Linaro: A Year of Change

David A Rusling, CTO
April 2011
So, What is this Linaro Thing?
Linaro
What I Thought in 2009 / 2010

- ARM’s were moving out of ‘traditional’ embedded Linux
- Diversity is good, fragmentation is bad
- Lack of upstreaming is inefficient
- Lack of collaboration
- Not enough engineers working in ARM Linux

Therefore, we need to do something...
Why is ARM Embedded Linux Different?

- Product timelines are extremely short (order of a few months)
- Products are highly diverse (SoCs and Systems)
- Products generally not based on leading edge open source
- Perceived value in proprietary components
- OEMs, ODMs, SiP are caught in a trap
Linaro

- Linaro launched in June 2010
- Core engineering ~20
- Envisaged as an upstream engineering machine working on common embedded Linux problems
- Work in the open (see http://wiki.linaro.org)
- Spent the first six months building teams, fixing obviously broken things...
Social Engineering

True
- Gatekeepers have strong views
- Open source release times are driven by engineering needs / realities
- Open source demands deeper involvement

False
- Open source is difficult to deal with
- Open source doesn’t understand product time lines
- It’s cheaper with open source

Command and control versus distributed engineering
How Does Linaro Work?

- Technical agenda driven by the members
- Working groups
  - Work in upstream projects
  - Consolidation and innovation
- Platforms
  - Evaluation builds
  - Validation and benchmarking
- Landing teams
Linaro Teams

**Working Groups**
- Kernel Consolidation
- Toolchain
- Graphics
- Power Management
- Multimedia

**Platform Engineering**
- Validation & Benchmarking
- Evaluation
  - Builds Android, Ubuntu ...
- Infrastructure
  - Release management

**Landing Teams**
- Samsung LSI
- Texas Instr.
- ST-Ericsson
- Freescale
The First Six Months...

- May 2010 to November 2010
Summary

- Mostly about building teams
- Focused on two key areas:
  - Toolchain
  - Kernel
- Span out power management from kernel
- Most engineering was consolidation (fixing obvious broken things)
Biggest problem, first to tackle

Focus on ARMv7A, Thumb 2, Neon, SMP

Work in the upstream, but
- Current work lands in GCC 4.7
  - 4.7.0 ~April 2012
  - Distributions will use 4.7.1, ~July 2012 for their October 2012 release
- So, need consolidation builds...
Toolchain WG [2]

- **Scope**
  - Core toolchain, gcc, glibc, binutils, gdb
  - Remedial work on qemu, valgrind ...

- **Outputs**
  - Stable integration branches (current and preview, now 4.5 and 4.6)
  - Upstream patches and reviews
  - Monthly tarballs
Nov 2011 Technical Highlights

- Kernel Working Group
  - BSP review reports
  - Implemented ARM kernel security, devicetree fixes
- Multiple toolchain releases
  - 8 linaro-gcc releases (4.4 and 4.5)
  - 2 linaro-gdb releases (7.2)
  - cortexstrings
- Power Management tools
  - Powertop for ARM, powerdebug
- Hardware packs for all upstream supported boards
  - OMAP3, Versatile and a very basic mx51
  - BSP-based for other boards
- Toolchain adoption by Ubuntu, OxLabs, Openbricks, OE...
The Second Six Months...

- November 2010 to May 2011
Summary

- Created two new working groups, Graphics and Multimedia
- Landing teams started up for all members
- Kernel, Power Management and toolchain started to tackle difficult, long term problems
  - Device tree
  - Segmented memory
  - Vectorization
  - Hotplug
Kernel

- Segmented memory
- Device Tree
- Use the latest SoC features
  - Thumb 2, SMP, HIGHMEM, Neon, Cortex-A9
- Help upstream code
  - Review patches, BSPs
Graphics

- Advance ARM Graphics state of the art
- GLES backends for Cairo, Skia, meegotouch-compositor, chromium-wm and compiz
- Work with vendors and upstream to document a common ARM-standard acceleration framework
Multimedia

- Benchmarking and testing
- Tuning codecs (Neon, T2)
- OpenMax standardized components
- Frameworks (gstreamer)
Linaro Lava Labs

- Web dashboard
- Collect metrics on a variety of hardware
- Run individual test frameworks from various Linaro WGs
- Smoke tests, build tests, testsuites, benchmarks...
  - Stream, LTP, Open Posix Test Suite, gmpbench, gtkperf, x11perf, glmemperf, tiobench, qgears, es2gears, clutter-egl-es20, renderbench, glmark2-es2, UnixBench, Peacekeeper, Canvas Benchmark, GUIMark2, Biolab Disaster, Monster, Bbench v1.0, v8, quake3, coremark, sunspider, pybench, qa-regression-testing, LSB, MoonBat, Radiance test, Imbench, Kraken, Grafx Bot..
What Have I Learned So Far?

- That building a new engineering company from scratch (and donated engineers) is hard
- The ARM community can collaborate
- That things take longer than I want
- That there is still a lot to learn and do
- That open source is still wonderful
The Future...

- Beyond May 2011
Challenges

- Organizing ourselves better in open source communities such as kernel.org
- Boot architecture
- System wide power management (TTM, GEM, UMP other)
- Graphics – balancing proprietary and open source
- ‘Lumpy’ open source interaction
Graphics Acceleration

- Integrating GPU acceleration with distributions needs to be easier
  - Integration, redistribution and optimization
- The boundaries between open source and proprietary need to be more clearly drawn
  - Scope for moving these boundaries
  - As much a philosophical as a technical problem
Next Six Months

- Graphics
  - Embedded Memory Management
  - System-wide GPU Profiling
  - Compositing and OpenGL ES 2.0 Benchmarking

- Power Management
  - Consolidate around SMP Power Management
  - Drive forward Thermal management
  - Standardise measurement and tools
Next Six Months

- **Toolchain**
  - Improve GCC Performance
  - Widen the uptake of Linaro GCC through documentation, recipes, and binary releases
  - Standardize benchmarking
ARM Server

- ARM server hardware is coming
- The architectural details are published
- Initially, this is all about consolidation
- Longer term, this is about tuning for diverse system implementations
Contacting Us

- IRC #linaro on Freenode; Twitter: @LinaroTech
- linaro-{dev,kernel,toolchain}@lists.linaro.org
- launchpad.net/~linaro-community
- wiki.linaro.org, www.linaro.org
- Public (phone) confcalls
- May Linaro@UDS (Budapest), July Rally (Dublin)
www.linaro.org
Changes

- ARM ARM license
- ARM published its LPAE and Virtualization extensions very early in the cycle
- Consolidation
  - Device tree
  - Power management
  - Kernel
- Upstreaming early
  - Panda board
ARM Feature Development (5+ years)

- Prototype ideas
  - Understand hardware and software benefits and implications.

- Architect specifications
  - AEM models for architectural validation
  - Tool investigations
  - Fast ARM models for SW development.

- CPU design and implementations
  - RTL simulations, Palladium
  - FPGA systems for SW
  - Software and tools developments

- Open software developments

- SoC Delivery
  - Rev0 Silicon
  - Full speed SW development and tuning

- Product released
Software Development

Development

Synthesisable RTL

Host Simulation

FPGA Analysis

Maturity

alpha

beta

Early ACcess

launch & general access

Test Chip (s)

Next Rev.

Software platform is Palladium style. Very long iteration cycles and simulation platform a contended resource

Provides better execution times but again with low clock speeds and bandwidth limited good for early Alpha SW

Most suitable platform for SW development. The first opportunity to optimise SW at real clock speeds Beta SW onwards