - Virtualization requires 3 privilege CPU level, but ARM supports 2 level
    - Xen-ARM mode: supervisor mode (most privileged level)
    - Virtual kernel mode: User mode (least privileged level)
    - Virtual user mode: User mode (least privileged level)

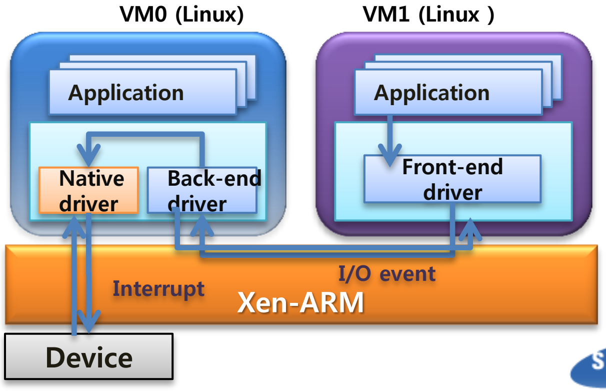




Physical Address Space

Virtual Address Space

- **Memory virtualization**
  - VM's own memory should be protected from others
    - Xen-ARM switches VM's virtual address space using MMU
    - VM is not allowed to manipulate MMU directly



- **I/O virtualization**
  - Split driver model of Xen-ARM
    - Client & Server architecture for shared I/O devices
      - Client: frontend driver
      - Server: native/backend driver

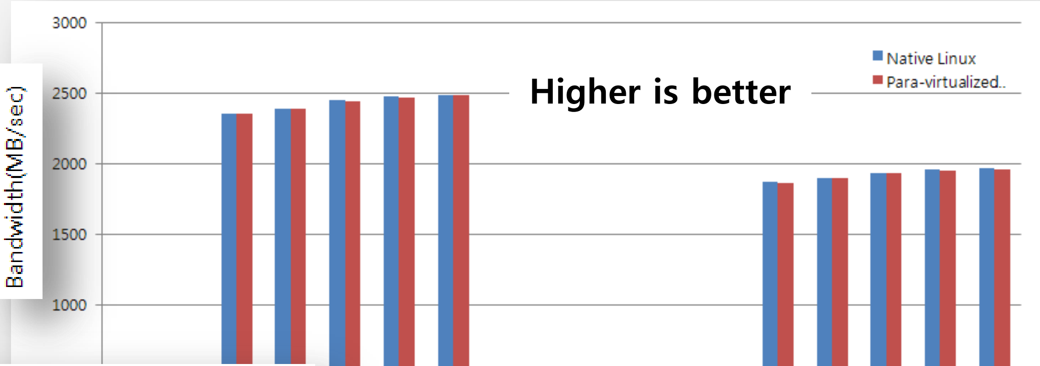
Performance Comparison

Micro-benchmark Results

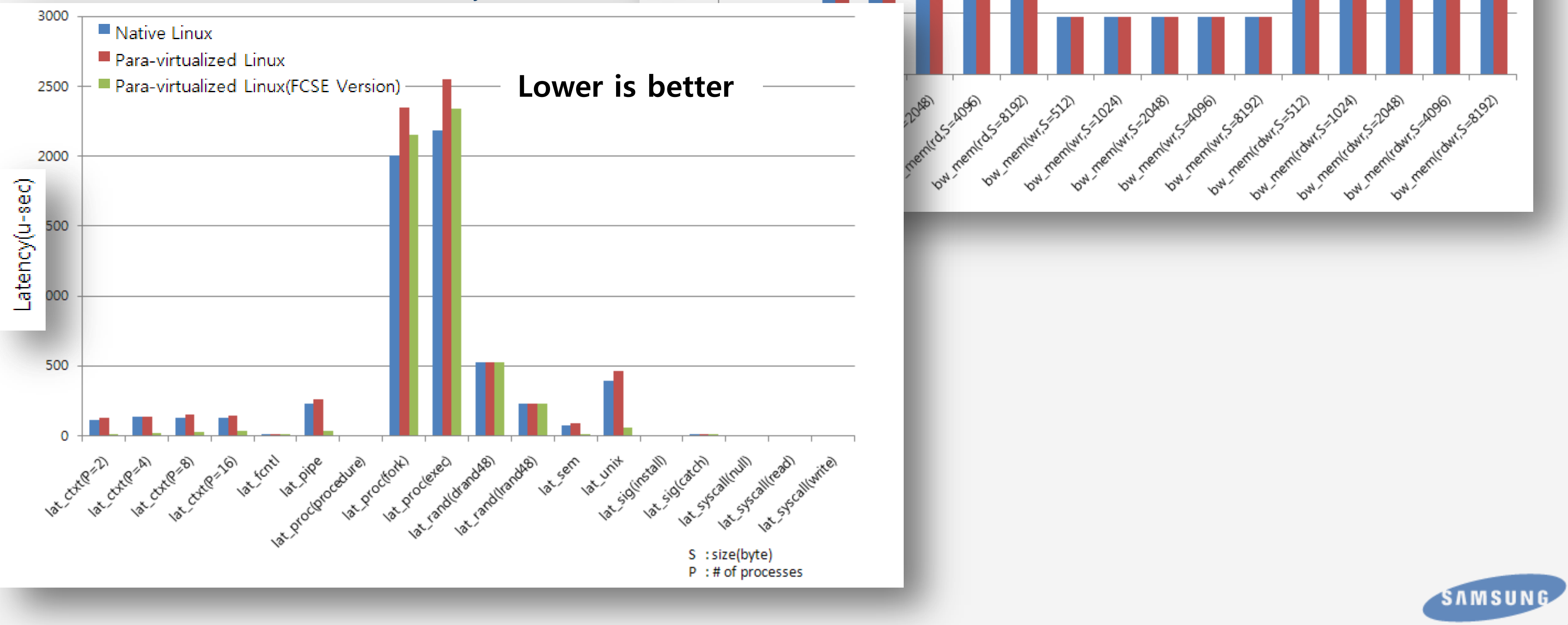
Micro-benchmark Results

- **Evaluation Environments : Samsung Blackjack Phone**
  - CPU : Xscale PXA310, 624MHz
  - L1 Cache : 32KB + 32KB
  - L2 Cache : 256KB (Disabled)
  - Memory : 128MB
  - Guest OS: Linux-2.6.21

LMBENCH Micro Benchmark (Bandwidth)



LMBENCH Micro Benchmark (latency)



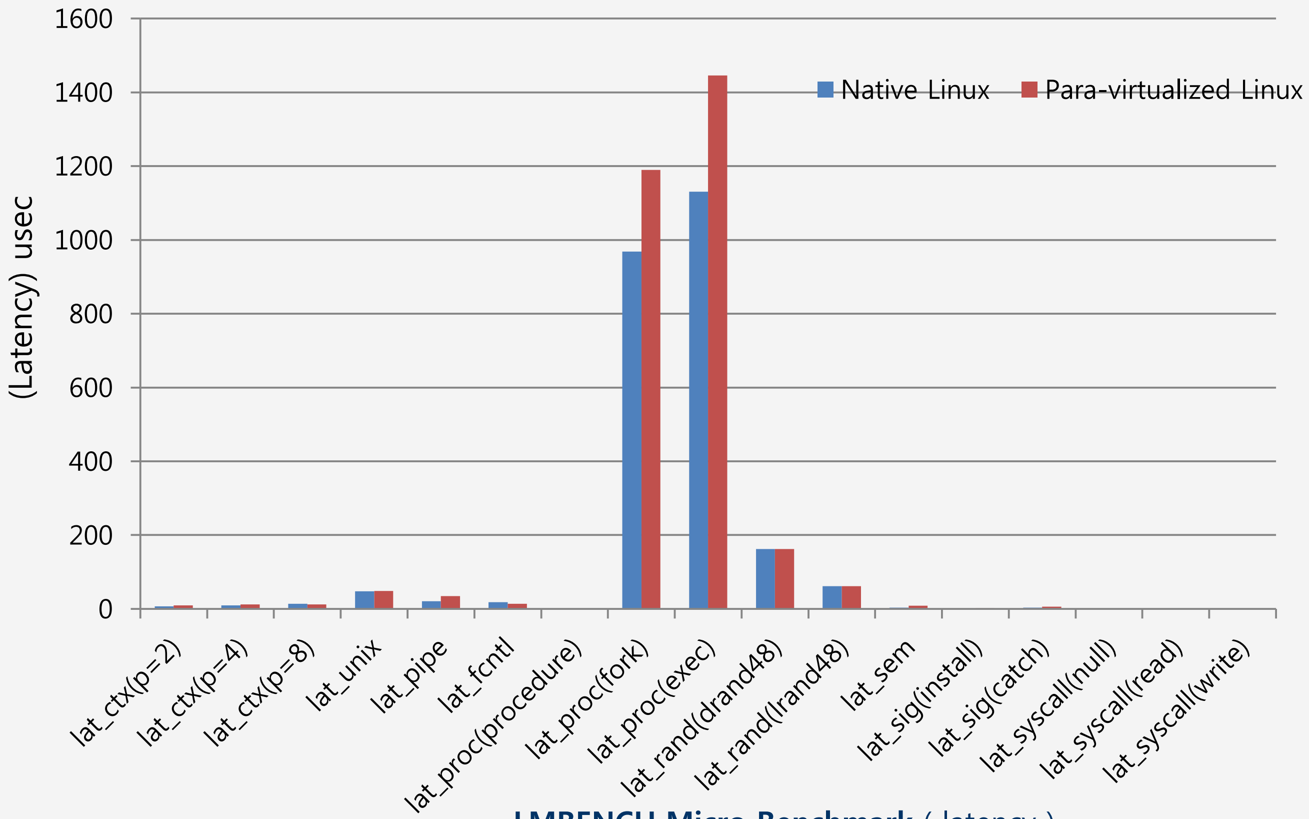
S : size(byte)
P : # of processes

Performance Comparison

Micro-benchmark Results

- **Evaluation Environments : nVidia Tegra250**
  - CPU : Cortex-A9 1GHz Dual Core
  - L1 Cache : 32KB + 32KB
  - L2 Cache : 1MB
  - Memory : 1GB
  - Guest OS: Linux-2.6.29

Lower is better



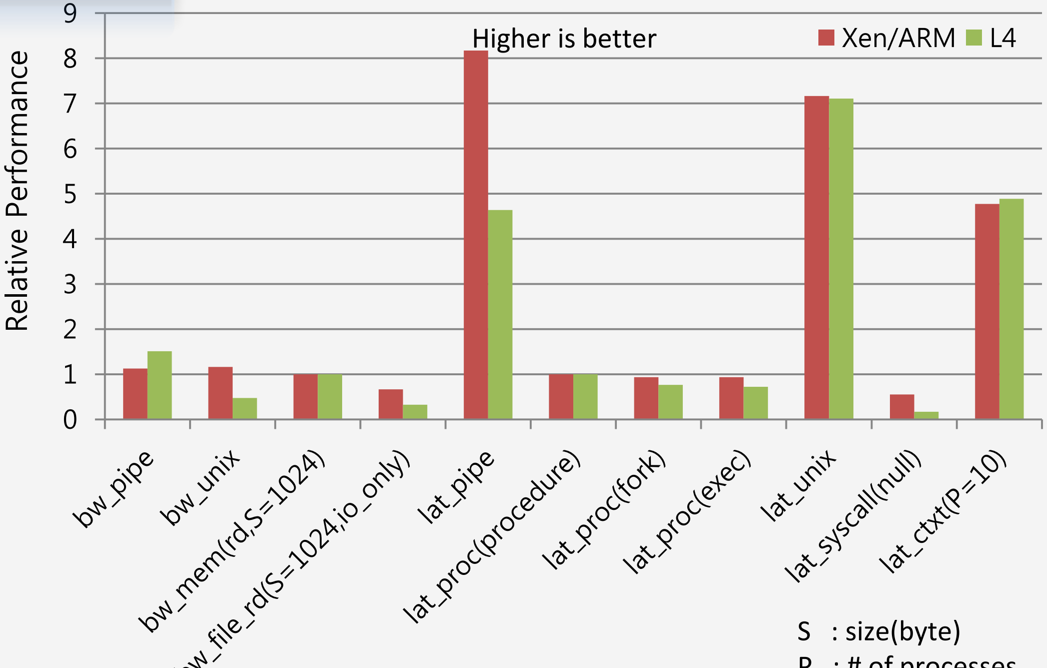
LMBENCH Micro Benchmark (latency)

Performance Comparison

Benchmark Results

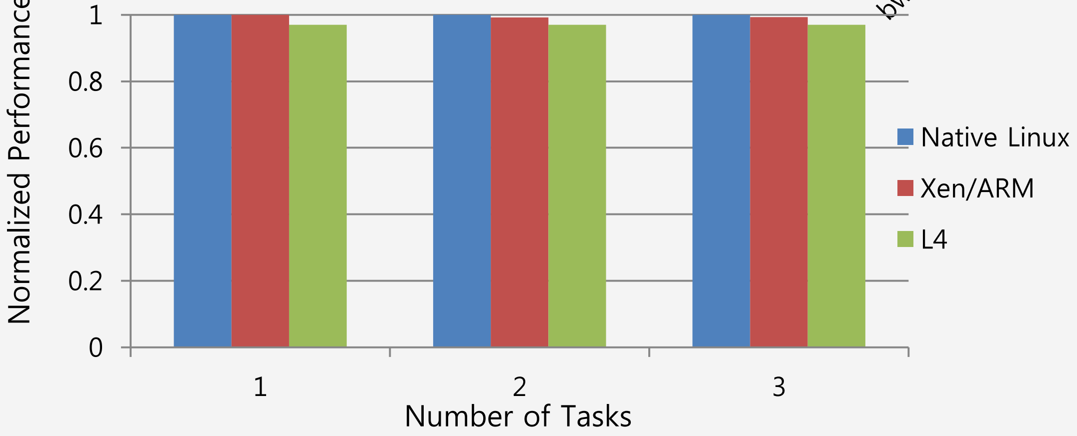
- **Evaluation Environments : Samsung Blackjack Phone**
  - CPU : Xscale PXA310, 624MHz
  - L1 Cache : 32KB + 32KB
  - L2 Cache : 256KB (Disabled)
  - Memory : 128MB
  - Guest OS: Linux-2.6.21

LMBENCH Micro Benchmark (latency)



S : size(byte)
P : # of processes

AIM7 Macro Benchmark

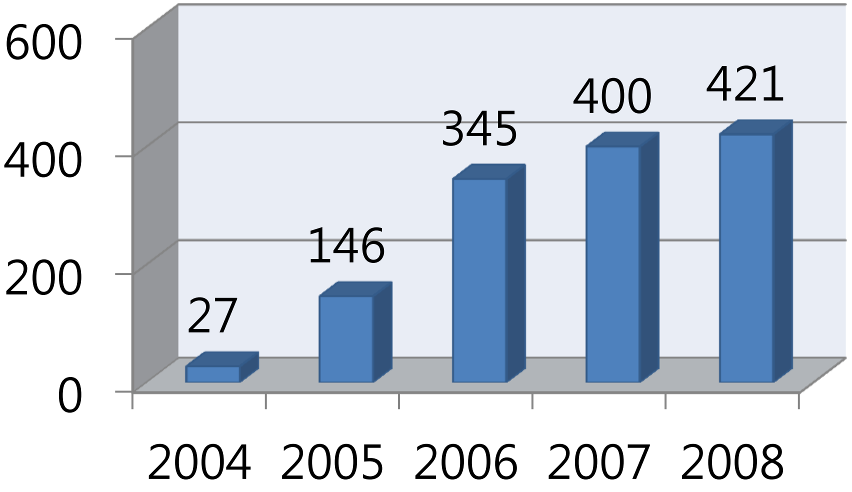


Xen ARM Application For Security

Mobile Malware

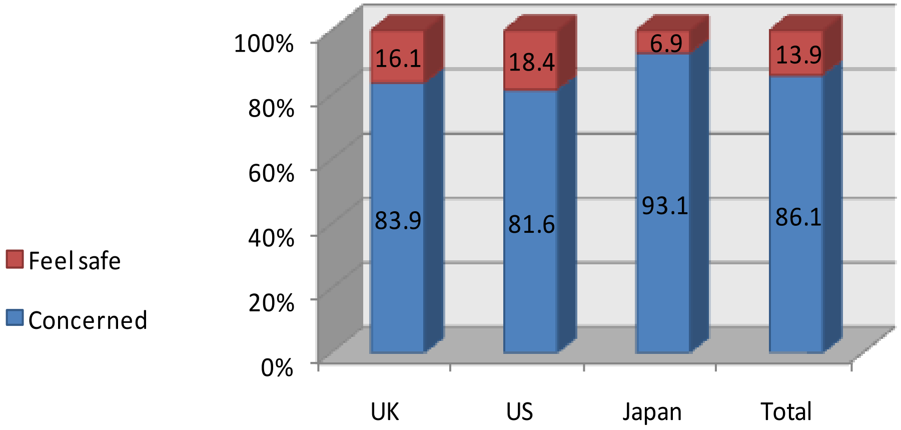
- Number of mobile malware
  - More than 420 mobile phone viruses (2008)
  - Tens of thousands of infections worldwide

[Source: F-Secure]



- Concerns about mobile phone security – by market

[Source: McAfee]

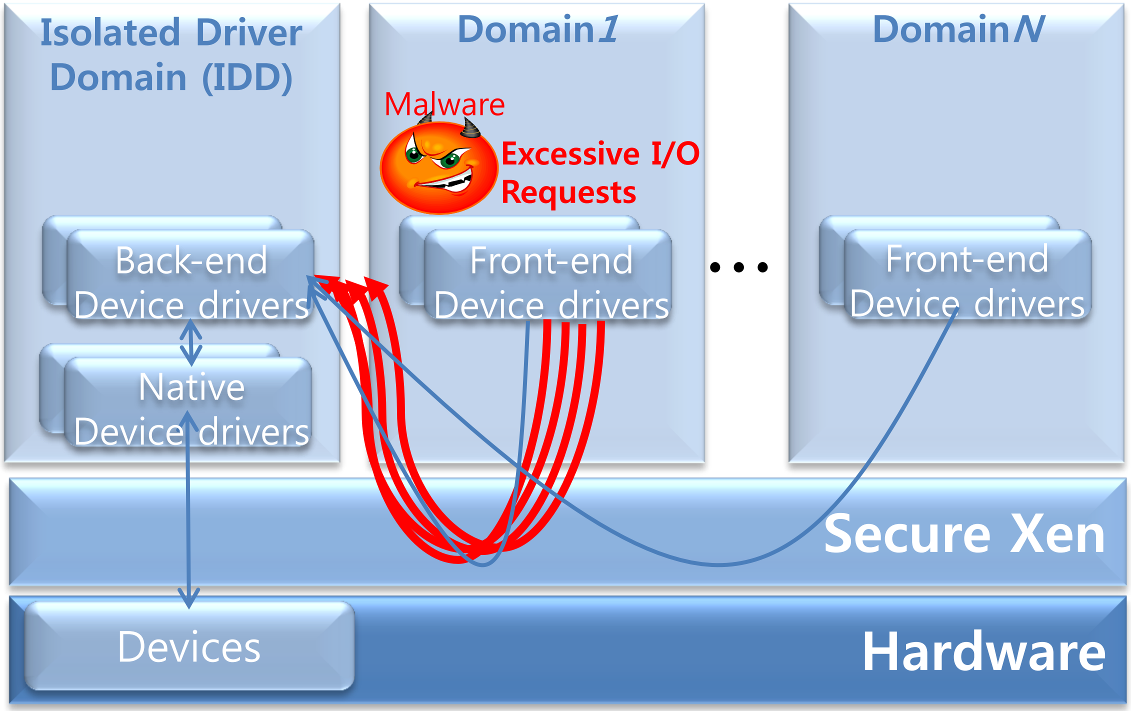


Access Control

Definition: Access control is a system which enables an authority to control area and resources in a given physical facility or computer-based information system [source: Wikipedia]

Problem with performance isolation

- **Availability threat: denial of service (DoS) attack from a compromised domain in a mobile device**
  - **CPU overuse:** a greater share of CPU time than initial allocation
  - **Performance degradation:** The Performance of other domains that share the same I/O device with the compromised domain
  - **Battery drain**

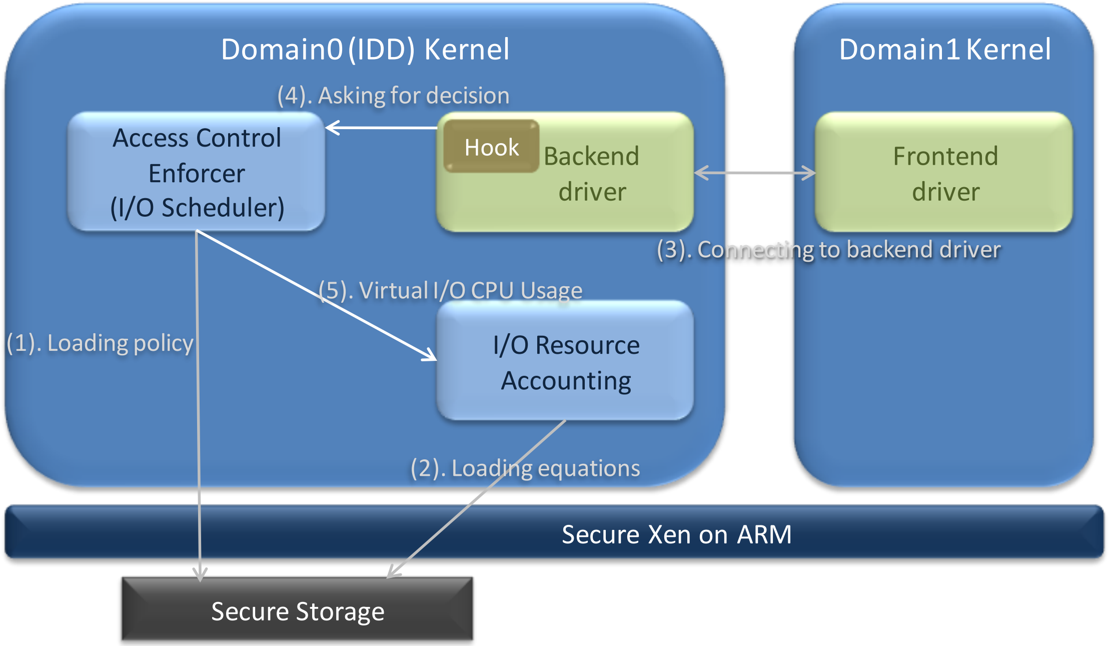


Access Control

Approach

- **Fine-grained I/O access control module in the IDD and coarse-grained access control module in Xen**
- **Estimation of CPU consumption by each virtual I/O operation using regression analysis**
  - Network and storage devices
- **I/O access control enforcement based on the policy and regression equations**

Target HW spec: XScale 624MHz, 128MB DRAM



Regression Analysis: Network

$f_{NET,\ Tx}(x) = 370.18 + 0.01 * x$

Regression Analysis: Storage

$f_{MTD,READ}(x) = 250 + 0.24 * x$

$f_{MTD,READOOB}(x) = 533 + 0.06 * x$

$f_{MTD,WRITE}(x) = 160 + 0.24 * x$

$f_{MTD,WRITEOOB}(x) = 583$

$f_{MTD,ERASE}(x) = 153.33 + 0.02 * x$

$f_{MTD,ISBADBLOCK}(x) = 58$

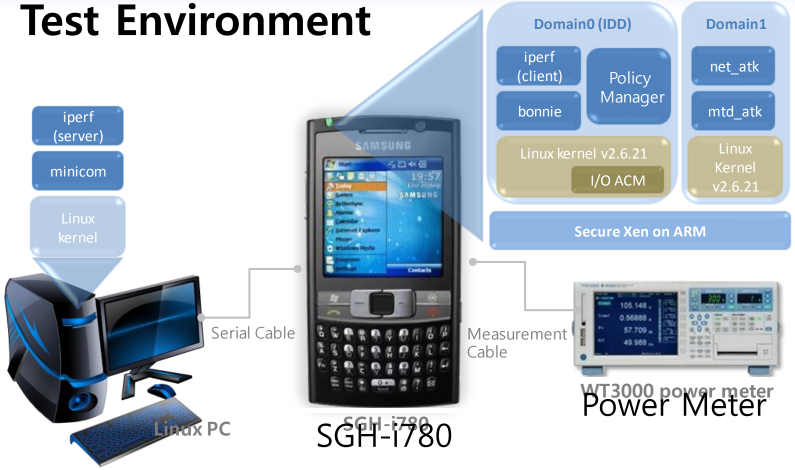
$f_{MTD,MARKBAD}(x) = 60$

x: bytes, f(x): usec

Access Control

Effectiveness

Test Environment

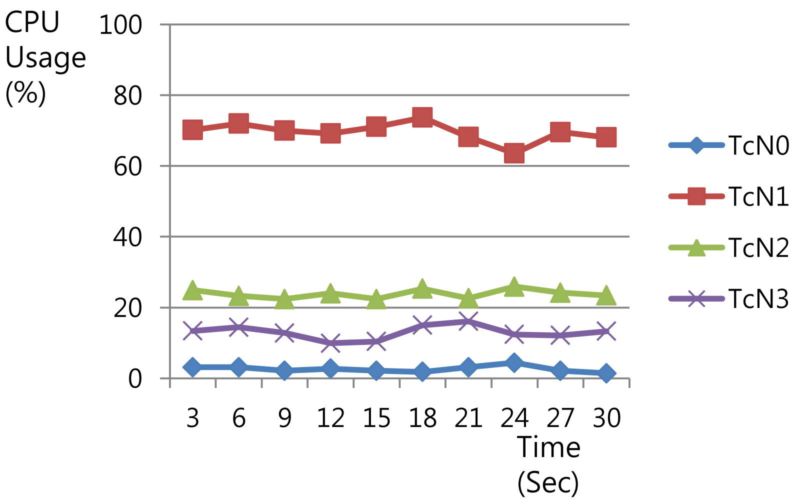


net_atk: UDP packet flooding (sending out UDP packets with the size of 44,160 bytes every 1msec)
mtd_atk: excessive NAND READ operations (scanning every directory in the filesystem and reading file contents)

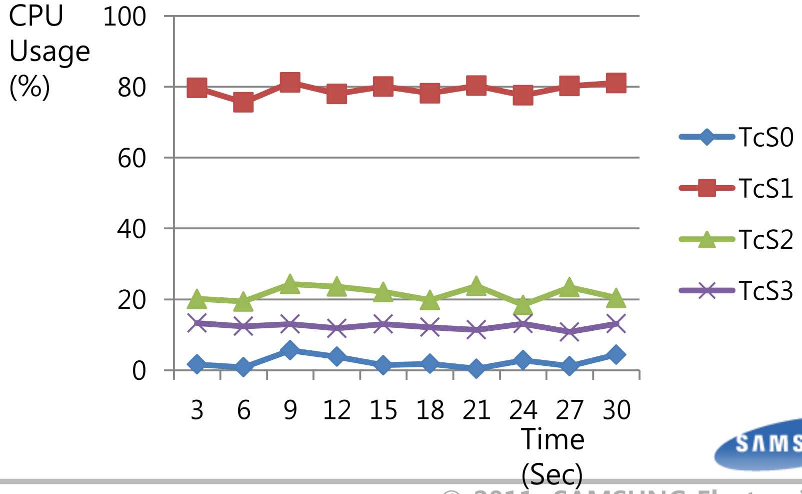
Test Cases

	Network I/O Test Cases	Storage I/O Test Cases
No Attack	TcN0	TcS0
Under Attack (No I/O ACM)	TcN1	TcS1
Under Attack (20% I/O ACM Policy)	TcN2	TcS2
Under Attack (10% I/O ACM Policy)	TcN3	TcS3

CPU Utilization: Network



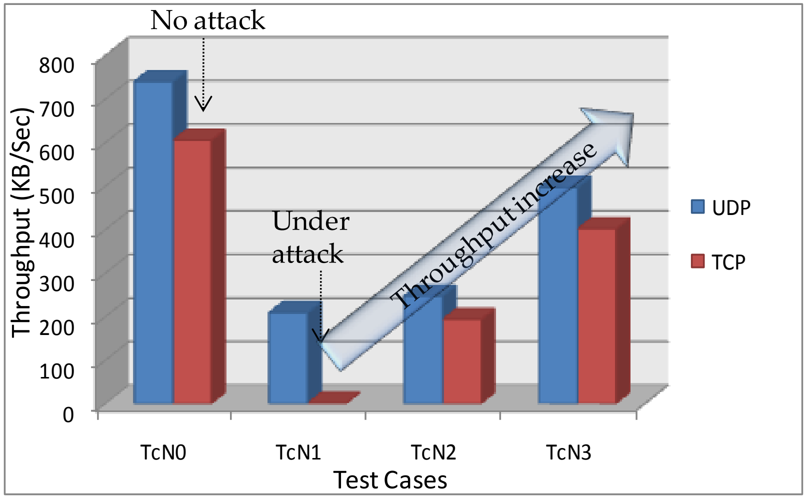
CPU Utilization: Storage



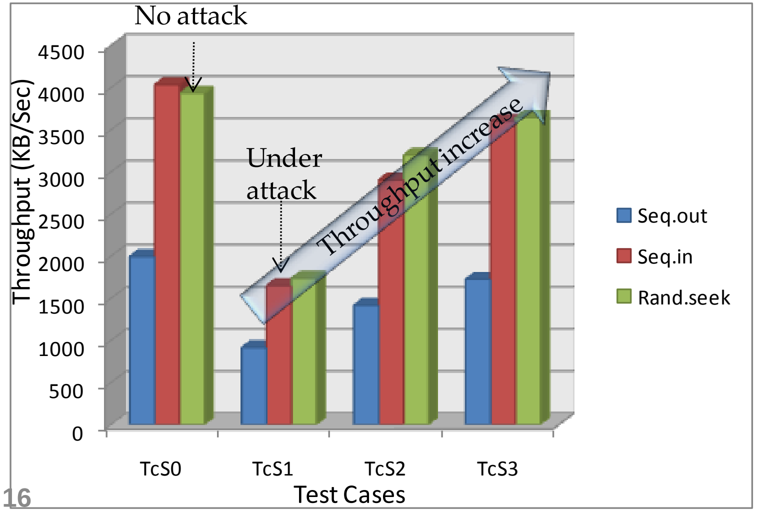
Access Control

Effectiveness

Throughput: Network

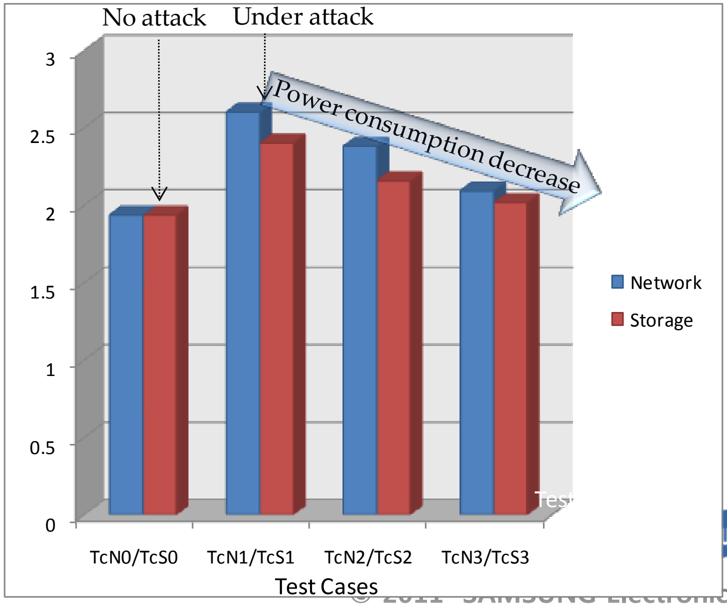


Throughput: Storage



- Throughput increase and power consumption decrease even under malware attack

Power Consumption



Xen ARM Application For Real-time

Xen ARM: Pre-emption

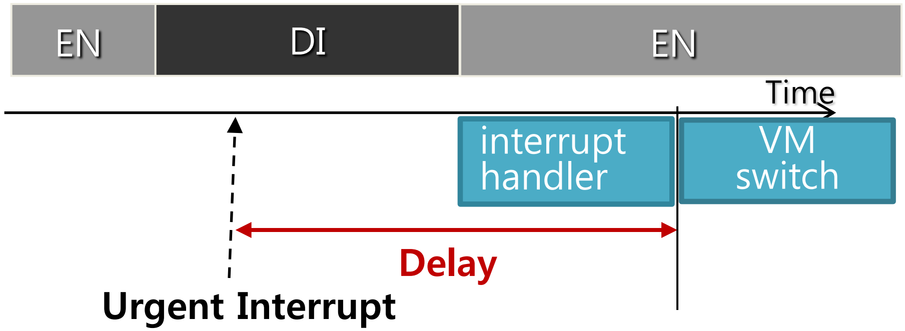
Status of real-time support

- The jitter of timer interrupt latency by the hypervisor is bounded within 10% compared with native real-time OS.

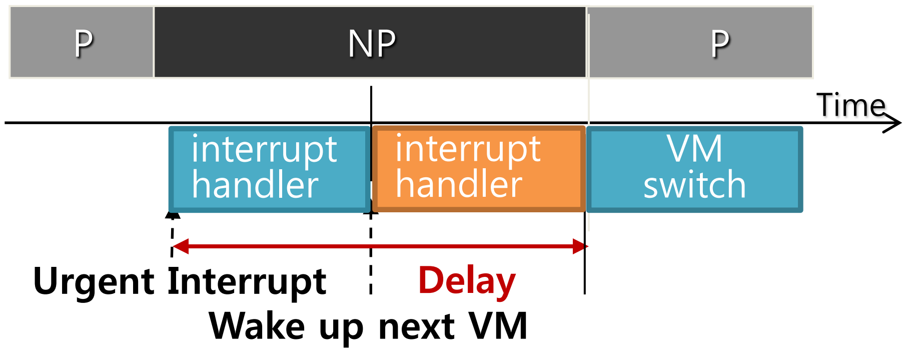
Technical Issue

- DI and NP sections should be minimized.

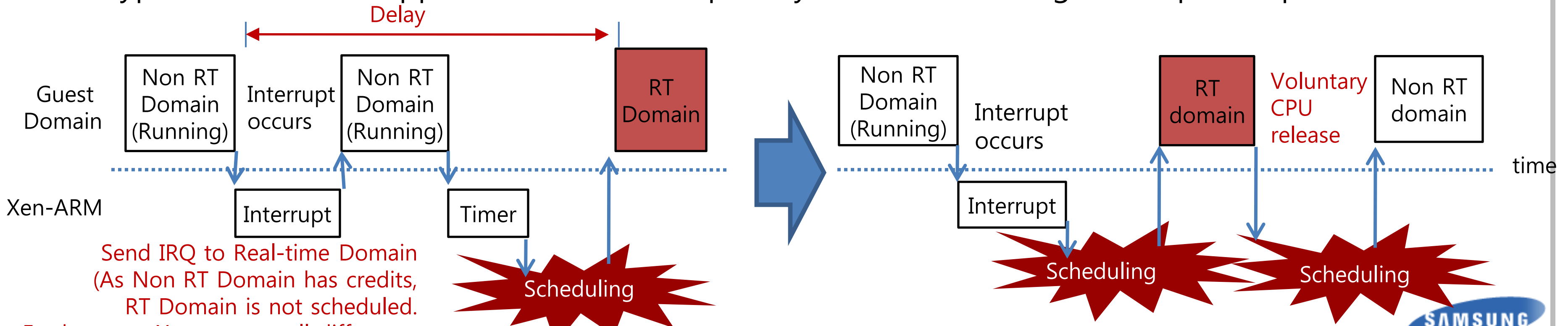
Latency caused by interrupt-disabled(DI) section



Latency caused by non-preemptible(NP) section



- Hypervisor should support RT Domain via priority-based scheduling, VMM pre-emption and so on.



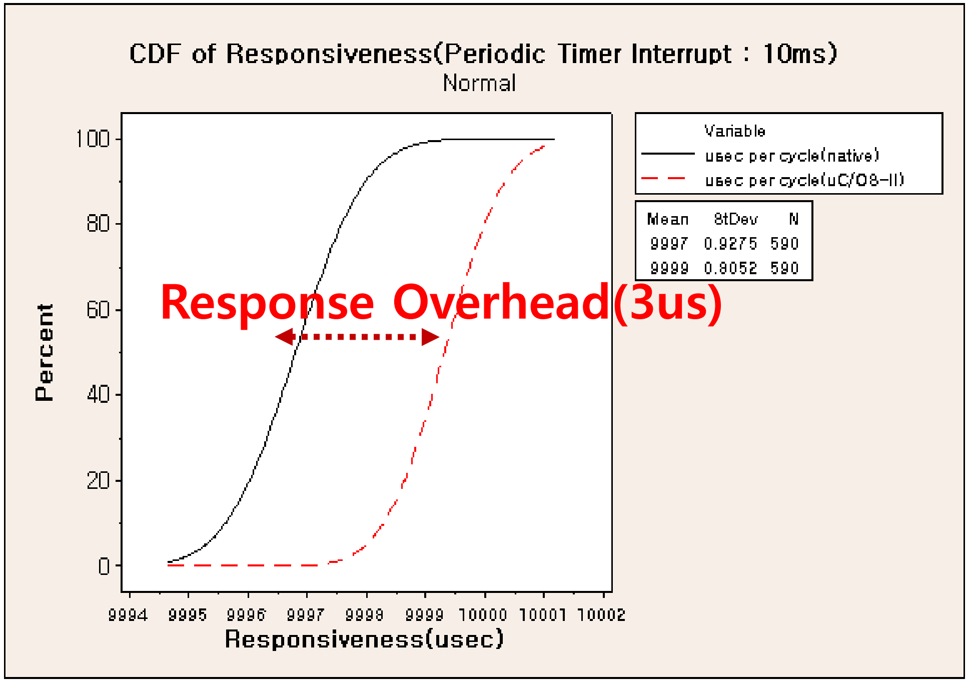
Send IRQ to Real-time Domain
(As Non RT Domain has credits,
RT Domain is not scheduled.
Furthermore Xen cannot tell difference
between Non RT Domain and RT Domain.)

Preemption

Real-time Performance

- **Evaluation Environment**

Category		Description
H/W (Tegra250)	CPU	Cortex-A9 / 1GHz / Dual Core
	RAM	1GB
S/W	Hypervisor	Xen-ARM
	Guest OS (DOM0)	Linux-2.6.29 (Running Busy Loop Task)
	Guest OS (DOM1)	uC/OS-II (Running RT Task : Cyclictest benchmark)



Response Overhead(3us)

- **Cyclictest benchmark repeats**
  1. RT task sleeps for 10ms
  2. Timer interrupt will occur after 10ms
  3. Timer interrupt wakes up the RT domain(uC/OS-II)
  4. uC/OS-II preempts Xen-ARM
  5. RT task is scheduled
  6. RT task logs timestamp

Native(uC/OS-II)		
Min	Avg	Max
9995	9996.810169	10000
Xen-ARM(uC/OS-II)		
Min	Avg	Max
9996	9999.327119	10001

Unit : usec

Q & A