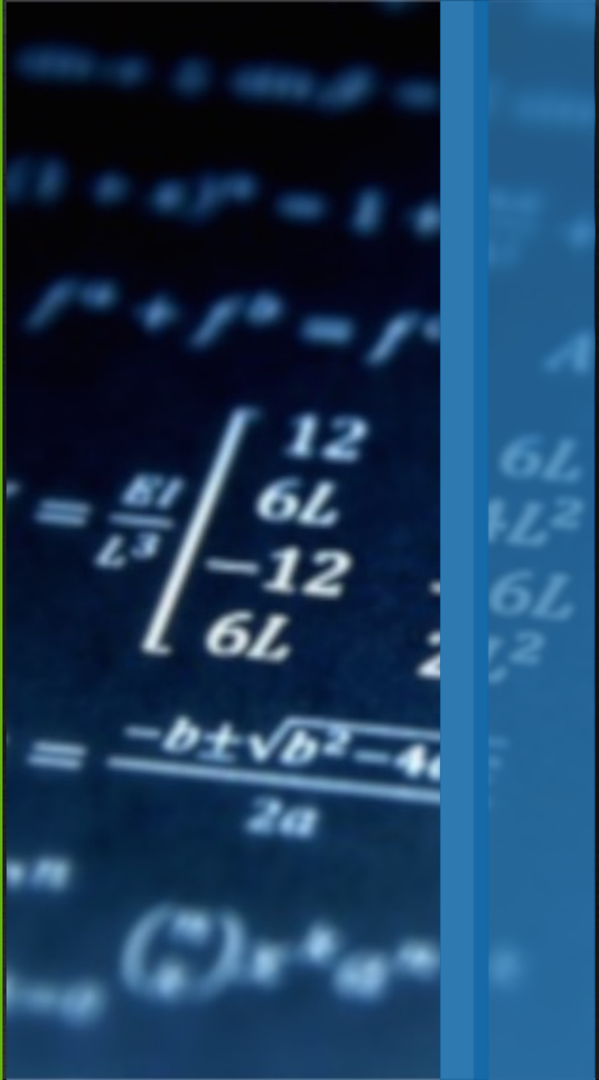
Securing Embedded Systems with the Hypervisor

Lars Kurth
Community Manager, Xen Project
Chairman, Xen Project Advisory Board
Director, Open Source, Citrix

lars_kurth
larskurth

www.slideshare.net/xen_com_mgr/presentations & xenbits.xenproject.org/people/larsk



Why Virtualize in Embedded Systems?

Consolidation

Reduce cost, size, weight and power consumption

Reduce development costs: platform independence

Security and Safety

Separate safety critical apps from general apps

Safety Certification of the Hypervisor

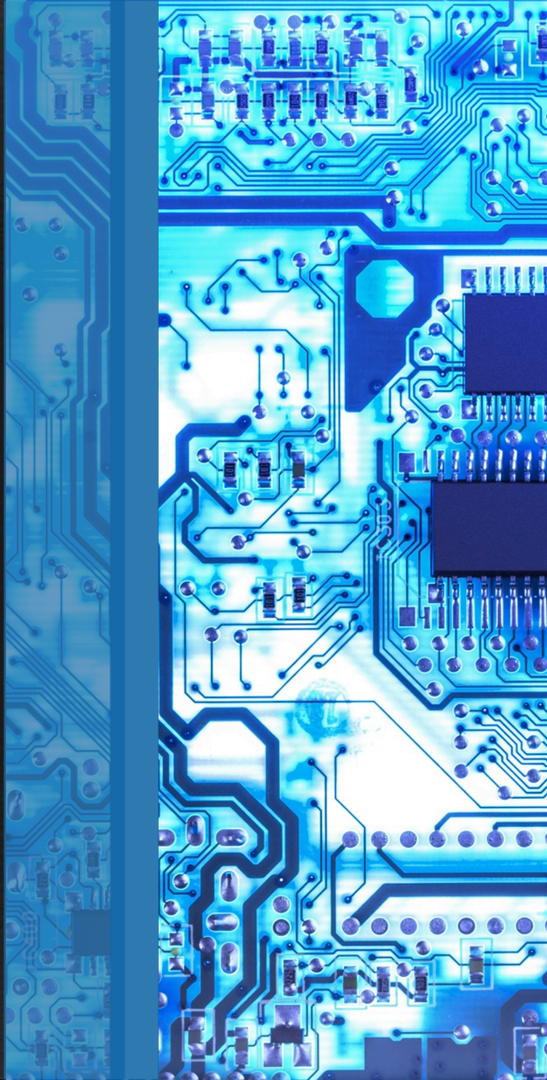
Embedded Requirements

Minimal IRQ latency

Low or 0 scheduling overhead

Drivers for special I/O devices

Flexible architecture



Hypervisor Architectures

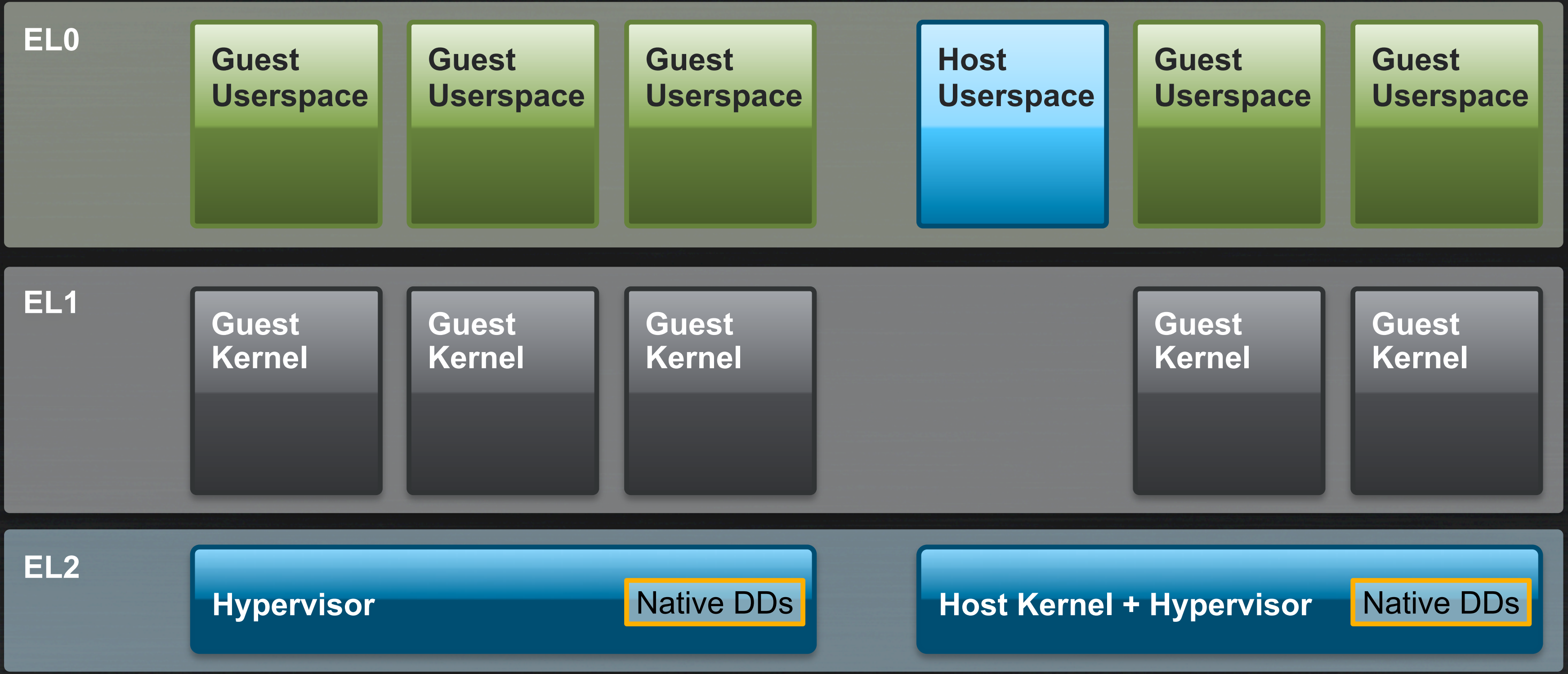
ARM Exception/Privilege Levels

EL0/PL0 least privileged mode used for applications (user mode)

EL1/PL1 privileged mode used for running kernels such as the Linux kernel

EL2/PL2 This has a higher level of privilege and can be used to run a hypervisor which takes control of the system and can host multiple "guest" operating systems

Type 1 & 2 Hypervisors on ARM

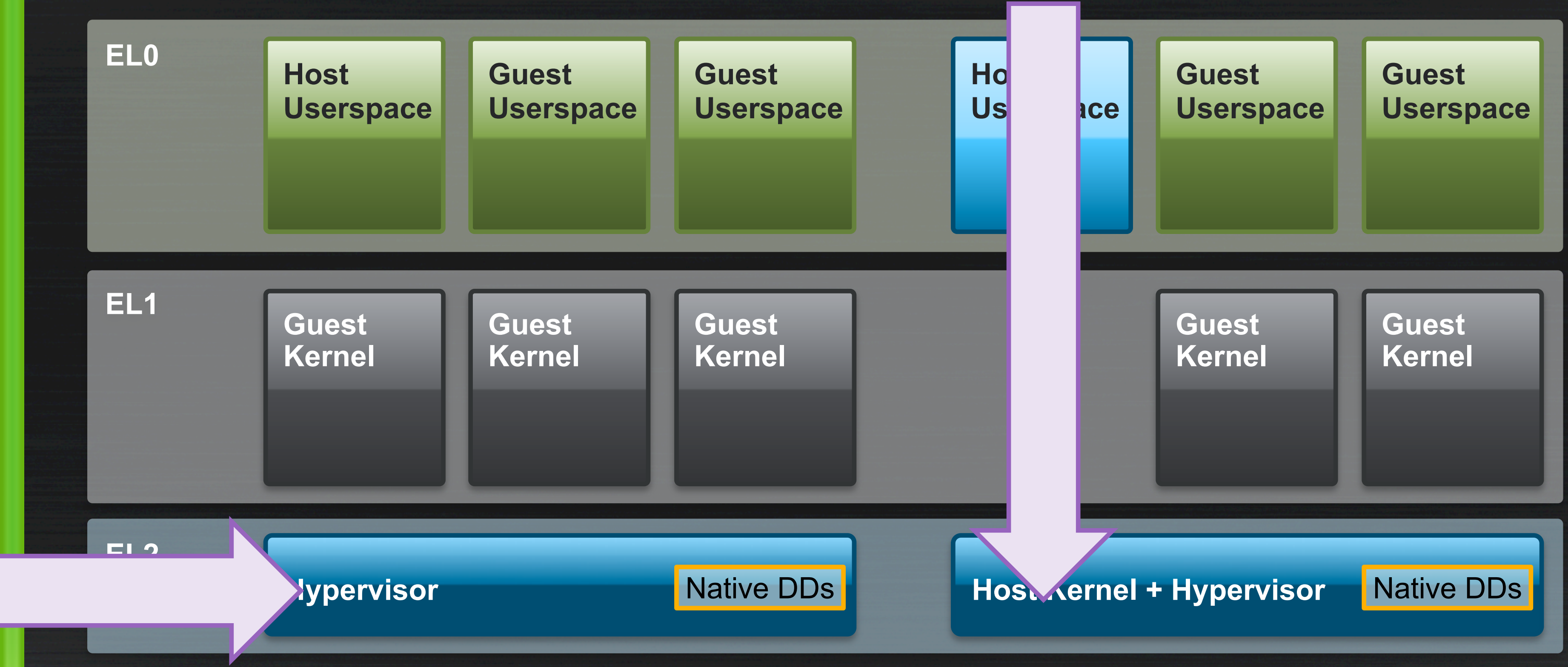


EL0	Guest Userspace	Guest Userspace	Guest Userspace	Host Userspace	Guest Userspace	Guest Userspace
EL1	Guest Kernel	Guest Kernel	Guest Kernel		Guest Kernel	Guest Kernel
EL2	Hypervisor Native DDs			Host Kernel + Hypervisor Native DDs		

Traditional Embedded Type 1 Hypervisor

Type 2 with VHE/ARMv8.1 (e.g. KVM)

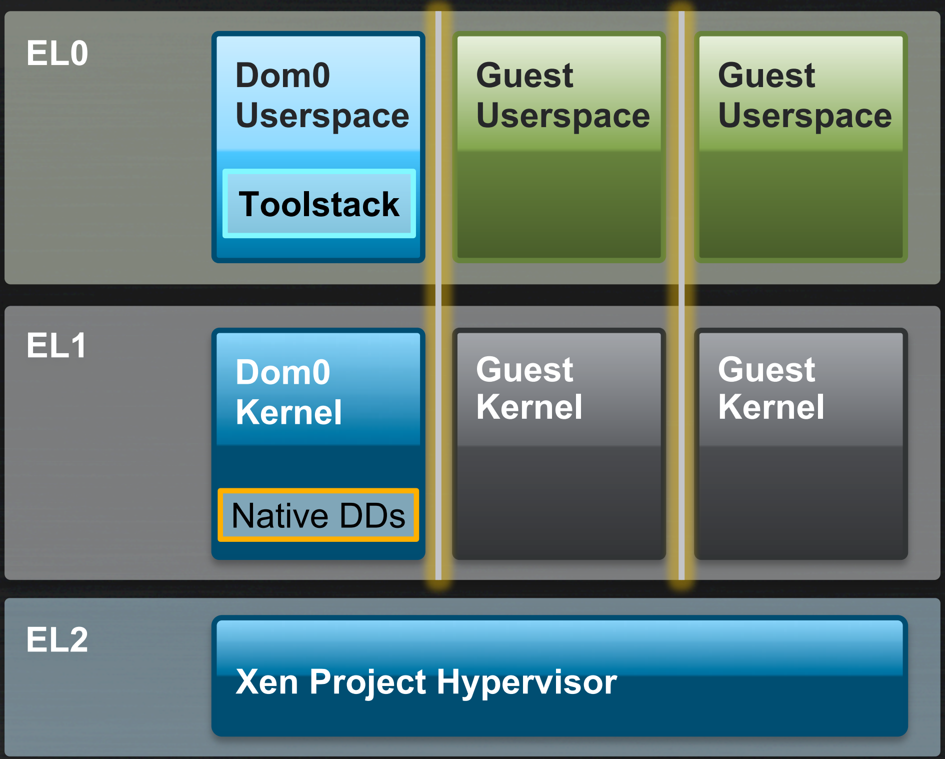
System Control Plane



Traditional Embedded Type 1 Hypervisor

Type 2 with VHE/ARMv8.1 (e.g. KVM)

Xen Project: Type 1 with a twist



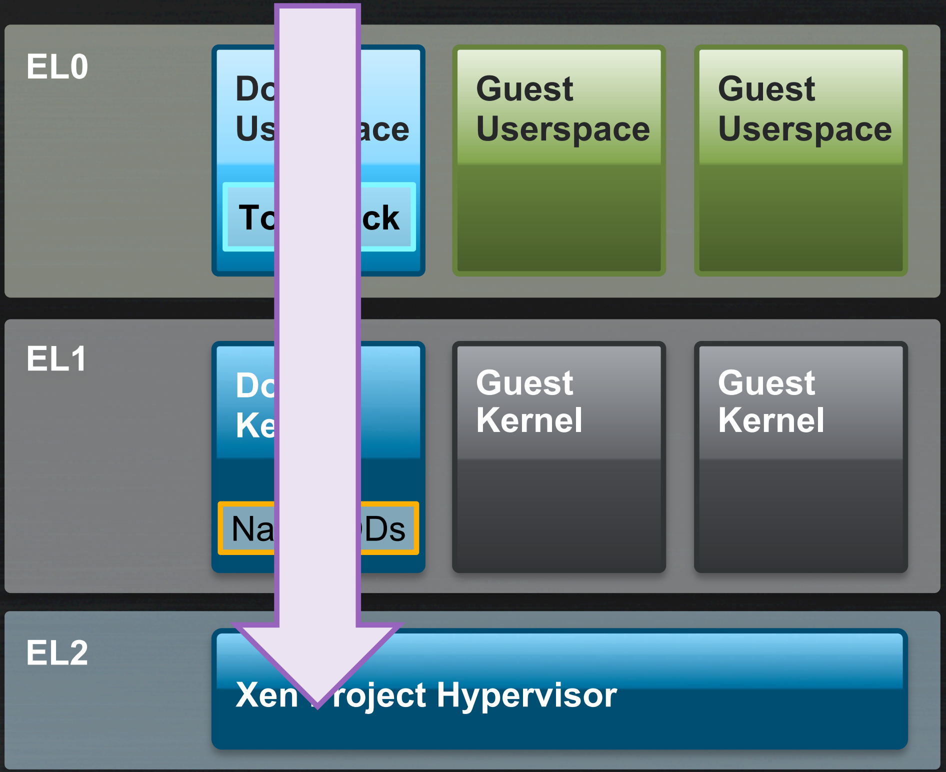
Strong Isolation

Device Drivers run in EL1, not EL2

Protected Address Spaces: *Grant tables*

Trusted Computing Base (TCB)

Xen: Type 1 with a twist



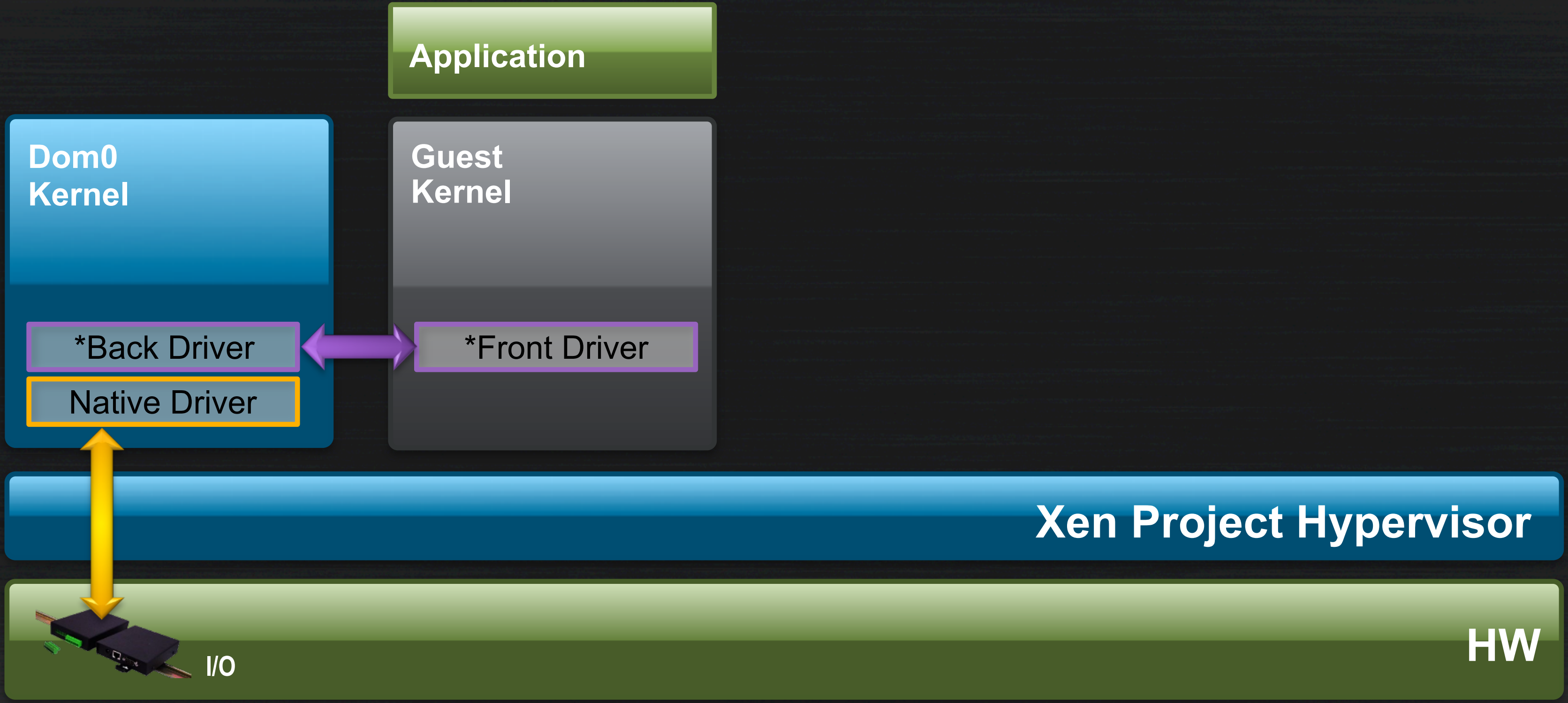
Control Plane

Server: sysadmin

Embedded: config/setup, system health monitoring (watchdog), maintenance, SW updates, …

PV Drivers and Protocols for various use-cases

PV Drivers: I/O in Xen



Existing

net, block, console
keyboard, mouse, USB
framebuffer, *GPU sharing**

New in Xen 4.9

9pfs (share a filesystem between VMs)
Pvcalls (forward POSIX calls across VMs)
multitouch, sound, display, DRM

Developing New Ones

Easy to write (GPL and BSD samples)
Kernel and User Space

*) A number of different approaches by different vendors in different market segments are being deployed, which are PV-like, but not strictly a PV protocol

Security Properties of Xen

System Partitioning

Sandboxing drivers & system components

Fine-grain control of VM capabilities

Enables multi-layered security approach

Other Security Features

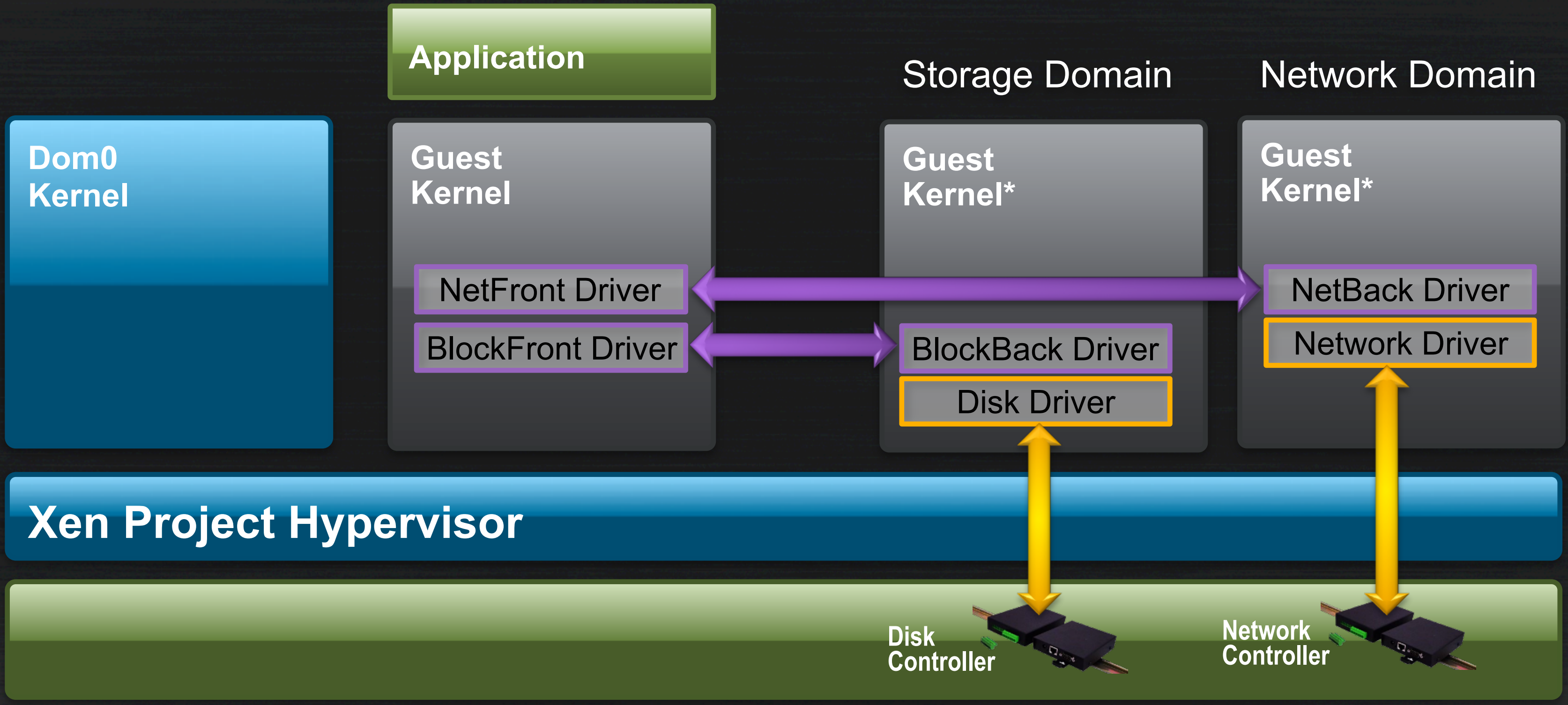
Trusted Execution Environment (TEE)

Virtual Machine Introspection, alt2pm

Live Patching

More in my talk today at 14:55
Live Patching, Virtual Machine Introspection and Vulnerability Management

Sandboxing: Disaggregation



Application

Storage Domain

Network Domain

Dom0 Kernel

Guest Kernel

NetFront Driver

BlockFront Driver

Guest Kernel*

BlockBack Driver

Disk Driver

Guest Kernel*

NetBack Driver

Network Driver

Xen Project Hypervisor

Disk Controller

Network Controller

Driver Domain Guest OS*: Linux, BSD, MiniOS, unikernel, …

XSM/FLASK Explained

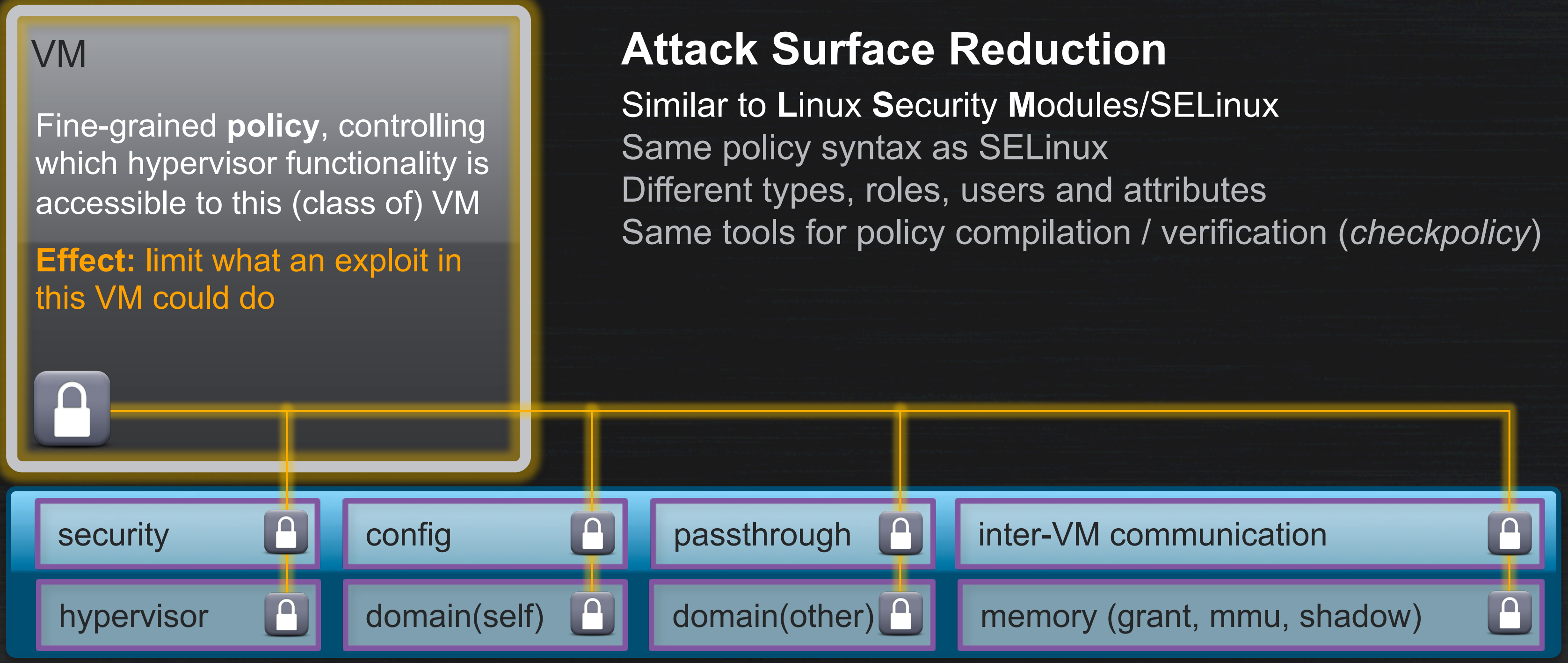
Attack Surface Reduction

Similar to **L**inux **S**ecurity **M**odules/SELinux

Same policy syntax as SELinux

Different types, roles, users and attributes

Same tools for policy compilation / verification (*checkpolicy*)



VM

Fine-grained **policy**, controlling which hypervisor functionality is accessible to this (class of) VM

Effect: limit what an exploit in this VM could do

Xen Project in Security Applications

If you want to know more …

Documentation

wiki.xenproject.org/wiki/Dom0_Disaggregation
wiki.xenproject.org/wiki/Xen_Security_Modules_:_XSM-FLASK

Products & Projects

Qubes OS

www.qubes-os.org

Secure OS

OpenXT

www.openxt.org

FOSS Platform for security research, security applications and embedded appliance integration building on Xen & OpenEmbedded

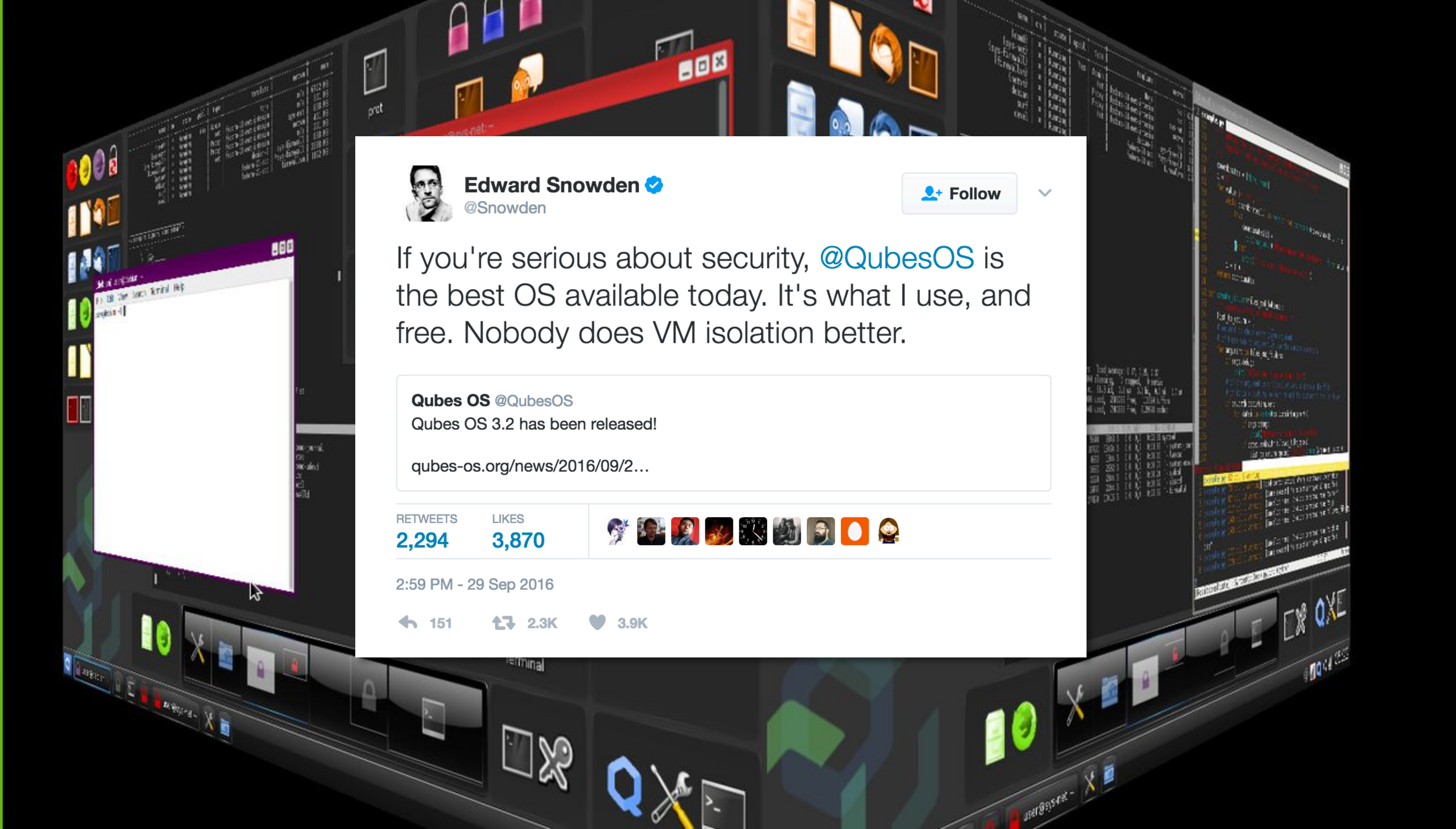
BAE SYSTEMS ais

Crucible:Defense

starlab.io

Xen Project based virtualization platform for technology protection, cyber-hardening, and system integrity for aerospace & defense systems

Pratap Sankar @ Flickr



Edward Snowden
@Snowden

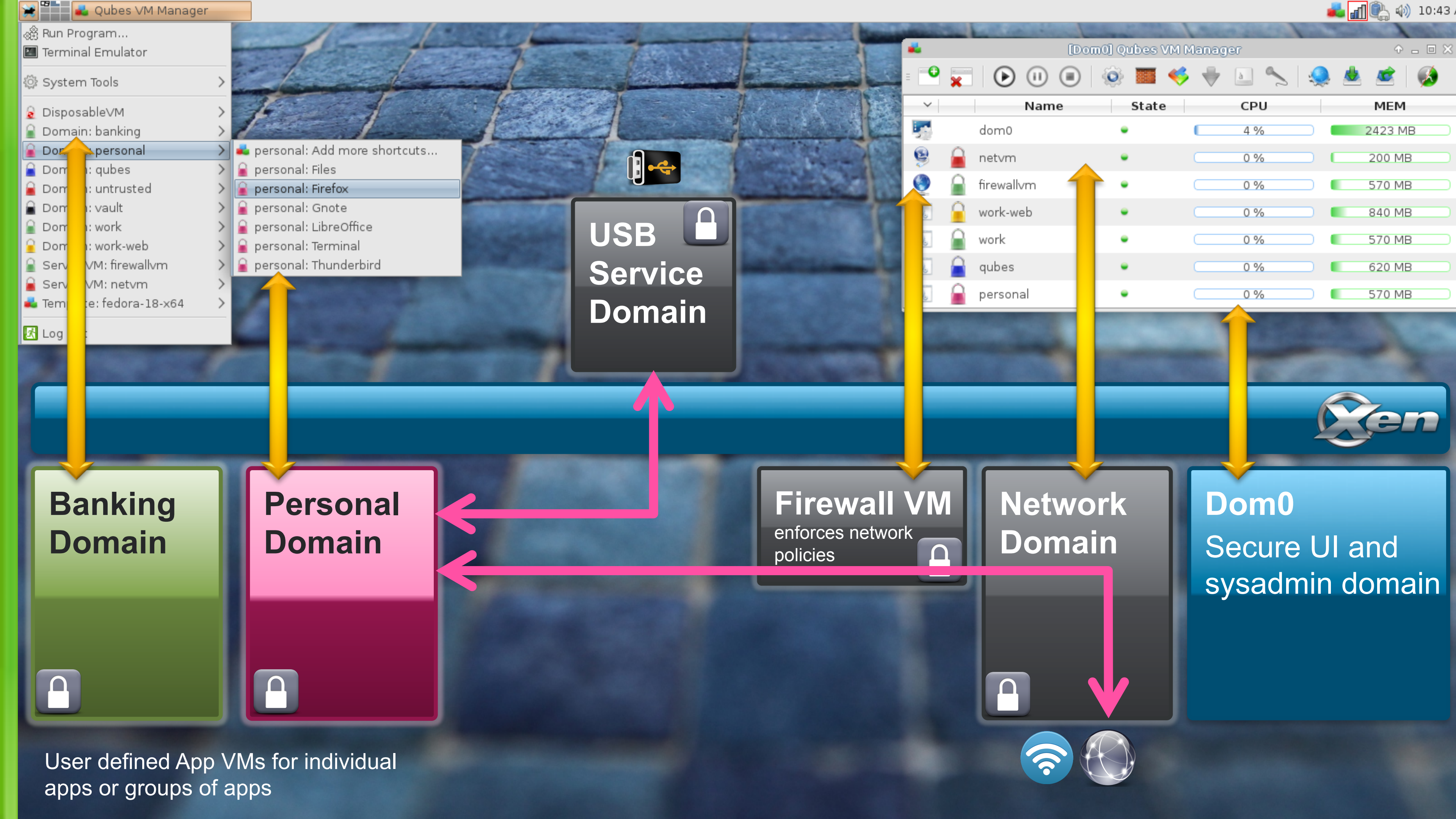
If you're serious about security, @QubesOS is the best OS available today. It's what I use, and free. Nobody does VM isolation better.

> **Qubes OS** @QubesOS
> Qubes OS 3.2 has been released!
>
> qubes-os.org/news/2016/09/2…

RETWEETS	LIKES
2,294	3,870

2:59 PM - 29 Sep 2016

151 · 2.3K · 3.9K



Name	State	CPU	MEM
dom0	●	4 %	2423 MB
netvm	●	0 %	200 MB
firewallvm	●	0 %	570 MB
work-web	●	0 %	840 MB
work	●	0 %	570 MB
qubes	●	0 %	620 MB
personal	●	0 %	570 MB

USB Service Domain

Banking Domain

Personal Domain

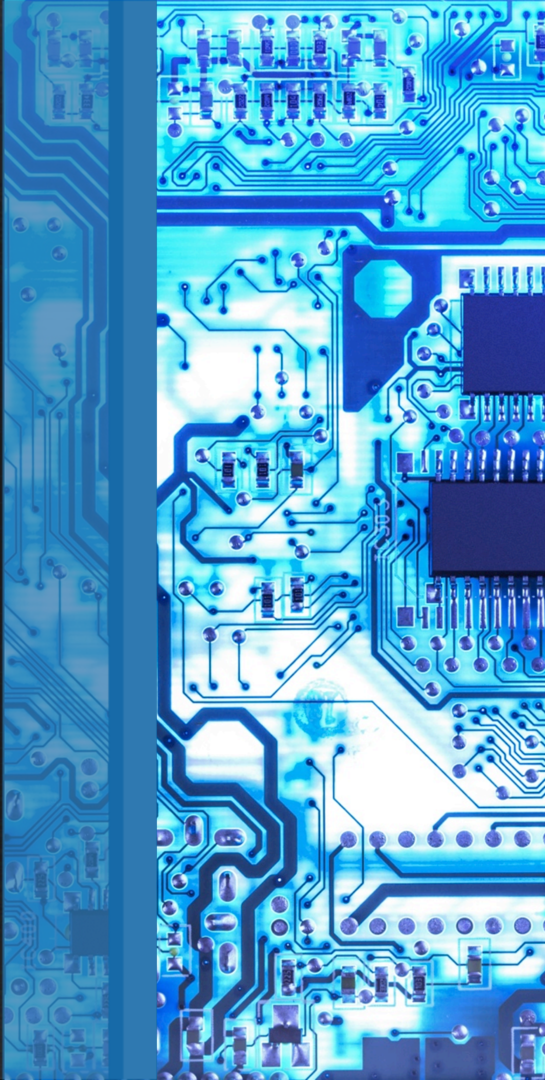
Firewall VM
enforces network policies

Network Domain

Dom0
Secure UI and sysadmin domain

User defined App VMs for individual apps or groups of apps

Xen Project in Embedded and Automotive



Embedded Vendors using Xen

Dornerworks

dornerworks.com/xen

Consulting
Xen Embedded Distros

Xen for Xilinx Zynq
Xen for NXP i.MX 8

ARLX Hypervisor
DO-178 (EAL6+), IEC 62304, ISO 26262
MILS EAL
FACE, VICTORY, ARINC 653

Starlab

starlab.io

Crucible and Crucible:Defense
Xen embedded hypervisor
In progress: DO-178, MILS EAL

Uses a minimal Dom0 using MiniOS, disaggregation and XSM/FLASK

AIS

ainfosec.com

BAE Systems

baesystems.com

Galois

galois.com

Maintain FreeRTOS Xen Port
Developed and maintain HalVM

Precedents of military grade certification for Xen based systems

www.slideshare.net/xen_com_mgr/art-certification & www.youtube.com/watch?v=UyW5ul_1ct0
xenbits.xenproject.org/people/larsk/XPDS14 - Xen and the Art of Certification.pdf
www.linux.com/news/xen-project/2017/2/how-shrink-attack-surfaces-hypervisor

Pratap Sankar @ Flickr

Automotive Vendors using Xen

GlobalLogic

Product: Nautilus

bit.do/gl-nautilus

First product in production expected in Q1 2018

Supports:

HW: Renesas R-Car Gen2 & Gen3, TI Jacinto6, Intel Apollo Lake, Qualcomm 410C, Sinlinx A33

Guests: Linux up to 4.9 • Android M, N, N-Car • QNX, ThreadX, FreeRTOS

PV Drivers for: GPU, Audio, HW accelerated Video codecs, DRM, …

Contributions:

27 smaller features from 2013 to 2016

EPAM

Demo

Next slide

Interesting Features:

Container based telematics applications running in a Xen VM that can be downloaded from a cloud service

Ongoing Contributions:

ABIs for PV Sound, PV Display & PV DRM

Leading development of co-processor sharing framework

LG Electronics

Demo

bit.do/lg-xen-demo-2016

Bosch Car GmbH

Contributions

10 smaller features in 2016

Perseus

Founded by Xen maintainer

bit.do/perseus-2017

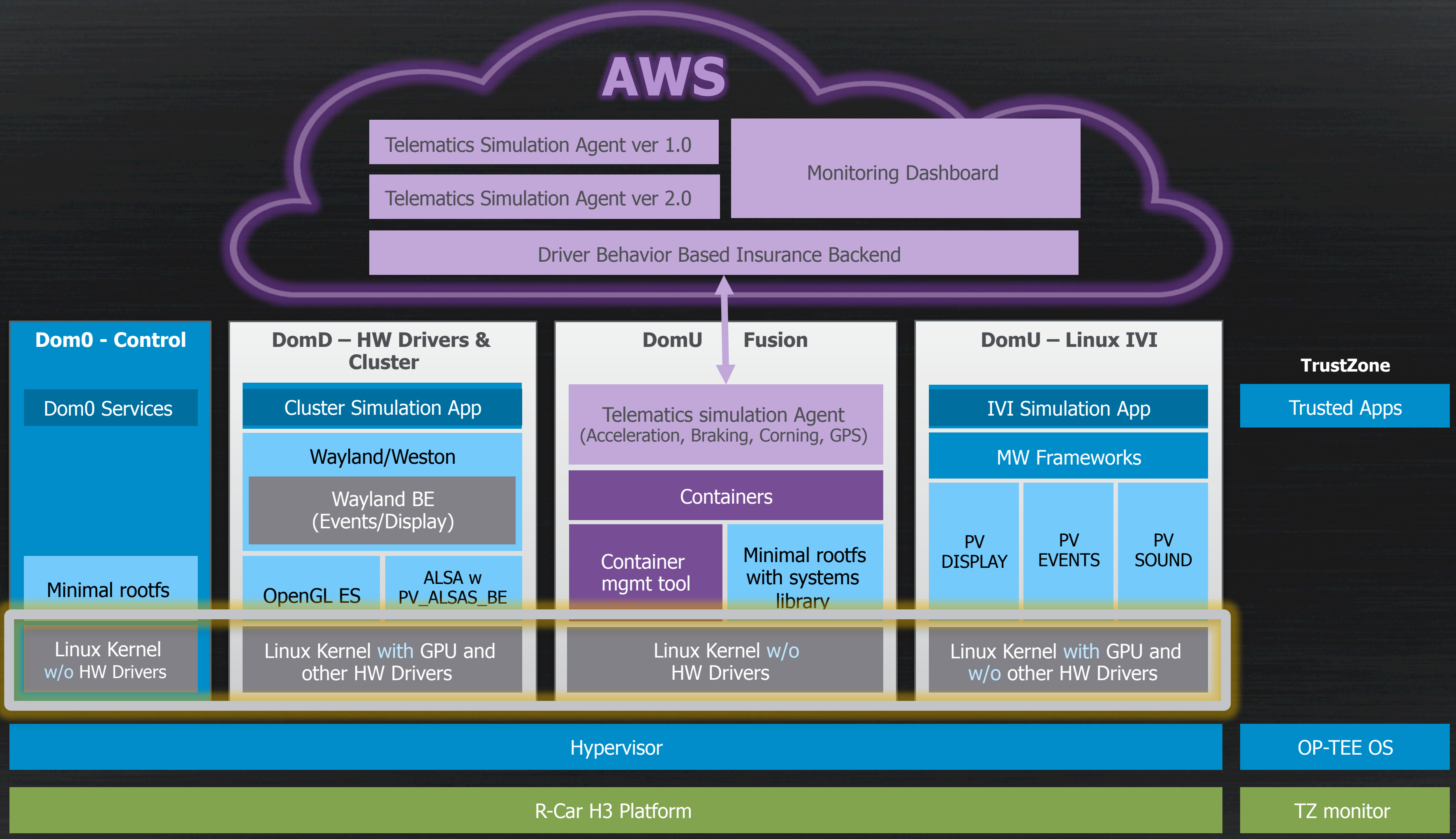
Pratap Sankar @ Flickr

EPAM Cloud Fusion Demo

xenbits.xenproject.org/people/larsk/

LCC17 - The Internet of Transportation[1080P].MP4

Pratap Sankar @ Flickr



AWS

Telematics Simulation Agent ver 1.0

Telematics Simulation Agent ver 2.0

Monitoring Dashboard

Driver Behavior Based Insurance Backend

Dom0 - Control

Dom0 Services

Minimal rootfs

Linux Kernel w/o HW Drivers

DomD – HW Drivers & Cluster

Cluster Simulation App

Wayland/Weston

Wayland BE (Events/Display)

OpenGL ES

ALSA w PV_ALSAS_BE

Linux Kernel with GPU and other HW Drivers

DomU Fusion

Telematics simulation Agent (Acceleration, Braking, Corning, GPS)

Containers

Container mgmt tool

Minimal rootfs with systems library

Linux Kernel w/o HW Drivers

DomU – Linux IVI

IVI Simulation App

MW Frameworks

PV DISPLAY

PV EVENTS

PV SOUND

Linux Kernel with GPU and w/o other HW Drivers

TrustZone

Trusted Apps

Hypervisor

OP-TEE OS

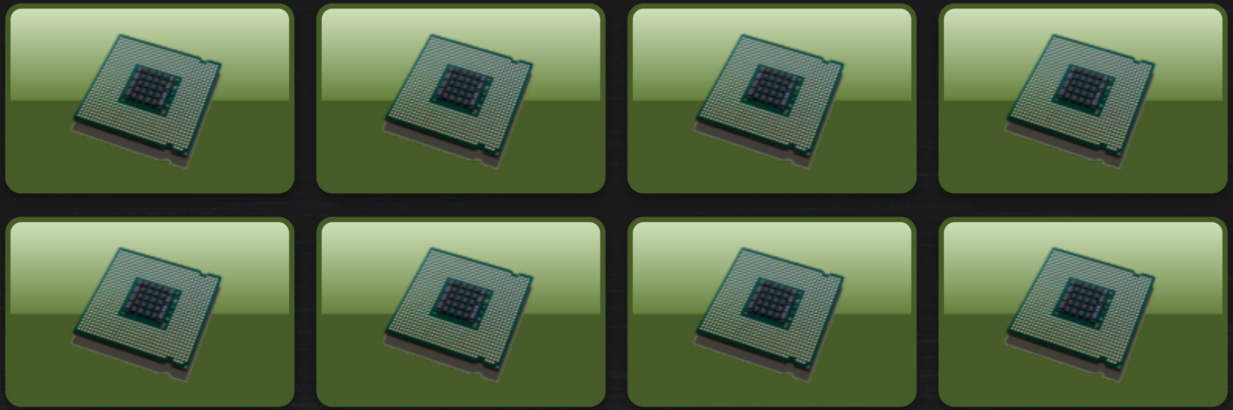
R-Car H3 Platform

TZ monitor

Schedulers & Interrupt Latency

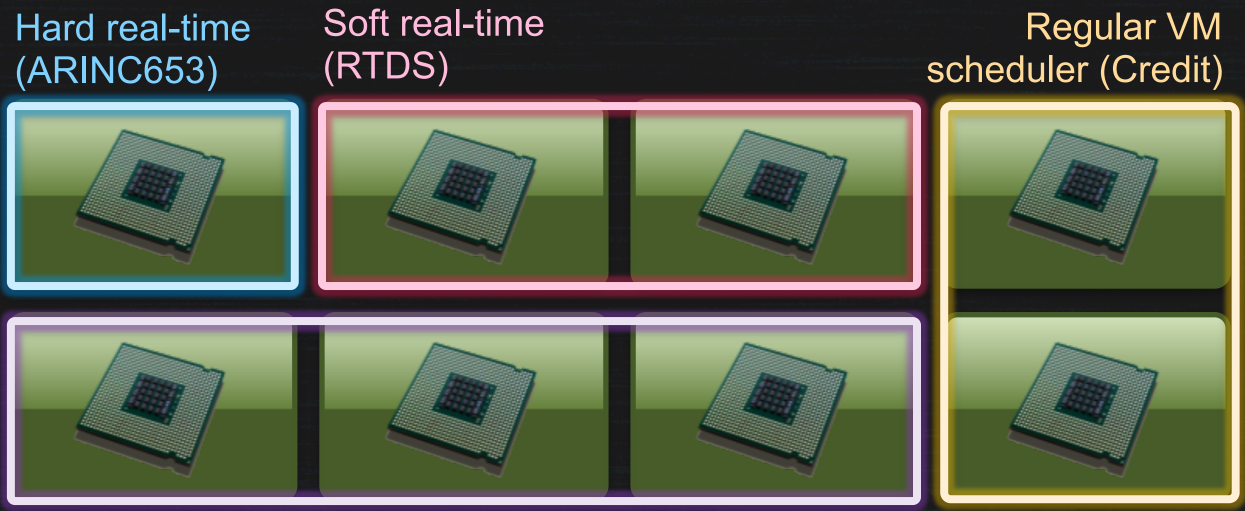
Partitioning the System

Xen supports **several different** schedulers with different properties.



Partitioning the System

Xen supports **several different** schedulers with different properties.



Hard real-time (ARINC653)

Soft real-time (RTDS)

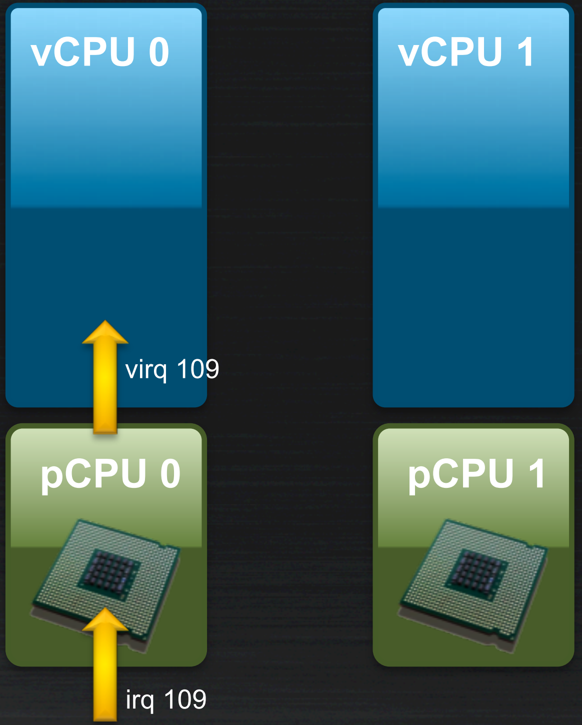
Regular VM scheduler (Credit)

Dedicated to 1 VCPU via pinning and Null scheduler
➔ no scheduler overheads

Xen Schedulers

Scheduler	Use-cases	Today	Future plans
Credit	General Purpose	**Supported** **Default**	Supported Optional
Credit 2	General Purpose Optimized for lower latency, higher VM density	**Supported**	**Default**
RTDS	Soft & Firm Real-time **Multicore** Embedded, Automotive, Graphics & Gaming in the Cloud, Low Latency Workloads	Experimental Better XL support <1µs granularity	Supported Hardening Optimization
ARINC 653	Hard Real-time **Single core** Avionics, Drones, Medical	**Supported** Compile time	
Null	Hard Real-time	Experimental	Supported

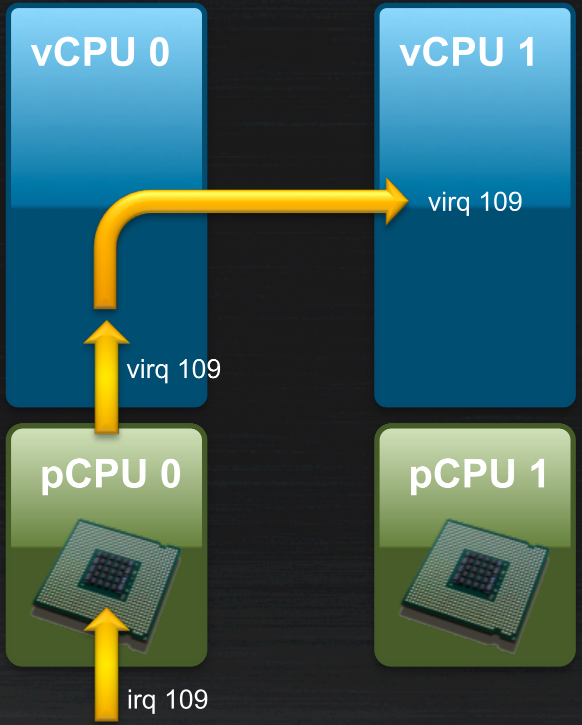
IRQs: Physical follows virtual



IRQ injection

Always on the CPU running the vCPU

IRQs: Physical follows virtual



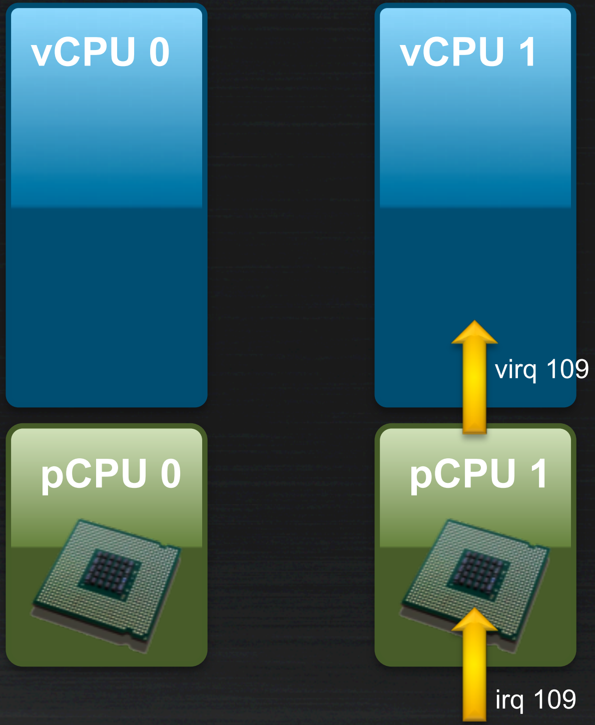
IF

vIRQ target changes or vCPU is moved

THEN

vIRQ is moved immediately

IRQs: Physical follows virtual



Xilinx ZynqMP board
(four Cortex A53 cores, GICv2)

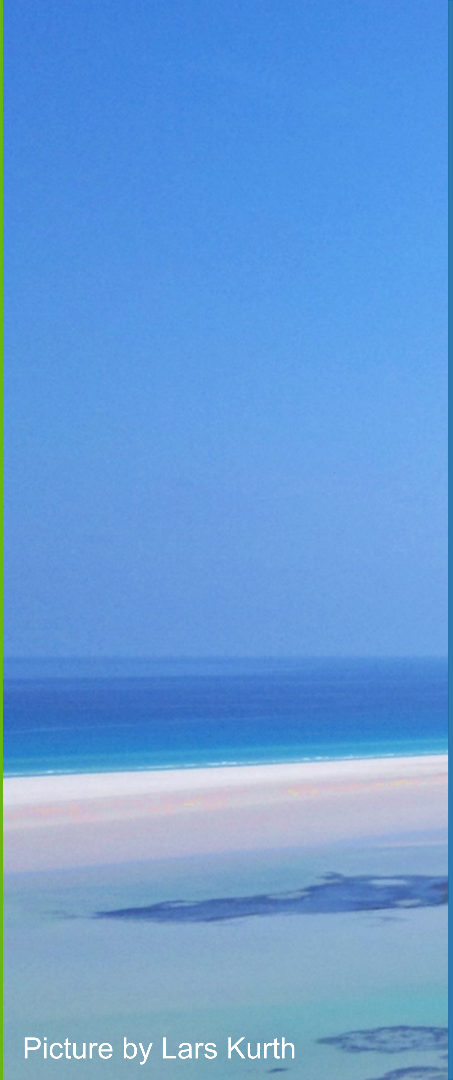
WARM_MAX (excluding the first 3 interrupts): <2000ns

Without Null scheduler
See blog.xenproject.org/2017/03/20/xen-on-arm-interrupt-latency/

IRQs always shadow the vIRQ

➔ minimizes latency

Why should I use Xen?



Picture by Lars Kurth

Extremely Flexible and Versatile

Proven in many different markets

Easy to port to new environments

Easy to develop new PV drivers

Highly customizable

Security and Resilience

Isolation, Partitioning, Security Features

Safety

Examples of Military Grade Certification

BUT: looking at ways to make this easier and cheaper

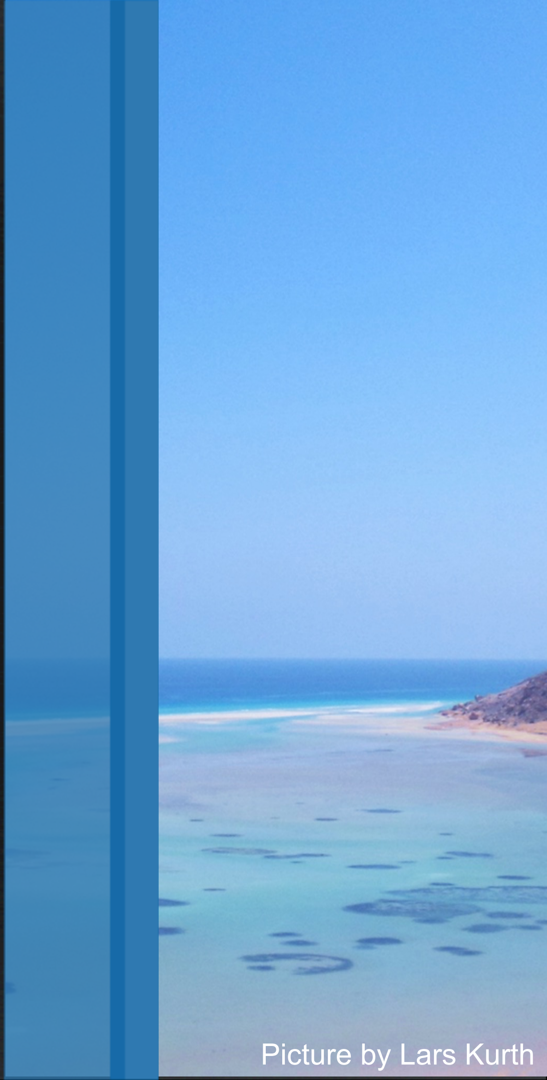
Challenges still being addressed

Standardization of more I/O devices via PV protocols

Standardization of GPU and co-processor sharing

RTOS or other minimal OS as Dom0

Testing of embedded Hardware by the project



Picture by Lars Kurth

Questions

xenbits.xenproject.org/people/larsk

Picture by Lars Kurth

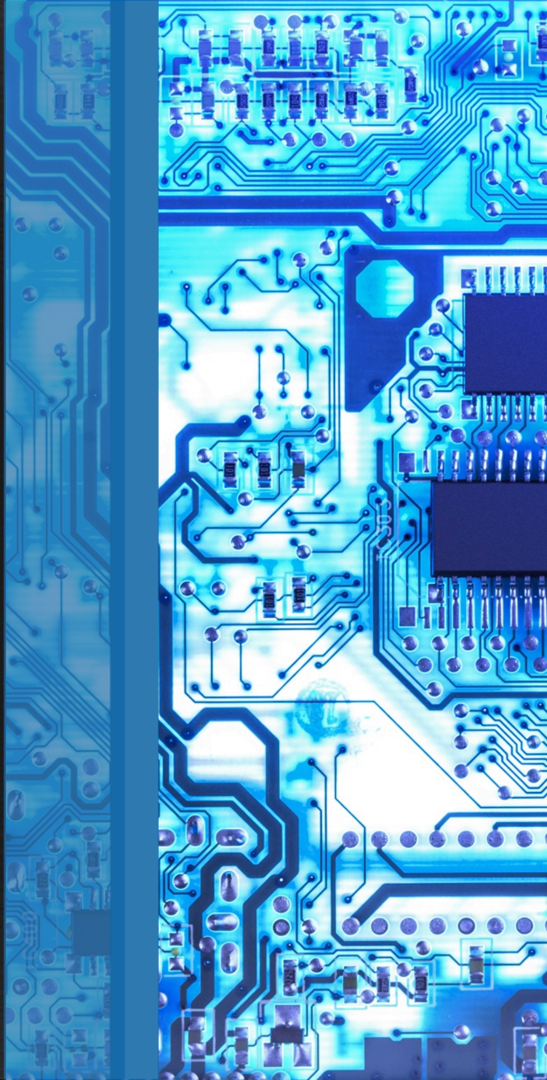
More Resources

Developer Portal: bit.do/xen-devs
Xen on ARM whitepaper: bit.do/xenarm-white
Xen on ARM wiki: bit.do/xenarm-wiki

Port Xen to a new SOC: bit.do/xenarm-porting
Add Xen support Xen to your OS: bit.do/xenarm-os

Device Passthrough presentation: bit.do/xenarm-pt
OE meta-virtualization Xen recipe: bit.do/xenmeta
OpenXT (Xen + OpenEmbedded): openxt.org
Xenbedded presentation: bit.do/xenbedded

Monthly ARM Community Call: bit.do/xenarm-call



Engage!

Lists and IRC on freenode:
xen-devel@lists.xenproject.org
xen-users@lists.xenproject.org
#xenarm or #xen-devel

Xen Project Developer and Design Summit:
July 11-13, Budapest, Hungary