



Virtualizing the Locomotive: Ready, Set, Go!

Mark Kraeling, GE

KVM Forum

August 19, 2015

Imagination at work.

Presentation Disclosure

The view, opinion, and position expressed by the author is the author's alone, and do not necessarily reflect the views, opinions or positions of GE Transportation, GE, or any employee thereof. The author makes no representations as to accuracy, completeness, timeliness, suitability or validity of any information presented and will not be liable for any errors, omissions, or delays in this information or any losses, injuries or damages arising from its display or use.



Agenda

The Locomotive System

Use Cases for Virtualization

Hardware Platform

What's Important to Us in Virtualization



Speaker Background

Mark Kraeling

- GE Transportation Cab Electronics
 - Products for Locomotive Onboard
 - Based in Melbourne, FL

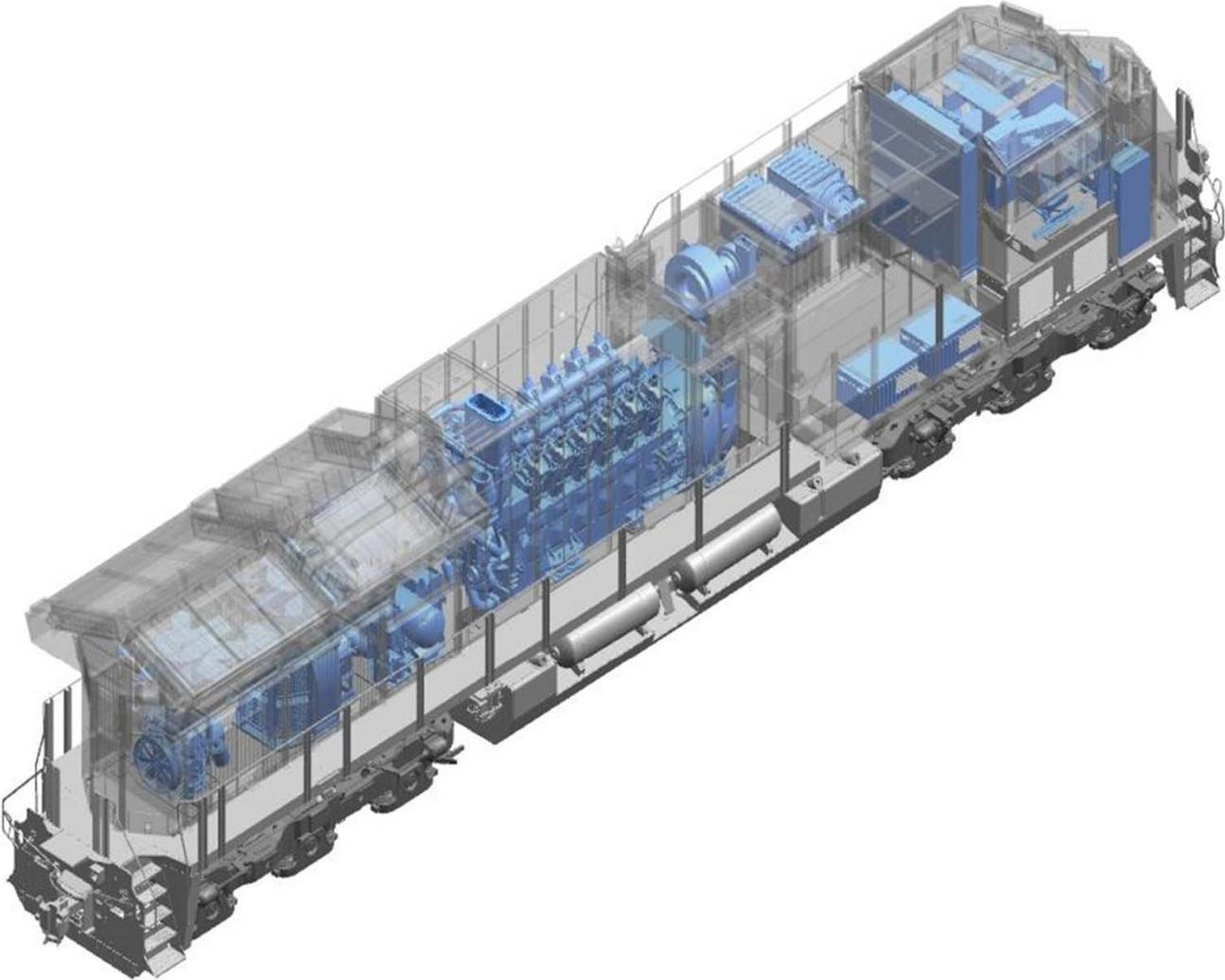
- Product Manager/Architect
 - Wired/Wireless Communications
 - Linux and ARM®-based Designs



The Locomotive System



The Locomotive



It's All About Information...

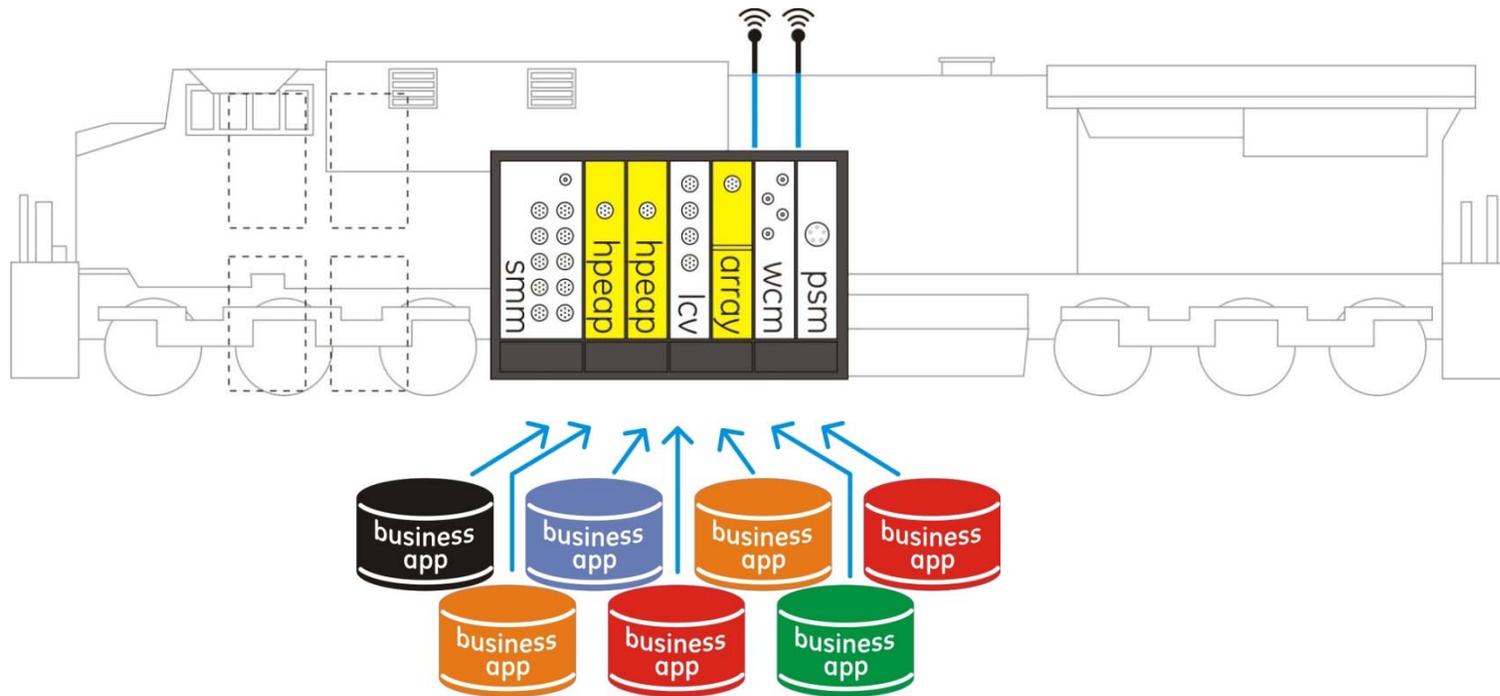


It's All About Information...



Smart Data Processing

Want to process data onboard – offload alerts, data snippets, and messages when needed

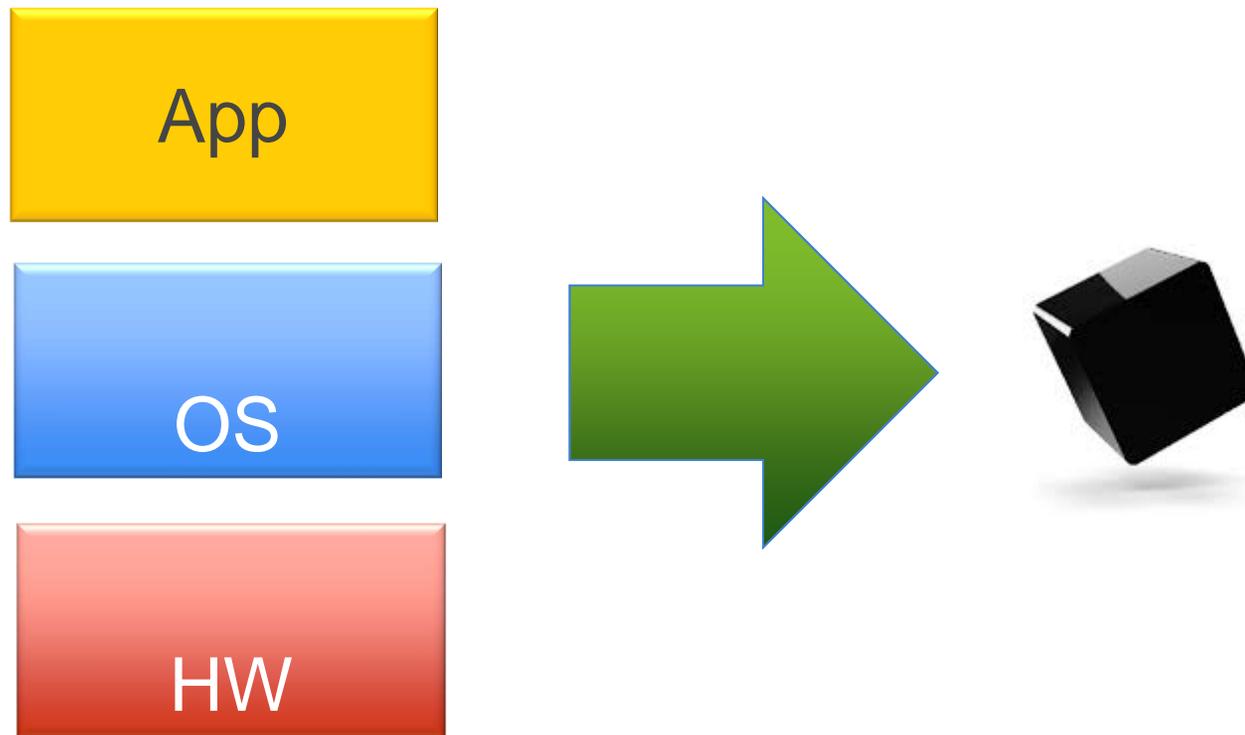


Use Cases for Virtualization



What GE Transportation Did...

Single OS runs on a single hardware platform, typically with a collection of applications based on resources



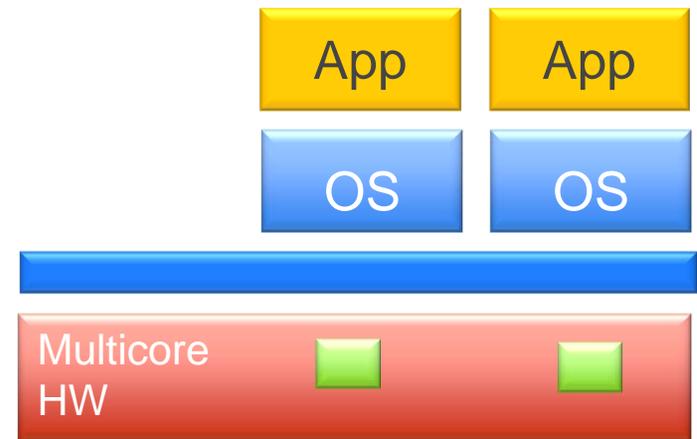
Virtualization Use Cases – Cost Reduction

Consolidation :

- Redeploy multiple discrete systems/domains onto a single multi-core processor

Benefits:

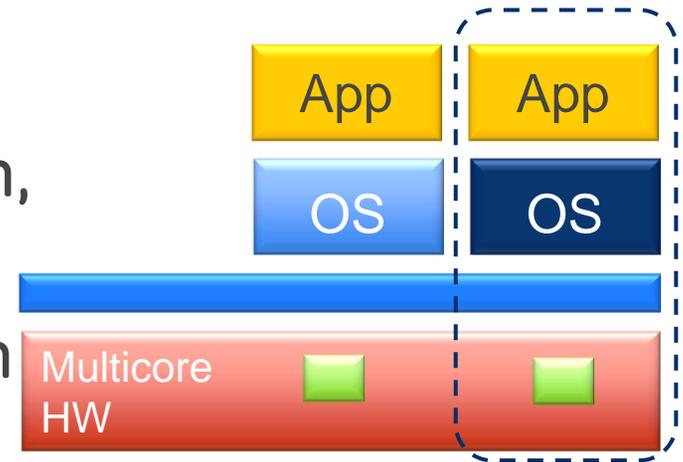
- Cost effective : bill-of-material, power
- Preserve investment : software re-use
- Improved hardware utilization
- Flexibility



Virtualization Use Cases – Reliability & Protection

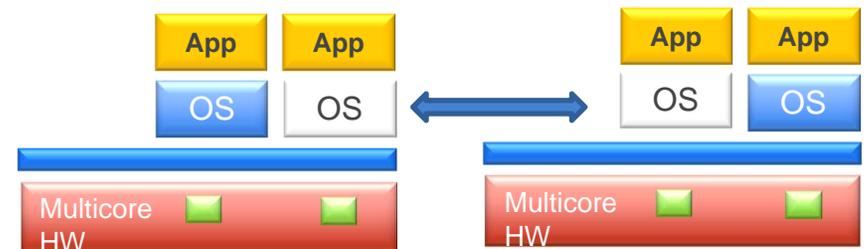
Sandboxing :

- Add untrusted software to a system, e.g. operator applications
- Run GPL based software in isolation
- Run test software safely
- Isolate security-sensitive tasks :
access rights control,
rule definitions, key management, ...



High availability

- active/standby configuration without additional hardware



Virtualization Use Cases – Flexibility & Scalability

Run legacy software / OS on Linux

Add functionality to existing system by dropping in a VM

Use different versions of the Linux kernel

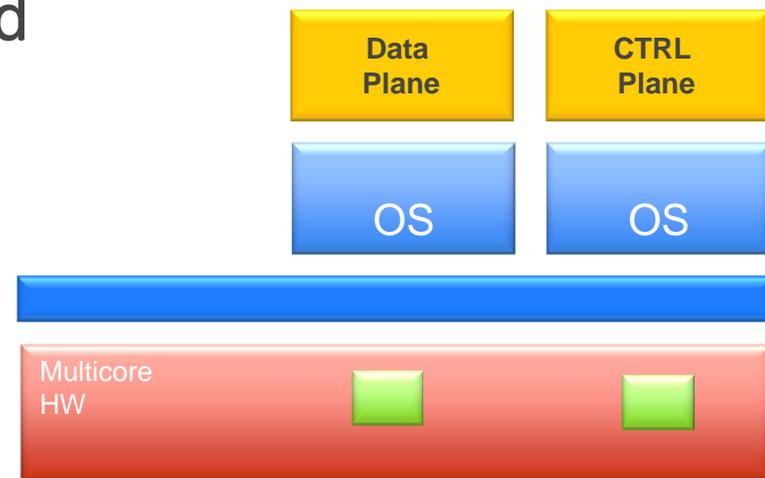
Better resource management

- Allocation of physical CPUs / control CPU load
- Create/destroy VMs as needed

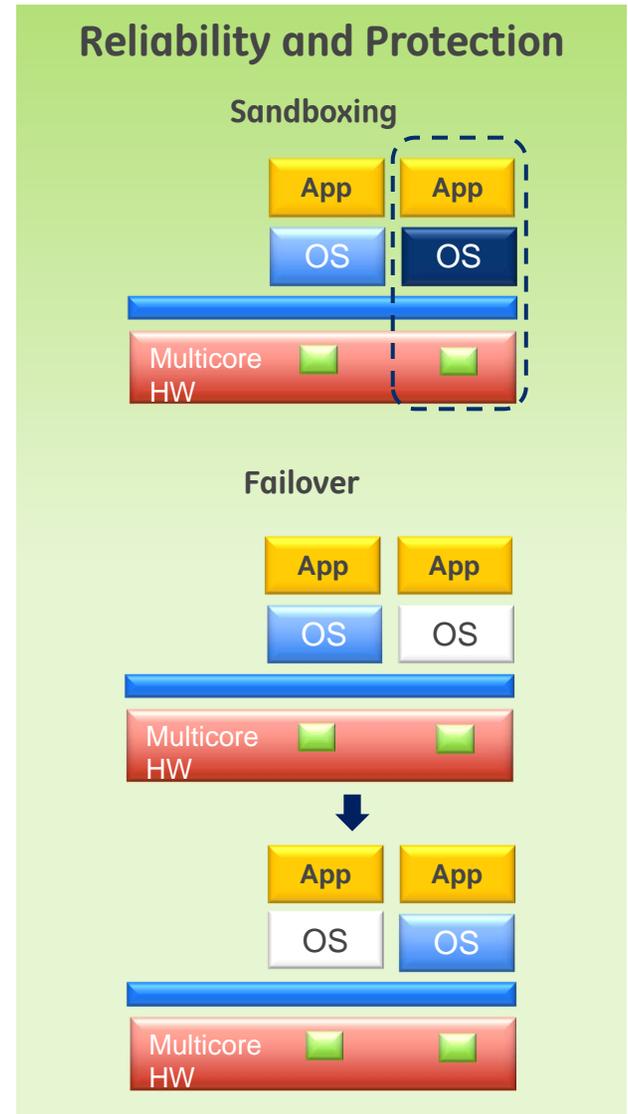
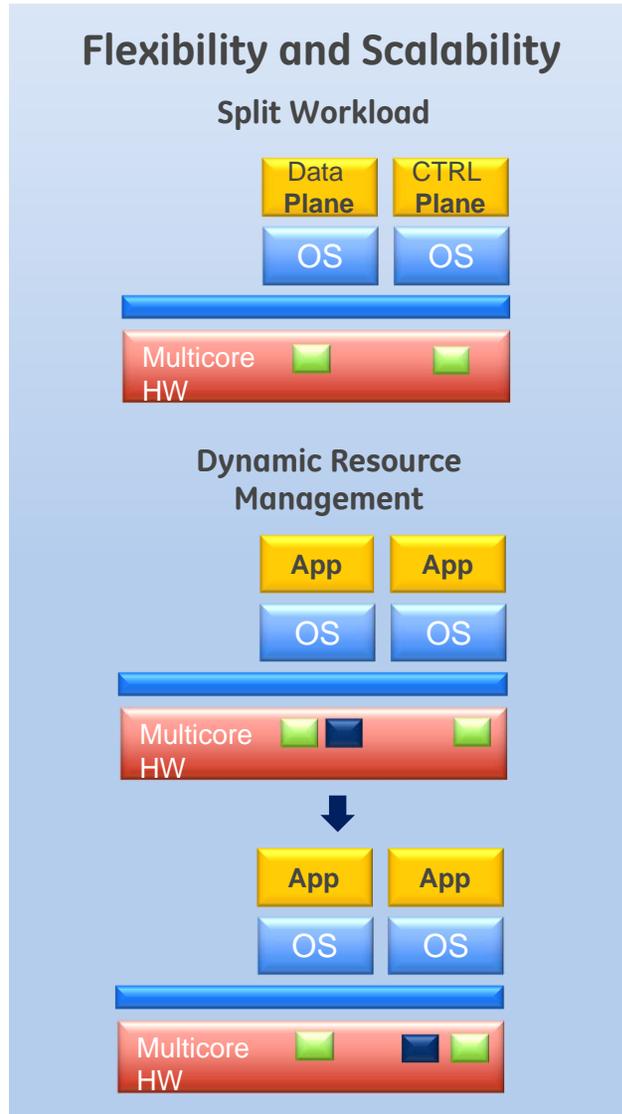
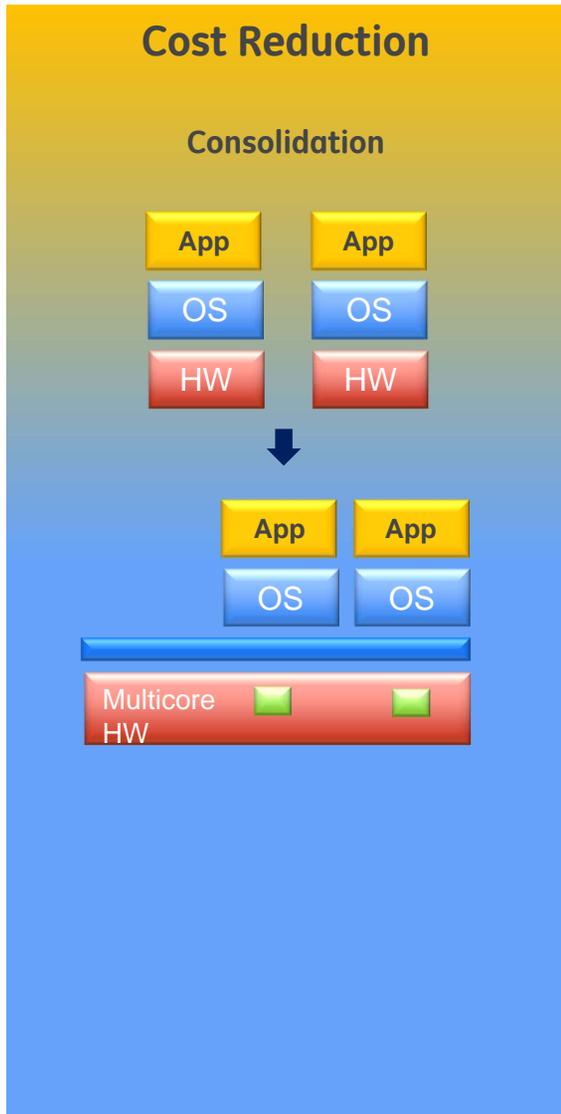
Maps well to split workload

- e.g. control plane, data plane

In-service upgrade



Virtualization Use Cases: Summary



Hardware Platform



Locomotive Chassis

Shift from Separate Boxes to Modules that Support Applications

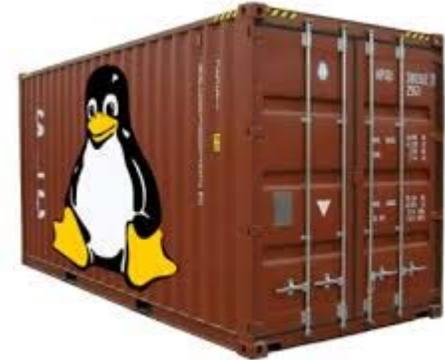
- Simplification
- Reduced cost
- Free up space
- Redundancy



Linux Containers

Don't Shoot the Messenger...

- Being considered for ARM®-based solutions where operating systems could share a common kernel
- Performance on 32-bit becomes important



Lightvisors

- LXC / LXDM (OS)
- Docker / Rocket (Apps)

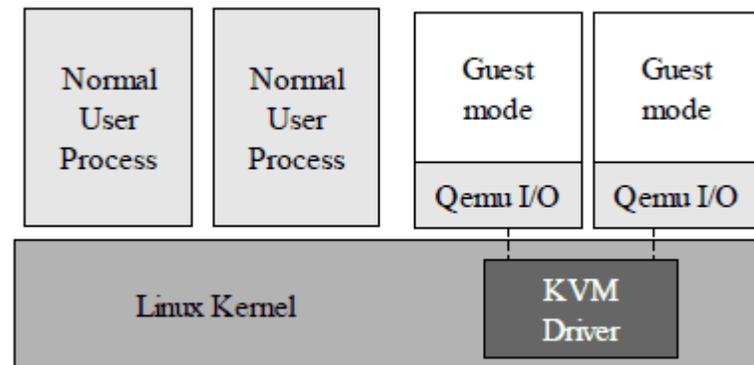


What's Important to Us in Virtualization



KVM Execution

- Boot times not necessarily important – typically applications power-up and run for a very long time
- For 32-bit processors, need hypervisor to limit amount of resources that are required (low CPU and RAM memory overhead)



<http://www.linuxinsight.com>



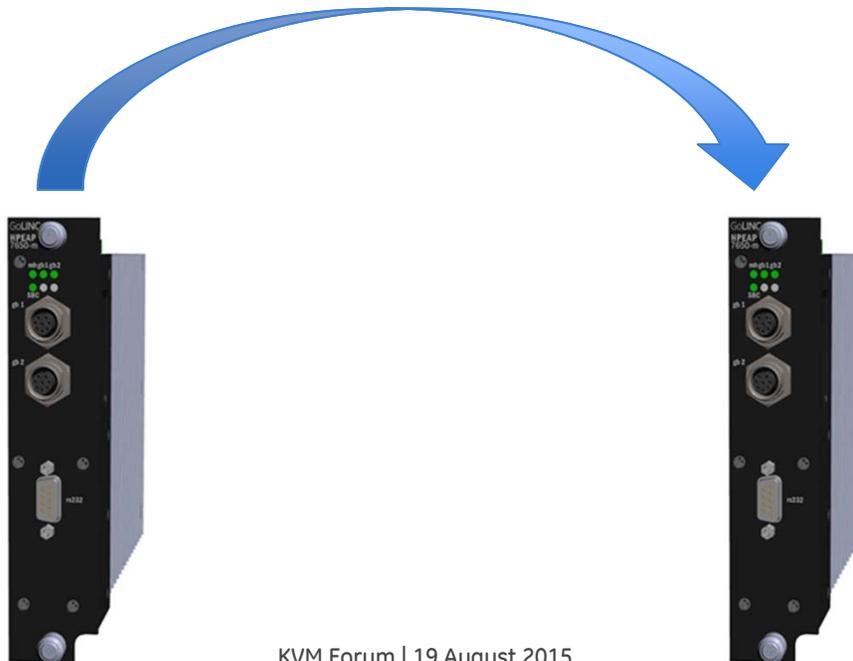
Local Management of Guests

- Need rules-based capability for HyperVisor to automatically restart guests, or even start a different guest
- HyperVisor to Guest handshaking / status
- Better pool management – HyperVisor can coordinate with others in the pool, and make smarter decisions including migration of guests where it makes sense



Standardized Imaging

- Probably something easy... but...
- Need non-UUID fixed ways to move entire HyperVisor images from one module to another
- Includes pooled pairs being able to be swapped out and have pools maintained (network-based rules)



Conclusions

- Virtualization is necessary for a locomotive mobile data center – many applications and data resources
- 32-bit hardware (to date) has not quite had enough horsepower to run KVM effectively, but more work is required here
- 64-bit hardware necessitates virtualization – must be able to leverage multiple applications on hardware

