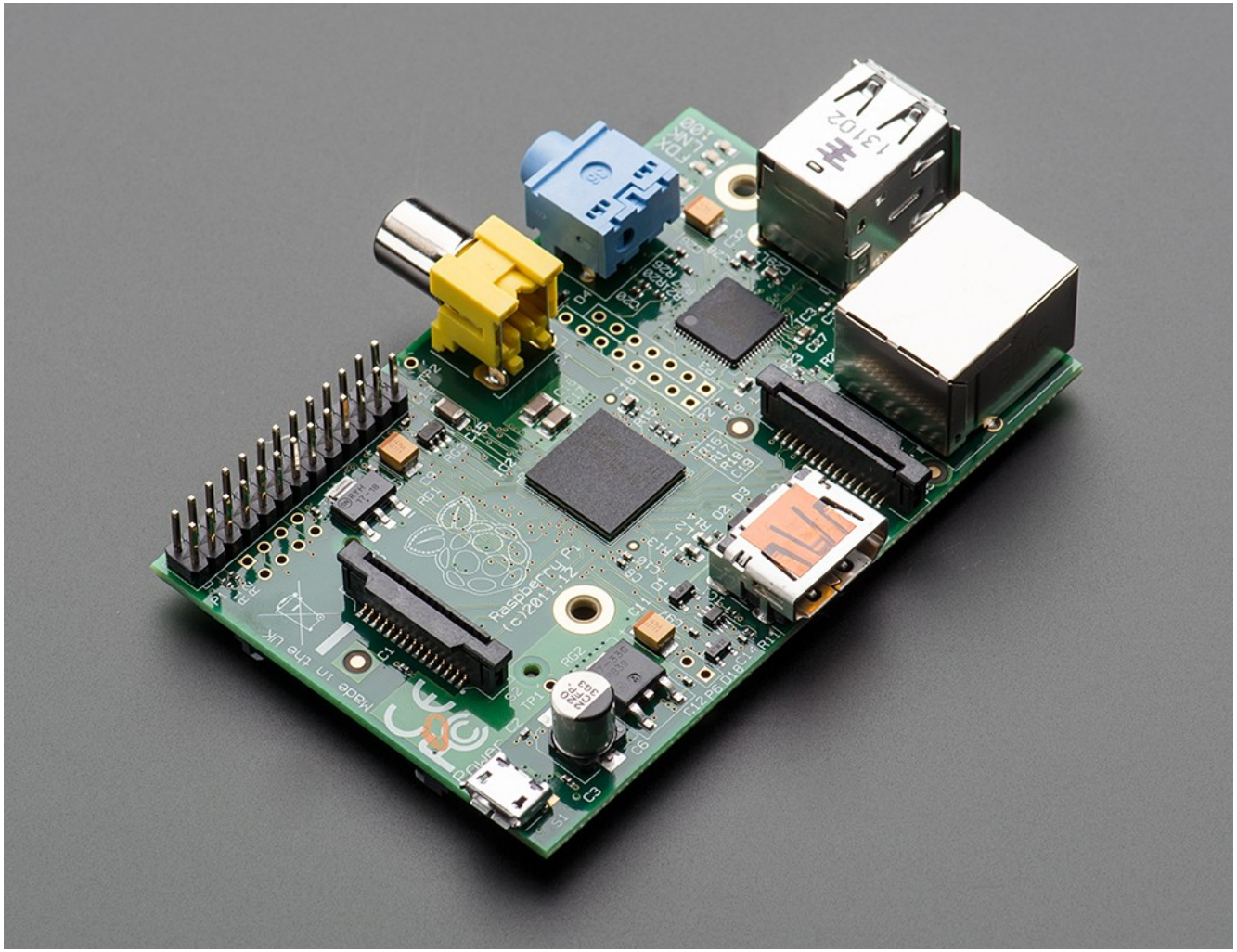
LISA '13
27th Large Installation System Administration Conference
November 3–8, 2013 • Washington, D.C.

Hyperscale ARM Servers

Jon Masters

Chief ARM Architect, Red Hat

usenix
THE ADVANCED COMPUTING SYSTEMS ASSOCIATION



Raspberry Pi Model B
(Supported by Pidora, a Fedora Remix)

World's first bicycle-powered ARM Server
(HP Redstone/Calxeda)

HP Moonshot SL1500 Chassis
(3 ARM cartridges announced so far)

The old Fedora ARM Project build system

The current Fedora ARM Project build system
(Boston Limited Viridis (Calxeda))

Warren East Announcing Linaro Enterprise Group

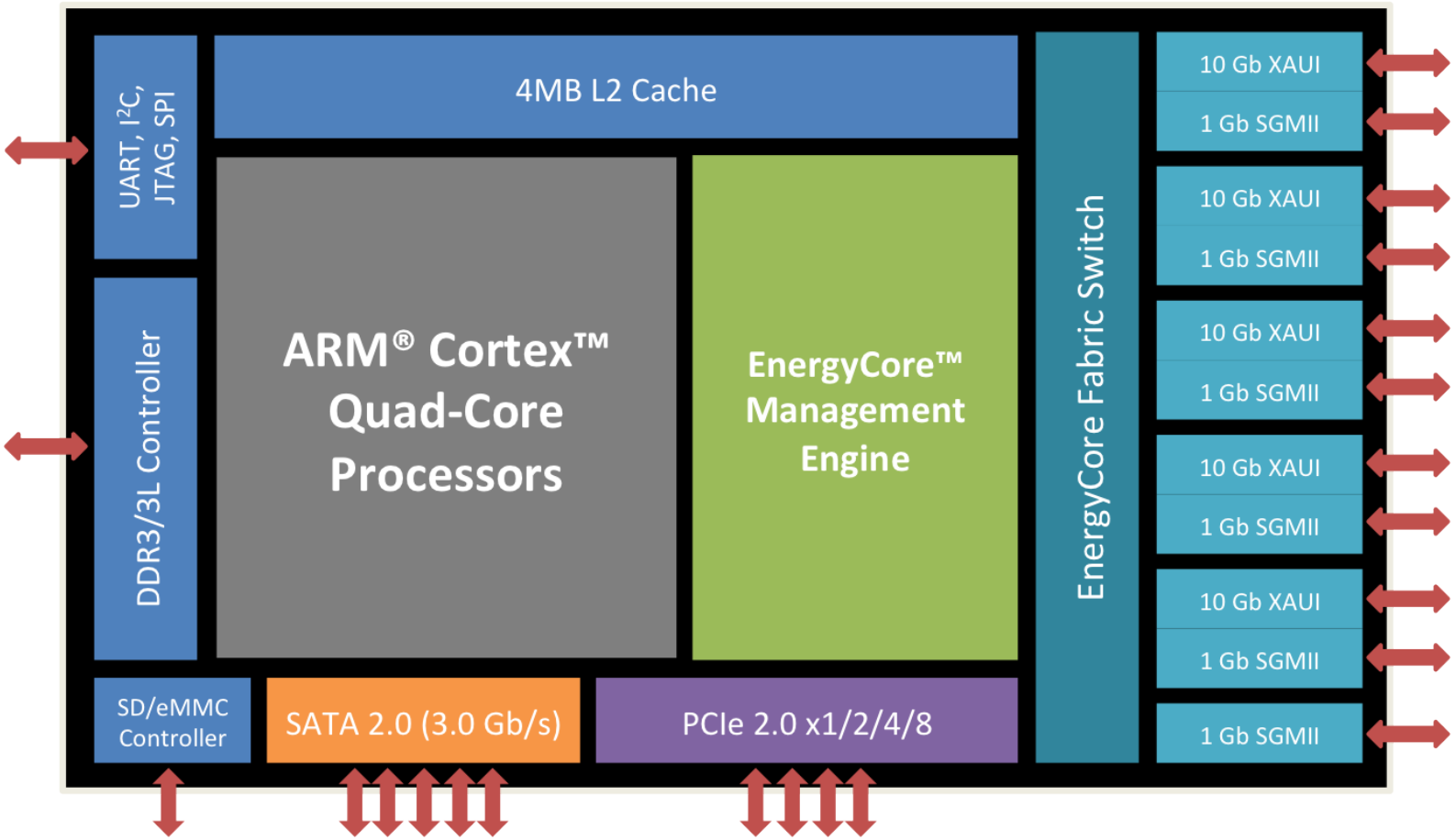
What a difference a year makes...

- 32-bit ARM servers now used in production
  - Running our 32-bit Fedora builds today
  - Building competence ahead of 64-bit
- Hyperscale Computing now a broader term
- Red Hat co-founded Linaro Enterprise Group
  - LEG celebrates one year this week
- Fedora AArch64 Bootstrap exercise
  - Nearly complete Fedora builds

What is Hyperscale Computing?

- SoC Commoditization of traditional server
  - Much higher density (1,000s nodes/rack)
  - Lower energy profile, reduced PUE
- Fabric Interconnectivity to other nodes
  - Disaggregation (OpenCompute example)
- Integrated Management and Offload
  - Orchestration at very large scale
  - Value in accelerators offloading from CPU
- Failure-in-Place

- 1,000-10,000 server nodes per rack
  - Tightly connected, rack-level granularity
- Aggregate performance
  - Targeting “good enough” linear CPU
    - PPA tradeoffs favor “sweet spot”
  - Increasingly data-movement focused
- Technical challenges
  - (re)provision elastically, and very often
  - Leverage integrated management, etc.

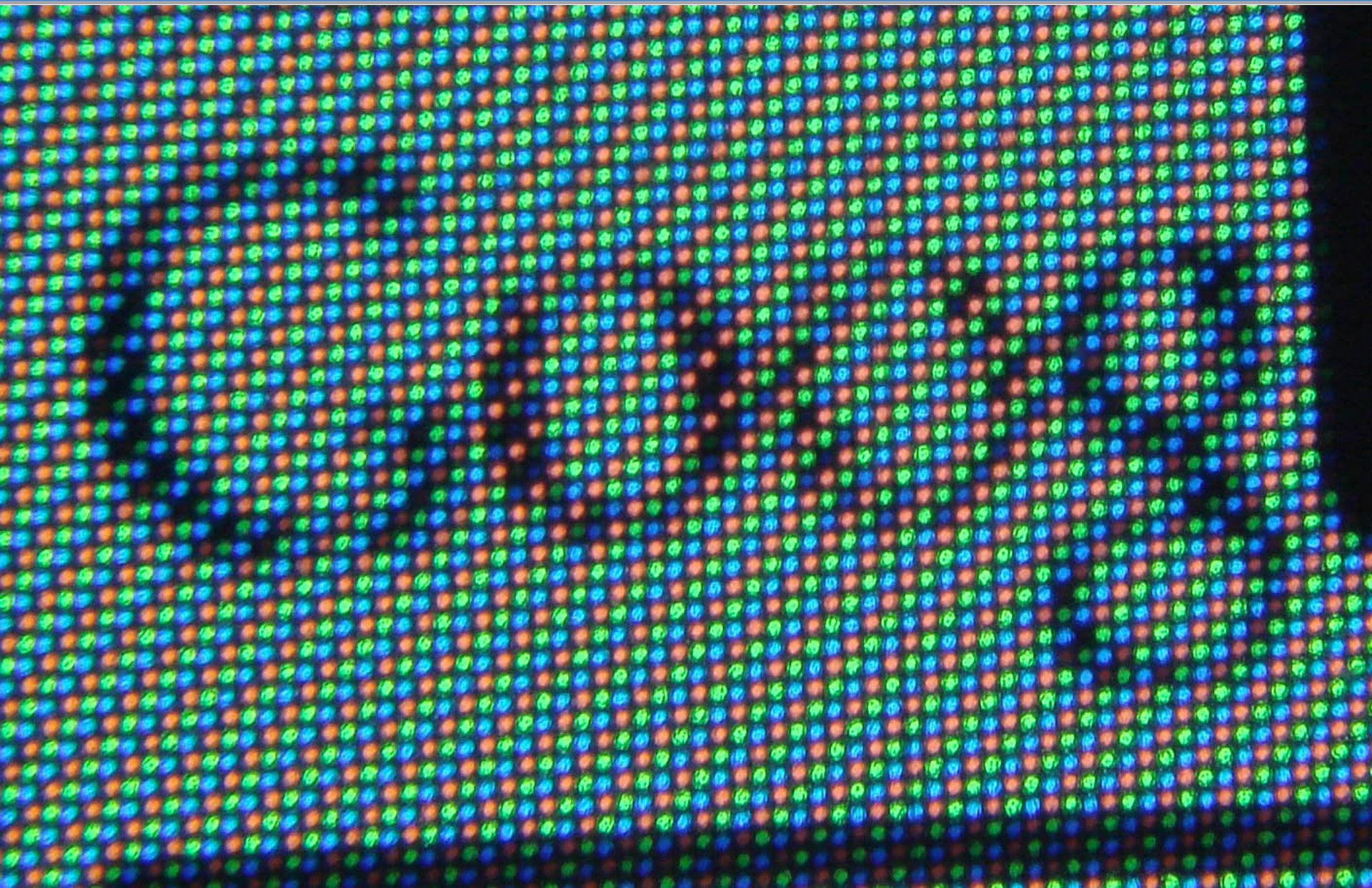

UART, I2C, JTAG, SPI
4MB L2 Cache
DDR3/3L Controller
ARM® Cortex™ Quad-Core Processors
EnergyCore™ Management Engine
EnergyCore Fabric Switch
10 Gb XAUI
1 Gb SGMII
10 Gb XAUI
1 Gb SGMII
10 Gb XAUI
1 Gb SGMII
10 Gb XAUI
1 Gb SGMII
10 Gb XAUI
1 Gb SGMII
1 Gb SGMII
SD/eMMC Controller
SATA 2.0 (3.0 Gb/s)
PCIe 2.0 x1/2/4/8

- System-on-Chip revolutionized mobile
  - Now is revolutionizing the server space
  - CPU, IO, devices all integrated on-chip
  - Servers add offload/acceleration engines
- Increasingly verticalized/differentiated world
  - Future servers have much more variety
  - Organized, value-added differentiation
- Fabric and management integrated on-chip
  - IPMI, industry standard interfaces

- Fabric powers future datacenters
  - The next battleground between vendors
  - Obviate cabling/top-of-rack switches
- Disaggregate resources (split depreciation)
  - Separate Compute from other resources
  - Connect network, memory, storage, PCIe
  - 100GBit+ networking, silicon photonics
- NFV and SDN are just workloads
  - Separate physical/virt fabric topology

- Datacenter management of whole racks
  - Chassis-level asset/topology tracking
  - SoC-level KVM/power via IPMI, etc.
- Offload/acceleration engines
  - Traditional network acceleration
  - FPGA accelerators, custom IP blocks
- Compute is the boring part of the story

LISA '13
27th Large Installation System Administration Conference
November 3–8, 2013 • Washington, D.C.



- The (very) low-hanging fruit
  - Web Tier
  - Cloud Computing
- New opportunities
  - Network Function Virtualization
  - Software Defined Networking
- Existing markets
  - Traditional Enterprise computing

The Red Hat ARM Team (RHAT)

- Formed three years ago to focus on 64-bit
  - Small core team focused on architecture
- Accelerated development of 32-bit Fedora
  - Bootstrapped ARMv7 “hard float” ABI
- Lead the bootstrapping of 64-bit Fedora
  - cross-compilers, models, and hardware
- Co-founded Linaro Enterprise Group
  - 2 assignees working on ACPI, validation

32-bit vs. 64-bit ARM

LISA '13
27th Large Installation System Administration Conference
November 3–8, 2013 • Washington, D.C.

- Drop-in alternative driven by demand
  - Must “just work” for boot/install/etc.
  - Cannot require users to change workflow
- Must be standardized
  - “An ARM is an ARM is an ARM”?
- Must be manageable (orchestration)
  - Leverage OpenStack
  - On-chip management, etc.
- Learn from other architectures

- U-Boot is an embedded 32-bit bootloader
  - Intended for vertical integration scenario
- Not a standardized environment
  - Different magic values per platform, etc.
  - Many vendor extensions not upstream
- DeviceTree used to describe hardware
  - Bindings evolving over time

- Coordinated base platform standards
  - ABIs, Binary Formats, Linux specific bits
- Remove gratuitous differentiation
  - Base hardware assumptions
    - Standardized, compatible interfaces (AHCI/XHCI/UARTS...)
    - Enumerable hardware (ACPI, PCIe, etc.)
  - Boot Architecture
    - UEFI abstracts underlying hardware
    - Network, local, removable media

- Red Hat assisted in Fedora ARMv7 bootstrap
  - Smokescreen “hard float” ABI as cover
  - Learning exercise anticipating ARMv8
- Derived a multi-stage process for bootstrap
  - Cross-compilation process at first
  - Switch to native early (for correctness)
  - Migrate into containerized build (mock)
  - Transition to web-based build (koji)
- Preparation for Koji currently in progress

- Early access to architecture documentation
  - Initiated long-term work (OpenJDK, etc.)
- ARM FAST/Foundation Models
  - Cross-compiled minimal environment
- Booted minimal environment on models
  - Built 12,000 packages very (very) slowly
- Applied Micro X-Gene hardware transition
  - Can build software in hours, not weeks

- Red Hat assisted in initial kernel review
  - Actively developing new kernel features
- Lending resources to critical pieces in Linaro
  - UEFI kernel shim, ACPI integration, etc.
- Porting Red Hat-specific components
  - Orchestration and management software
- Assisted in hardware validation and test
  - Extensively collaborated w/Applied Micro

- Applied Micro X-Gene XC-1 Server Board
  - 8-core, multi-GHz, custom 64-bit design
- Fedora 19 Remix
  - Running exclusively in 64-bit AArch64
  - 32-bit AArch32 is not supported
- 12,000+ software packages ported
  - Including LAMP stack, GlusterFS

- Questions?

https://fedoraproject.org/wiki/Architectures/ARM/
#fedora-arm (Freenode IRC)