Linux and Open Source Consolidation on ARM

Philippe Robin
Director, Open Source
ARM Ltd.

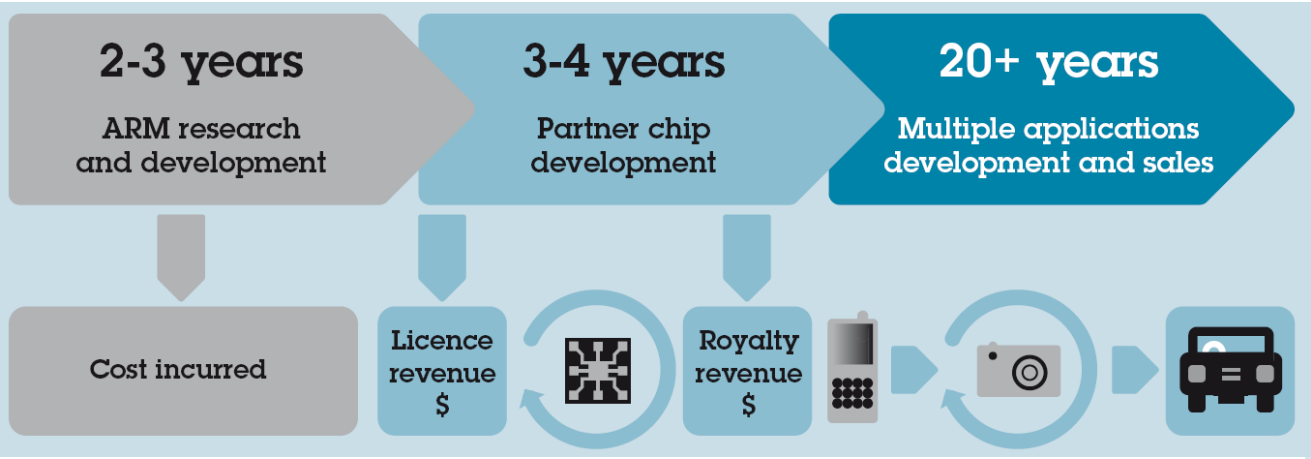
SCALE 2012
January 20-22

Outline

- ARM – company, architecture and products
- Scope for competition and differentiation with open source
- Partners, platforms and open source adoption
- Linaro - partner and community engagements
- Openness and development cycle
- Conclusion

ARM - Introduction

- ARM is Global leader in the development of semiconductor IP
  - R&D outsourcing for semiconductor companies
- Innovative business model
  - Upfront license fee – flexible licensing models
  - Ongoing royalties – typically based on a percentage of chip price
  - Technology reused across multiple applications
- Long-term, secular growth markets

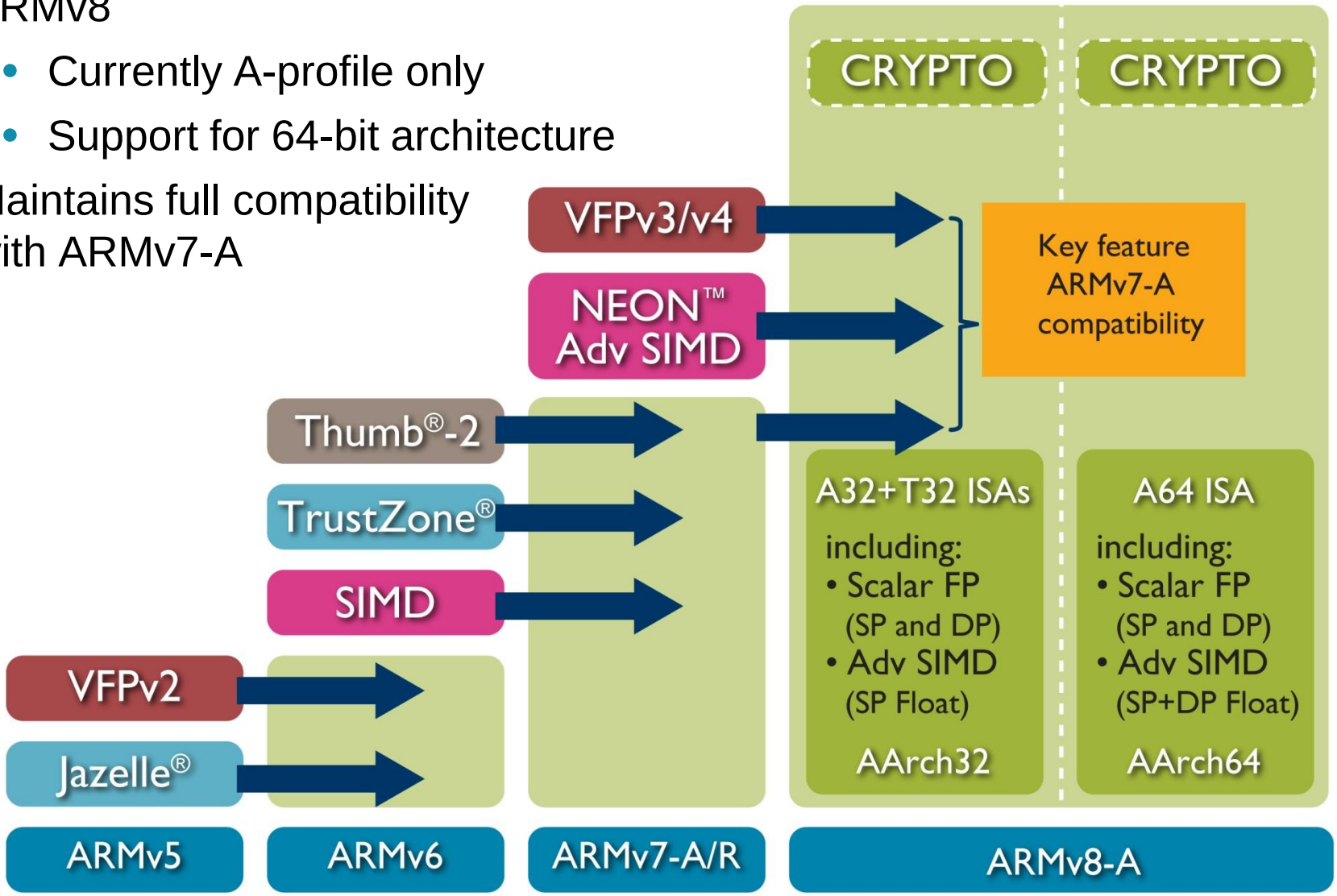


Approximately 800 licenses
Growing by 75-100 every year

>6bn ARM-based chips in '10
>25% CAGR over last 5 years

ARM Architecture Evolution

- ARMv8
  - Currently A-profile only
  - Support for 64-bit architecture
- Maintains full compatibility with ARMv7-A

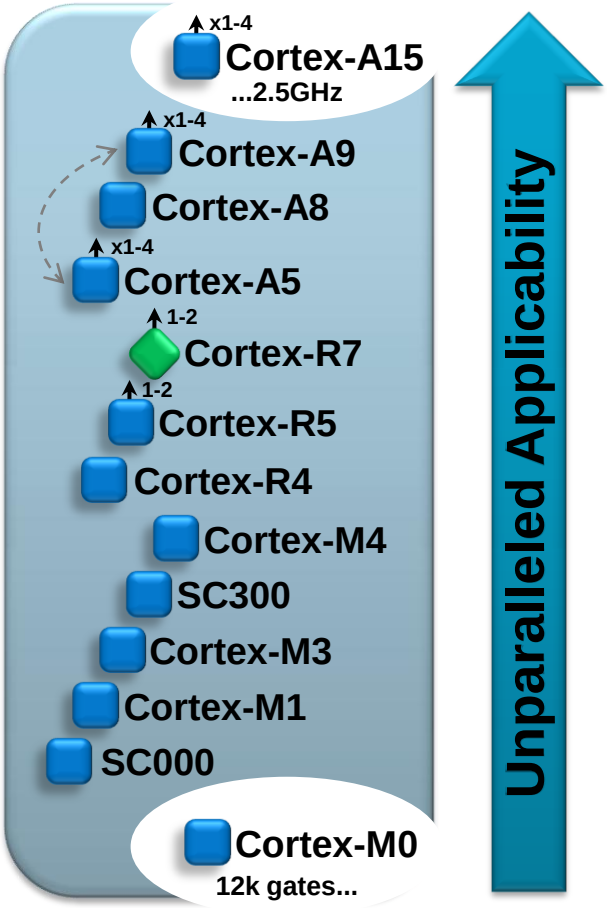


ARMv7 - Cortex Processor Family

Architectural innovation, compatibility across diverse application spectrum

Cortex™
Low-Power Leadership from ARM®

- ARM Cortex-**A** family:
  - Applications processors for feature-rich OS and 3rd party applications
- ARM Cortex-**R** family:
  - Embedded processors for real-time signal processing, control applications
- ARM Cortex-**M** family:
  - Microcontroller-oriented processors for MCU, ASSP, and SoC applications



From CPUs to Products

- OEMs are innovating with broad range of products
  - Latest Cortex CPUs include Cortex-A15 and big.LITTLE
    - Enabling support for virtualization, MP clusters
  - Differentiation at SoC integration and at product level
- Most products use ARMv7 CPUs but...
  - Companies also use their own IP blocks and optimized components for dedicated functions
  - Leads to dedicated software support because of variations in how some peripherals are working
  - For example, NEON may not necessarily be present on all Cortex-A based platforms as this was not mandated

Growth Segments

Differentiation and Software Support

- http://lwn.net/Articles/464530/ - LCE2011 Kernel developer panel
  - "*He (Torvalds) likes the architecture and instruction set of the ARM processor, but he doesn't like to see the fragmentation that happens because there is no standard platform for ARM like there is for x86. There are no standard interrupt controllers or timers, which is a mistake, he said*"
  - *The good news is that ARM Linux is getting better, and the ARM community seems to be making progress, so he is much happier with ARM today than he was six months ago. It's not perfect, and he would like to see more standardization, but things are much better. Torvalds said that he doesn't necessarily think that the PC platform is wonderful, but "supporting only a few ways to handle timers rather than hundreds is wonderful*"

The Case of the Linux Kernel

- Situation summarized recently by Paul McKenney's in LWN note last May:
  - http://lwn.net/Articles/443510/
  - ~75% of all the architecture-specific changes in <*linux-next*> were for ARM architecture
  - Many ARM sub-architecture implemented a number of drivers
    - IRQ, clocksource, GPIO, PCI, IOMMU, CPUFreq etc.
    - ➔Best to move most of this code out of the sub-platforms, find the commonalities, and eliminate the duplications
- This is emphasized as Linux and Open Source are more popular than ever with OEMs and consumer devices
  - Partners are now contributing back to the upstream projects!
  - Software need to adapt to scale to quite large number of platforms
  - Contributors to be more disciplined with contributions and maintaining
  - Also ensure maintenance of Linux kernel on older platforms

Building Single Kernel Image

- Enable a single kernel binary image to support as many ARM platforms as possible
  - Build support for platforms from the same family into a single kernel
    - Selected at run time through the '*machine_desc'* structure
  - Support multiple CPUs (MMU/TLB/cache) into a same kernel (selected at run time)
  - Use *'irq_chip'*, '*gpio_chip'* and '*platform_device'* for peripherals
  - Enable multiple clock API implementation and selection at run time
- Eliminate duplicated code supporting similar hardware blocks and eliminate 'hard-coded' values
  - Interrupt support code is an example for this
- This is coupled with enabling FDT support to describe the mix of peripherals available on the SoC at runtime
- Single kernel images are necessary to support infrastructure and server style platforms

Some of the Key Areas for Consolidation

- Platform Support
  - Flat Device Tree – community and Linaro efforts
    - http://devicetree.org/Device_Tree_Usage
  - Enable single Linux kernel boot image
    - https://wiki.ubuntu.com/Specs/ARMSingleKernel

- Power Management
  - http://elinux.org/images/0/09/Elce11_pieralisi.pdf

- Multimedia

  - http://www.linaro.org/linaro-blog/2011/07/15/multimedia-on-linux/
- Graphics
  - http://www.linuxplumbersconf.org/2011/ocw//system/presentations/567/original/umm.odp
  - Memory management requirements to support GPU and Multimedia

Linux Power Management on ARM

- ARM partners are taking a more aggressive and holistic approach to power management with the Linux kernel
  - Generalize code for SMP power down so that this is used by more platforms
    - This applies to CPU coordination and hotplug, suspend() and idle()
  - Additional work on CPU idle and '*sched_mc*' as long term solution to manage processor's idle status
    - Scheduler migrate threads in a power efficient manner
    - https://wiki.linaro.org/WorkingGroups/PowerManagement/Specs/sched_mc
    - http://status.linaro.org/11.12/group/tr-power-smp-sched-mc.html
  - PM notifiers and CPU/L2 suspend/resume code provide the basis for common power management on ARM platforms
- Code is being consolidated on large number of ARM SoCs
  - Code reviews & sprints at Linaro Connect / ELC events with Linux kernel developer community

Graphics, DMA and Memory Management

- Contributions to contiguous memory allocator (CMA) work
  - Jonathan Corbet describes the situation in the following LWN articles
    - http://lwn.net/Articles/447405/ and http://lwn.net/Articles/440221/
  - Need to align interfaces for the management of IOMMUs
  - Address issues summarized by Arnd in http://lwn.net/Articles/440227/
  - Some of the subsystems involved include:
    - DRM (rendering and display), V4L2, ALSA (sound), DSP
- Efforts to reduce the number of alternative implementations for memory allocation and buffer management
  - Including GEM, PMEM, CMEM, UMP
- Open source GPU drivers?
  - As people may ask, the following blog outlines some of the issues:
    - http://www.linaro.org/linaro-blog/2011/11/30/a-case-for-open-source-gpu-drivers/

Open Source Development Tools

- Ensure continuous tracking and monitoring of upstream projects such as GCC, binutils and LLVM
  - Systematic test and build with validation against platforms
    - Linaro LAVA infrastructure, distributions and vendor's activity
  - Early contribution to upstream projects for new CPUs
  - Enable open, reproducible build and test frameworks

- Improve base Linux support for debug and profiling on ARM
  - Enabling support for performance events
  - Provide support for new debug and profiling features (STM/Coresight)
- Tools and OS support to analyze power consumption
  - PowerDebug and Powertop - http://www.linuxpowertop.org/
  - Kernel support through sched_mc
    - https://wiki.linaro.org/WorkingGroups/PowerManagement/Specs/sched_mc

Partners, Platforms and Communities

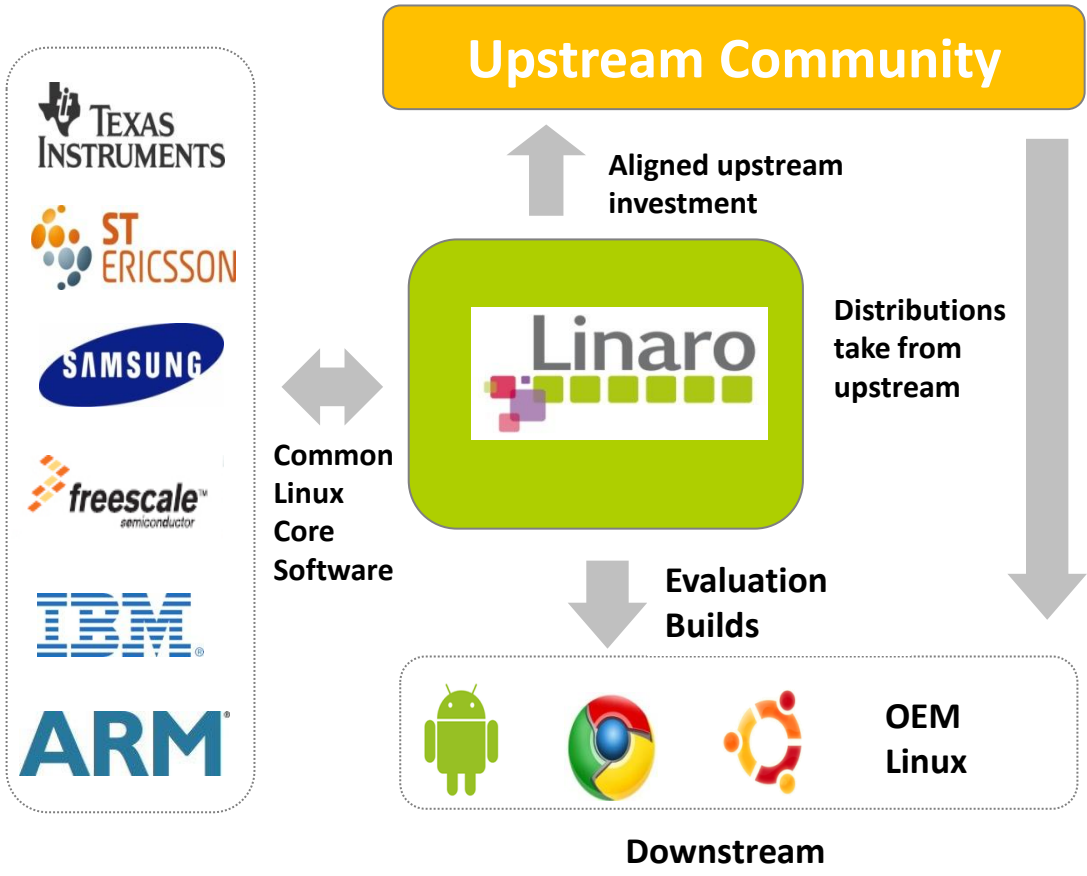
Upstream Open Source Projects and Communities
(Linux Kernel, GNU Tools etc.)

From Embedded to Full Distributions

- Historically, ARM Linux developments were focused on Embedded and done using 'DIY-style' approach
  - Easier to customize to device/product requirements with constrained memory footprint and ensure real-time performance
  - Frameworks such as OpenEmbedded enable cross-build of Linux kernel, Development Tools and other packages
  - Many changes often viewed as ad-hoc and not necessarily pushed back upstream which does not necessarily help to consolidate upstream
- Move to native build environment with familiar desktop distributions
  - ARM platforms now suitable to this development model
  - Started initially with Debian and Ubuntu distributions
    - Helped build and test large number of upstream packages
    - Validated transition to Thumb-2 and Hard-FP
- Now transitioning to infrastructure and server platforms
  - ARMv7 / Cortex-A15 is stepping stone to enable transition
  - ARMv8 architecture more suited to server distributions and platforms

Linaro – Partner Collaboration

- Created to enable collaboration to optimize open source software for ARM-based System-on-Chips
- Address under-resourced projects and contribute optimizations upstream
- Distributions usually draw directly from upstream
- Produce validated kernel and tools for direct use by OS distributions
- Provide framework for open collaboration
  - Linaro members
  - Community and Open Source developers



Linaro – Developer Collaboration

- ‘Connect’ events to bring Partner developers and community members together to discuss on range of technical issues
  - https://wiki.linaro.org/Events/
  - http://elinux.org/Events/Kernel_Summit_2011_ARM_Subarch_Maintainership_Workshop
  - https://wiki.linaro.org/Events/2011-06-MMWG
- Creation of ARM SoC GIT repository
  - Enable further testing and integration of the Linux kernel on range of SoCs ahead of pulling changes into main kernel tree
    - http://lwn.net/Articles/443515/
- Further Optimize and validate OSS Development Tools
  - Maintain and validate sets of development tools
    - https://launchpad.net/gcc-linaro
    - Include source and binary versions to facilitate development
    - Also support the embedded community with ‘bare-metal’ GNU GCC Tools

Upstream, Upstream .... Upstream!

- Upstream contributions and collaboration are the primary objectives

> Also notable is the continued slow climb by companies like Texas Instruments and Samsung; Nokia, instead, appears to be about to fall out of the top 20. The handling of Linaro deserves an explanation: contributions by Linaro assignees is normally credited back to their home companies. Nonetheless, Linaro makes an appearance on its own here as the result of the work of an increasing number of engineers employed by the organization itself.

Source: LWN, Sept 28, 2011

(None)	1111	13.1%
Red Hat	882	10.4%
(Unknown)	749	8.8%
Intel	616	7.3%
Broadcom	428	5.1%
Novell	380	4.5%
IBM	301	3.6%
Texas Instruments	276	3.3%
(Consultant)	223	2.6%
Freescale	182	2.2%
Linaro	170	2.0%
Samsung	162	1.9%
Google	150	1.8%
Wolfson Microelectronics	142	1.7%
Fujitsu	131	1.5%
Renesas Technology	100	1.2%
Oracle	82	1.0%
MiTAC	80	0.9%
Nokia	79	0.9%
(Academia)	73	0.9%

- Further testing and validation of upstream project to help track and fix bugs early
  - Linaro LAVA infrastructure and regular builds enable better tracking of bugs or regressions
    - http://www.linaro.org/linaro-blog/2011/08/12/lava-fundamentals/
  - Partners are also building and testing upstream code base and contributing patches/changes

Timing and Openness

- There are big variations in development cycles of new versions of the Architecture, CPUs, Platforms and Products
  - New versions of the architecture typically take a number of years whereas new CPUs, products and platforms work on shorter cycles
  - Parts of the cycles include critical periods where new IP and products are developed and software is used as validation exercise
    - Initial software support is usually closed as it is not mature and still too early to be exposed
- Provide suitable materials and frameworks to developers
  - Concise and comprehensive documentation
  - Development boards with suitable debug capabilities as well as software models
  - Community environment for questions and collaboration
- It is always a challenge to get appropriate dialogue between hardware designers and software developers
  - Hardware vs software mindset

Conclusion

- Issues developers are facing with ARM-based platforms are not really different to other architectures
  - Scaling is generally an issue for software, the growing popularity of ARM platforms exacerbate this

    but...
  - Companies are actively working with the community to improve support
  - This is also shows some success in the adoption of Linux and open source by companies!
- Timing is always challenging
  - Product cycle for consumer platforms is getting shorter and software support is increasingly needed much earlier in the cycle
  - Companies need to focus more on where there is real differentiation and where their real value-add is
  - Consumer product life cycle are relatively short compared to traditional PC/Server platforms
  - The high rate of changes in open source projects often make it challenging to track
- Enable effective collaboration between companies and the community
  - Engineering focused initiative like Linaro help working with existing upstream projects and communities
  - Adopt upstream practises used by respective communities and enable them with suitable documentation, tools and platforms