「Xen Developer Summit」の最新情報のご紹介

基調講演 @ OSAKA NDS Embedded Linux Cross e-Forum No.11

宗像尚郎

Automotive Grade Linux Advisory board member
Linux Foundation board member

2020-7-10

自己紹介

ルネサスエレクトロニクス という半導体メーカーで働いています

- ルネサスの概要と私の担当業務
  - ルネサスエレクトロニクス＝（日立＋三菱）＋ NEC の半導体事業部門
  - 車載向け SOC（System on Chip) のグローバルリーディングサプライヤーです
  - R-CarH3 = 64bit ARM Octa-core + GPU + LPDDR4 8G、1,384pin BGA
  - 産業コンソーシアム活動、コミュニティ活動でも指導的な役割をもっています
  - Linux Foundation のボードメンバー です
  - Automotive Grade Linux、yocto project の理事も務めています
  - 社内オープンソース開発チームを運営、コミュニティ開発でも顕著な実績
- Xen FUSA SIG メンバーとして、Xen の車載分野への適用 を検討しています

ルネサスはオープンソース開発活動への貢献実績において日本を代表する企業です

Xen Hypervisor とはどういうものか

Xen VMM （=hypervisor）の特徴

Xen ＝サーバー向け仮想化ソリューション ではデファクト

- Linux Foundation に帰属する オープンソースプロジェクト (GPL v2)
- サーバー向けには 2003 年から製品適用 されている（歴史は古く枯れている）
  - 0.38/1000 LOC と 極めて低い欠陥密度（=defect density)
  - 仮想マシンの挙動を監視する VMI(=Virtual Machine Introspection) にも対応
- 最新の (= 仮想化支援機構をもった) x86 と ARM アーキテクチャ に対応
  - OP-TEE や Linux kernel でもデフォルトで Xen がサポートされている
- ホスト OS を必要としない type1 (ベアメタル) ハイパーバイザー
  - さまざまな仮想化実装モデル (準仮想化、完全仮想化) に対応
  - CPU 内蔵ハードウエアを利用して 強固なセキュリティメカニズム も実現可能

Xen は Enterprise Grade Hypervisor として実績があり ARM CPU でも動作する

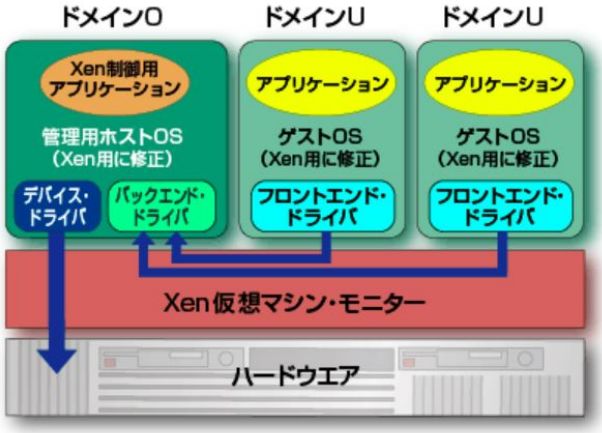
Xen が 準仮装化（Para Virtualization) による性能改善を実現！

仮装マシン上で OS を動かす には複数のやり方がある

- 完全仮装化
  - 既存の OS（Windows、Linux など）を そのまま仮装マシン上で動作させる事が可能
  - ソフトウエアエミュレーションは オーバーヘッドが性能面でのペナルティ
  - CPU の仮装化支援（Intel VT、AMD HyperV、ARM hypervisor extension） が利用できる場合には HVM（= Hardware Virtual Machine) モードで動作可能
- 準仮装化（PV）···· Xen のデフォルト
  - 性能ペナルティ解決のためハイパーバーザー経由でハードウエア資源を利用
  - 仮想マシン上で動かす OS を改造してハイパーバイザコールを発行 させる
  - Linux kernel は 2.6.23（2007 年）以降、標準で Xen mode をサポートしている

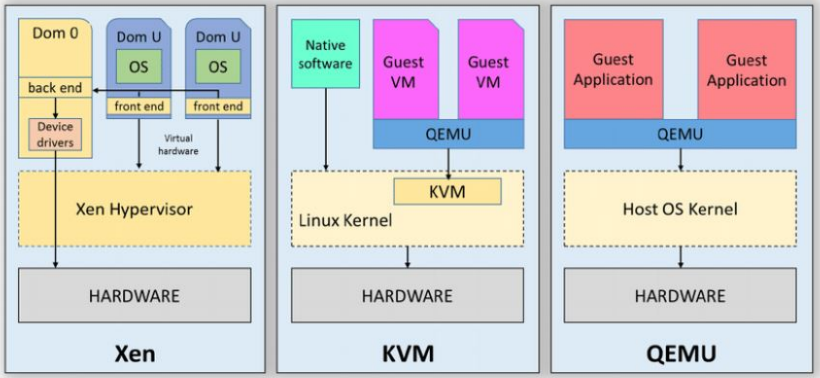
Xen は **CPU** の仮装化支援が無い時代に開発されたが、現在は完全仮装化にも対応済

Xen アーキテクチャ （Para Virtualization を利用を利用した場合）



https://xtech.nikkei.com/it/article/COLUMN/20060807/245320/

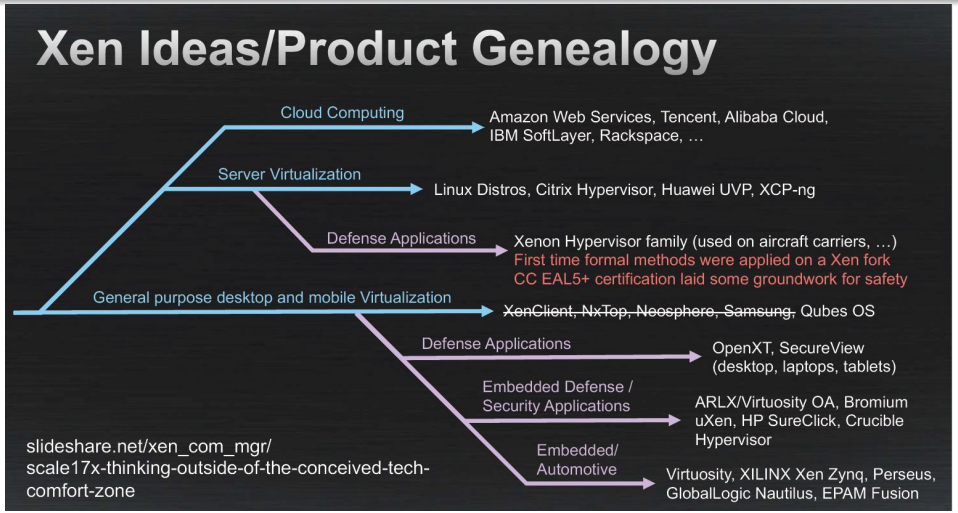
Xen、KVM、QEMU の仮想化実現方法の違い



Comparison of Xen, KVM, and QEMU.

https://www.researchgate.net/publication/281177318_Hardware_and_Software_Aspects_of_VM-Ba

Xen hypervisor の 採用分野（サーバー用途だけではない）



AWS は KVM サポートを開始した

AWS Storage Gateway が Linux KVM ハイパーバイザで使用可能に

投稿日: Feb 4, 2020

AWS Storage Gateway サービスでは、すべてのゲートウェイタイプを対象としたデプロイオプションの 1 つとして Linux Kernel-based Virtual Machine (KVM) ハイパーバイザを使用できることになりました。KVM ハイパーバイザベースのオンプレミスインフラストラクチャをご使用の場合は、貴社環境に Storage Gateway をデプロイすると、クラウドストレージを仮想化して無制限にアクセスできるようになります。

オープンソースの仮想化テクノロジーである KVM を使用すると、Linux ベースのホストマシンを 1 個または複数の仮想マシンに転換できます。この新機能で、Storage Gateway は KVM ハイパーバイザにデプロイできることになりました。Storage Gateway では、Red Hat Enterprise Linux 7.7、CentOS 7.7、Ubuntu 18.04 LTS、Ubuntu 16.04 LTS、および仮想化拡張機能を搭載した Intel x86 CPU を使用するホストマシン上の QEMU-KVM がサポートされています。

ハイブリッドクラウドストレージサービスである Storage Gateway では、NFS、SMB、iSCSI、iSCSI-VTL の各インターフェイスを使用することにより、オンプレミスのアプリケーションが実質的に無制限のクラウドストレージにアクセスできるようになります。本サービスを使用すると、データを AWS にバックアップおよびアーカイブし、オンプレミスのストレージをクラウドベースのファイル共有に移行できます。また、オンプレミスのアプリケーションから AWS 内のデータに低レイテンシーでアクセスできるようになります。Storage Gateway はオンプレミスの仮想アプライアンス (VMware ESXi、Microsoft Hyper-V、Linux KVM)、オンプレミスのハードウェアアプライアンス、AWS の Amazon EC2 インスタンスにもデプロイ可能です。

https://aws.amazon.com/jp/about-aws/whats-new/2020/02/aws-storage-gateway-available-linu

但し、AWS は Xen をやめて KVM に全面移行する... は誤解です

Amazon EC2 よくある質問

Q.AWS では、引き続き Xen ベースのハイパーバイザーに投資しますか?

はい。AWS がグローバルなクラウドインフラストラクチャを拡張するにつれ、EC2 での Xen ベースのハイパーバイザーの使用も引き続き増加します。Xen は、これからも、予測可能な将来を実現するための EC2 インスタンスのコアコンポーネントとして機能します。AWS は、Xen Project が Linux Foundation Collaborative Project として開始されたときから参加している創設メンバーであり、その Advisory Board に現在も積極的に参加しています。AWS がグローバルなクラウドインフラストラクチャを拡張するにつれ、EC2 の Xen ベースのハイパーバイザーも引き続き発展します。そのため、EC2 での Xen への投資は増え続け、縮小されません。

https://aws.amazon.com/jp/ec2/faqs/

Xen の 製品適用時のサポート体制 （但し Enterprise 用途向け）

クラウド、ネットワークおよび デスクトップ仮想化に最適化された 仮想化プラットフォームを実現

Citrix XenServerは、クラウド、ネットワークおよびデスクトップ仮想化等のために最適化された商用のハイパーバイザーです。オープンソースのXenServer やXenハイパーバイザーをベースに開発されています。

x86サーバー環境に対してWindowsやLinuxの仮想マシンを構築し、あらゆるリソースの一元的管理・運用を、容易かつスピーディに行うことができます。豊富な管理セットと自動化機能、高可用性機能、優れたアーキテクチャによる、データセンターに集中するワークロードの高い集約率とパフォーマンス、および自動化による管理性で最高の仮想化環境を提供します。これにより企業システムの柔軟性と迅速性、コスト効率を高めます。また、XenServerは、シンプルなライセンスモデルにより容易な導入を実現します。企業やサービスプロバイダーがクラウドを構築する中で、Xen / XenServerは、数多くのクラウドおよびXenDesktop, XenAppのデスクトップ仮想化基盤に提供されており、多くの導入実績があります。オープンソースXenServerはxenserver.orgよりダウンロード可能です。

XenServerの特長

基本機能
- オープンソースベース
- 64 ビット対応によるスケーラビリティ
- 最新ゲスト OS 対応
- クラウド OS との連携

優れた管理性
- 管理サーバー不要
- 直感的で使いやすい管理ツール
- ワークロードバランシング

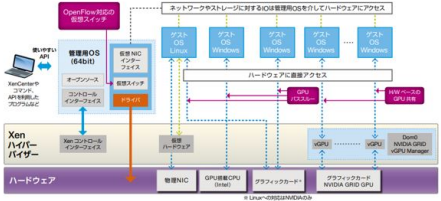
最高のパフォーマンス
- メモリおよびディスクのキャッシュ技術
- vGPU への対応と処理性能
- XenDesktop/XenApp の性能改善

クラウド時代のオープンソースベースの ハイパーバイザー

Citrix XenServer

最高の管理性と可用性、豊富な機能で
クラウド時代のシステム基盤を完全に最適化

XenServer 概要



Xen project active contributors

ARM は Xen project の正式サポートアーキテクチャ です

Xen ARM with Virtualization Extensions

For the ARM port supporting paravirtualized guests on processors without the virtualization extensions see Xen ARM (PV).

The ARM v7-A and ARM v8-A architectures include optional virtualization extensions that allow a hypervisor to manage fully hardware virtualized guests. These extensions are currently available in some ARM v7 processors such as the Cortex A15 and Cortex A7.

Hardware

CORE/SOC/BOARD	XEN GUIDE	NOTES
ARM Cortex A7/A15		
ARM Cortex A7/A15 Real-time System Model (FVP)	Fastmodels	Commercial emulator
Versatile Express	Versatile Express	With TC2 daughterboard
Calxeda EXC-2000	Midway	
Allwinner sunxi		
sun7i/A20	Allwinner	linux-sunxi community, e.g. Cubietruck
sun6i/A31		linux-sunxi community
Exynos5xxx		
Exynos5250	Arndale	www.arndaleboard.org
Exynos5410	OdroidXU	www.hardkernel.com
OMAP5		
OMAP5432	uEVM	www.ti.com
Renesas R-Car H2/H3		
Renesas R-Car H2	Lager	
Renesas R-Car H2	Stout	
Renesas R-Car H3	Salvator-X	

https://wiki.xenproject.org/wiki/Xen_ARM_with_Virtualization_Extensions

Xen ARM のサイズはリファクタリングにより x86 より相当小さい

ARM

Full ARM 64 and 32 bit, with everything enabled.

Components	K SLOC
/xen/common	33.4
/xen/arch/arm	19.8
/xen/drivers	16.0
Total	**69.3**

Xen on ARM64 with ACPI (used in servers) and ARM32 disabled is ~60K SLOC today.

Future:

A minimal Xen configuration for a small set of boards should be in the order of 40K to 50K SLOC, smaller if common code can be aggressively removed via Kconfig.

X86

On x86 Xen, there is little configurability today, but

Calculation	K SLOC
x86 with everything enabled	325
x86 PVH for Intel only, no server features	128

However, the 128K SLOC figure includes most Intel SKUs. Focusing on the latest hardware only should reduce this significantly.

Cost Example: DO-178C, 45K SLOC
DAL E (0.11 h/SLOC): ~2.4man years ··· ASIL-A
DAL C (0.20 h/SLOC): ~ 4.5man years ··· ASIL-B/C
DAL A (0.67 h/SLOC): ~15man years ··· ASIL-D
Hours for vendor with certification experience

Perspective: Total Xen Community Dev Effort
2014 - 2017: ~41 to ~50man years per year
Using conservative COCOMO model

Xen ARM の車載制御用途への対応状況

車載制御 (IVI, Gateway 用途) への仮想化導入時の課題

仮想化の実現手段に関わらず 以下の技術要件へのソリューションが求められる

- 複数のゲスト OS が同時動作（古典的な仮想化はゲストドメインは 1 つ）
  - 特定ドメインが占有する機能は パススルーモードで 1 対 1 マッピング
  - ゲストドメイン間で 複雑な周辺機能共有 に対応する必要がある
  - GPU などバスマスターとなる co-processor の仮想化 が必要
  - ゲスト OS からも Trusted Execution Environment (TEE) が利用できる
- パワーマネージメント と 性能チューニング
  - システム全体としての 電源管理 (パワーマネージメント)
  - ゲストドメインの リアルタイム性 の実現
  - ヘテロジーニアス構成の CPU マルチコアレイアウトへの対応
- 機能安全（ASIL-B) 対応
  - 現在 Xen ARM FUSA SIG にて実質的な対応が進行中

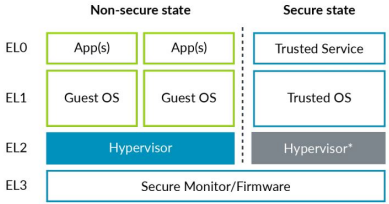
ARM プロセッサーの仮想化支援機構 (ARM v8 の例)

3 Virtualization in AArch64

Software running at EL2 or higher has access to several controls for virtualization:

- Stage 2 translation
- EL1/0 instruction and register access trapping
- Virtual exception generation

The Exception Levels (ELs) in Non-Secure and Secure states are shown here:



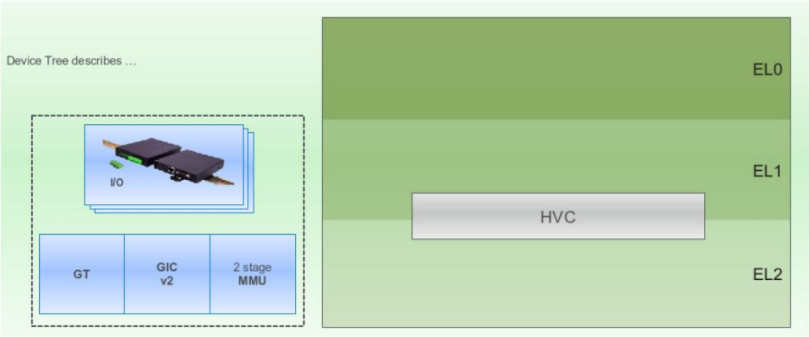
In the diagram, Secure EL2 is shown in gray. This is because support for EL2 in Secure state is not always available. This is discussed in the section on Secure virtualization.

https://developer.arm.com/architectures/learn-the-architecture/armv8-a-virtualization

Xen ARM は ARM hypervisor extension に完全対応

Xen on ARM: virtualization extensions

ARM virtualization extensions provide 3 levels of execution: EL0, user mode, EL1, kernel mode, and EL2, hypervisor mode. They introduce a new instruction, HVC, to switch between kernel mode and hypervisor mode. The MMU supports 2 stages of translation. The generic timers and the GIC interrupt controller are virtualization aware.

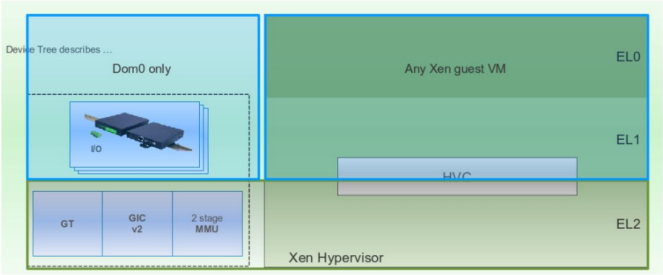


https://wiki.xenproject.org/wiki/Xen_ARM_with_Virtualization_Extensions_whitepaper

ARM CPU の privileged access（EL1）は DOM0 に集約される

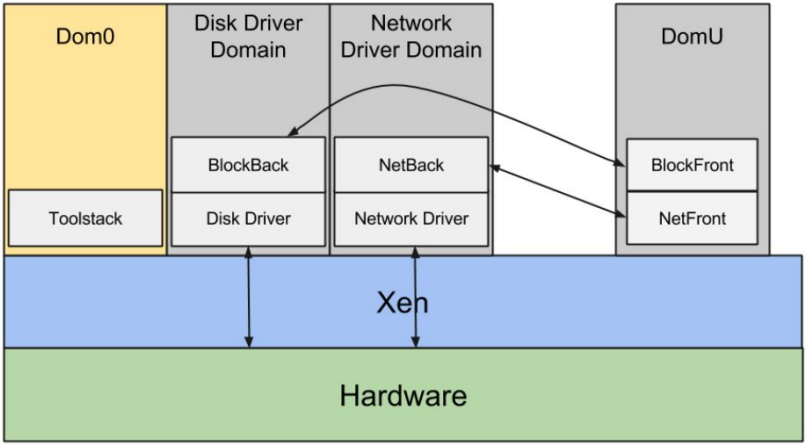
ARM virtualization extensions are a great fit for the Xen architecture:

- Xen runs entirely and only in hypervisor mode
  Xen leaves kernel mode for the guest operating system kernel and EL0 for guest user space applications. Type-2 hypervisors need to frequently switch between hypervisor mode and kernel mode. By running entirely in EL2 Xen significantly reduces the number of context switches required.
- HVC, the new instruction, is used by the kernel to issue hypercalls to Xen
- Xen uses 2-stage translation in the MMU to assign memory to virtual machines
- Xen uses generic timers to receive timer interrupts as well as injecting timer interrupts and exposing the counter to virtual machines
- Xen uses the GIC to receive interrupts as well as injecting interrupts into guests



https://wiki.xenproject.org/wiki/Xen_ARM_with_Virtualization_Extensions_whitepaper

DOMD (unprivileged driver domain) の分離による冗長構成 も可能



https://wiki.xenproject.org/wiki/Xen_ARM_with_Virtualization_Extensions_whitepaper

Xen ARM ：汎用周辺機能 PV ドライバーサポート状況

Upstreamed

- Xen protocols for Linux kernel:
  - PV Sound
  - PV Display
  - PV Camera
- Xen drivers for Linux kernel:
  - PV Sound
  - PV Display
- Multi-touch touchscreen support for existing Xen PV input protocol & Linux kernel driver
- Zero-copy buffers passing between Xen VMs for existing Xen grant Linux kernel device driver
  - Used in PV Display, can be extended to PV Sound and others

In progress

- Xen PV Camera driver for Linux kernel
- Xen PV USB implementation shall be re-considered for future use

Open source components

- Library to build Xen backends https://github.com/xen-troops/libxenbe
- PV Sound backend https://github.com/xen-troops/snd_be
- PV Display & Input backend https://github.com/xen-troops/displ_be
- PV Display Manager https://github.com/xen-troops/DisplayManager
- Pulse plug-in for GENIVI AudioManager
  - Minor fixes for GENIVI Audio manager were upstreamed to GENIVI repo

In progress

- Migrating to AGL Audio Manager and integrate virtualization support
- Migrating to AGL Display Manager and integrate virtual display manager

Xen ARM ：ゲストドメインからの OP-TEE の利用（開発中）

- In embedded system, guest domains usually require security features for:
  - Certificate validation and supplying
  - Signature verification
  - DRM and other media encryption/decryption
- Implementation of at least minimal required set of Global Platform APIs is a must for embedded & automotive system, many vendors implement rich trusted applications (TAs) and support secure devices, filesystems and so on.
- Obviously, guest domains shall have ability to use TEE services and run TAs, if system security configuration allows
- Xen intercepts and correctly alters SMC calls so that OP-TEE is aware of the calling VMID
- Global ancillary OP-TEE functions are still serviced only by the dom0

Upstreamed

- New memory management model for OP-TEE allowing multiple guests
- Support for HVC/SMC bridging for Xen
- Generic virtualization support for OP-TEE
- Generic TEE support for Xen and OP-TEE driver implementation is being merged
- Changes to OP-TEE driver Linux kernel for virtualization

In progress

- Virtualization of secure hardware
- Research of possible SEL2 scenarios support

車載制御（IVI, Gateway 用途）適用に向けた Xen ARM 最新状況

（今回、以下の詳細を説明できませんが）コミュニティ内での最新検討内容

- SOC の周辺機能のドメイン間共有の強化
  - PV ドライバーの増強（カメラ、マルチメディア対応など）
  - vIOMMU を活用し Xen セキュリティモデルを犠牲としない virtio 対応 ← Hot!
- リアルタイムスケジューリング対応
  - 各ドメインの 精密な時間管理 と full preemption モード対応
- パワーマネージメント対応
  - ゲストのイベント入力に対応した RT と CPU- aware power governanceの対応
- ヘテロジーニアス構成のマルチコアへの対応
  - ARM big.LITTLE、DynamIQ core scheduling、AMP コア間通信 対応
- セキュリティ機構の強化
  - VMI(=Virtual Machine Introspection)の ARM CPU への実装

Xen Developer virtual Summit 2020 Update （最近の話題）

Xen FUSA SIG の取り組み

FuSa SIG with Workstreams

Subgroups meet at least every other week. Partly resourced

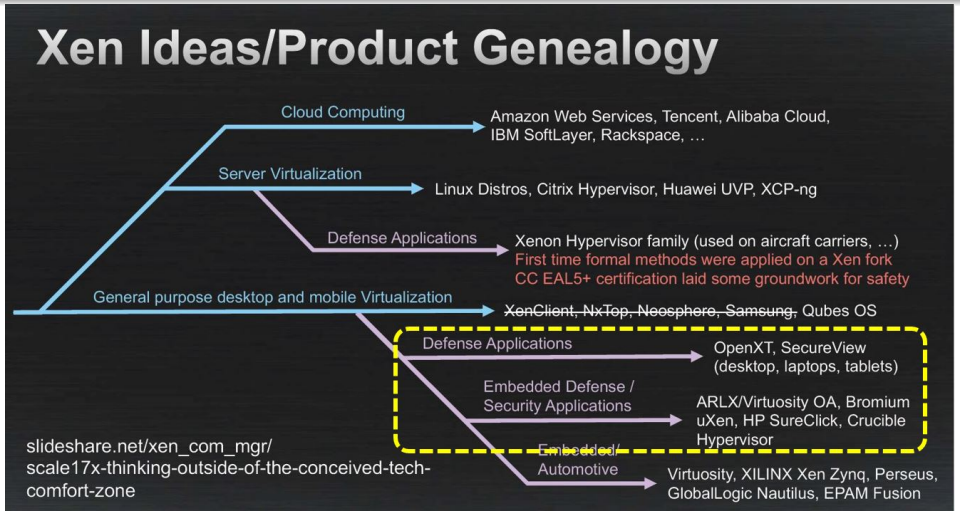
Community Reps
Lars Kurth (chair and project mgmt)
George Dunlap (committers)

Stream Owners and Implementers
Lars Kurth
THE LINUX FOUNDATION
epam
XILINX
RESILTECH
arm
RENESAS

Assessors
RESILTECH
exida
MIRA
TÜV SÜD ISO 9001

Other Members
ADIT

Xen hypervisor の 採用分野（サーバー用途だけではない） ［再掲］



FUSA 対応の例（MISRA コーディングルール対応）

Coding Standard vs Misra

Our coding standard violates MISRA C:2012, 15.6 (95% of Xen MISRA C violations on scope config)

Fixing this causes downstream pain: patch queues, backports, …

```
if ( flag_1 )
    if ( flag_2 )
        action_1 ( );
    else
        action_2 ( );
```

```
if ( flag_1 ) {
    if ( flag_2 ) {
        action_1 ( );
    }
    else {
        action_2 ( );
    }
}
```

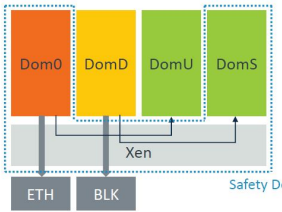
From Xen Developer Summit 2020 (1)

https://xen2020.sched.com/event/baX4/arm-contributions-to-xen-based-safety-systems-bertr

From Xen Developer Summit 2020 (2)　[projection only]

From Xen Developer Summit 2020 (3)　[projection only]

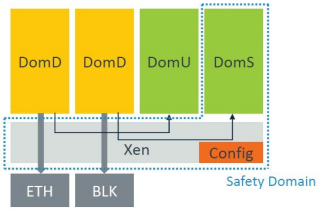
Safety Island: Standard system



- Dom0
  - Has access to all hardware
  - Create guests
  - Pass over hardware to guest
  - Control guests
  - Provide some virtual interfaces
- DomD
  - Get some hardware from Dom0
  - Can provide some virtual interfaces
- DomU
  - Rely on virtual interfaces
- DomS: safety guest
- Safety
  - Plus: flexibility
  - Minus: Linux or complex OS Dom0

From Xen Developer Summit 2020 (4)　[projection only]

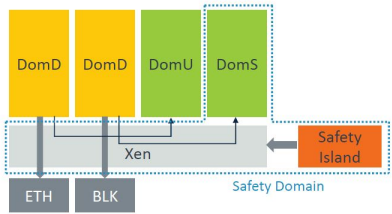
Safety Island: Dom0less



- Static configuration
  - Predefined guests
  - Predefined hardware access
- DomD
  - Get some hardware access
  - Can provide some virtual interfaces
- DomU
  - Rely on virtual interfaces
- DomS: Safety guest
- Safety:
  - Plus: safe configuration, no Dom0
  - Minus: no monitoring or flexibility

From Xen Developer Summit 2020 (5)　[projection only]

Safety Island



- Safety Island
  - Create guests
  - Control guests
- DomD
  - Get some hardware access
  - Can provide some virtual interfaces
- DomU
  - Rely on virtual interfaces
- DomS: safety guest
- Safety:
  - Plus: Monitoring and flexibility
  - Minus: Less flexibility then Dom0
  - Adapted to some safety use cases
    - And maybe others (security)

From Xen Developer Summit 2020 (6)　[projection only]

Safety Island

- Xen driver:
  - Communicate with the safety island
    - Hardware or application specific
  - Gateway with Xen interfaces
    - Defined subset of interfaces
    - Guest creation/destruction
    - Monitoring
- Safety island
  - RTOS or stand-alone application
  - Could be hosted in different places
    - On a dedicated core
    - On an other processor
    - On a different system
    - Inside Xen directly (simple driver only system)