ATLAS Nightly on ARM

Joshua Wyatt Smith

Supervisor: Andrew Hamilton

HEPPW 13/02/2015

DEPARTMENT OF PHYSICS
UNIVERSITY OF CAPE TOWN
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

Is it time for HEP to incorporate ARM CPUs into “online” and “offline” computing?

ToC

What is ARM?

Benchmarking on ARM

Software on ARM

} All in the context of "offline" computing

So far we've:

- benchmarked ARM processors (HPL, STREAM, PMBW),
- Run an E/p analysis using the Configuration Management Tool (CMT) and ROOT,
- built a dedicated PROOF (Parallel ROOT Facility) ARM farm.*

Currently:

- Building the entire ATLAS software stack (ATHENA) on ARM

*In collaboration with Wits

Intro

What is ARM?

Embedded boards

Table 3.1: The different setups with key features.

Setup	Processor	Cores	RAM	Cache	FPU	OS
Cubietruck	AllWinner A20, 1.2GHz	A7 dual core	2GiB DDR3	512 KiB L2	VFPv4	Archlinux, hard float
Wandboard-Quad	Freescale i.MX6 Quad, 996MHz	A9 quad core	2GiB DDR3	32KiB L1, 1 MiB L2	VFPv3	Archlinux, hard float
ArndaleBoard-K	Samsung Exynos 5250, 1.7GHz	A15 dual core	2GiB DDR3	32 KiB L1, 1MiB L2	VFPv4	Fedora 19, hard float
Jetson TK1	NVIDIA Tegra-K1, 2.3GHz	A15 quad core-plus-one	2GiB DDR3	2 MiB L2	VFPv4	Fedora 21, hard float
Hep306	Intel® Core i7-2600, 3.4GHz	quad core	16GiB DDR3	256KiB L1, 1MiB L2, 8MiB L3	-	Scientific Linux CERN 6

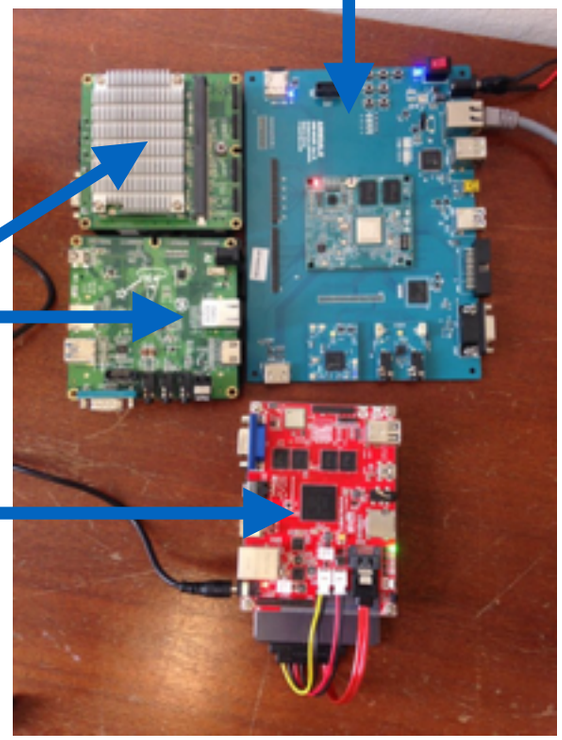
For reference

NVIDIA Tegra-K1
A15 Quad Core

Samsung Exynos 5250
A15 Dual Core

Freescale i.MX6
A9 Quad Core

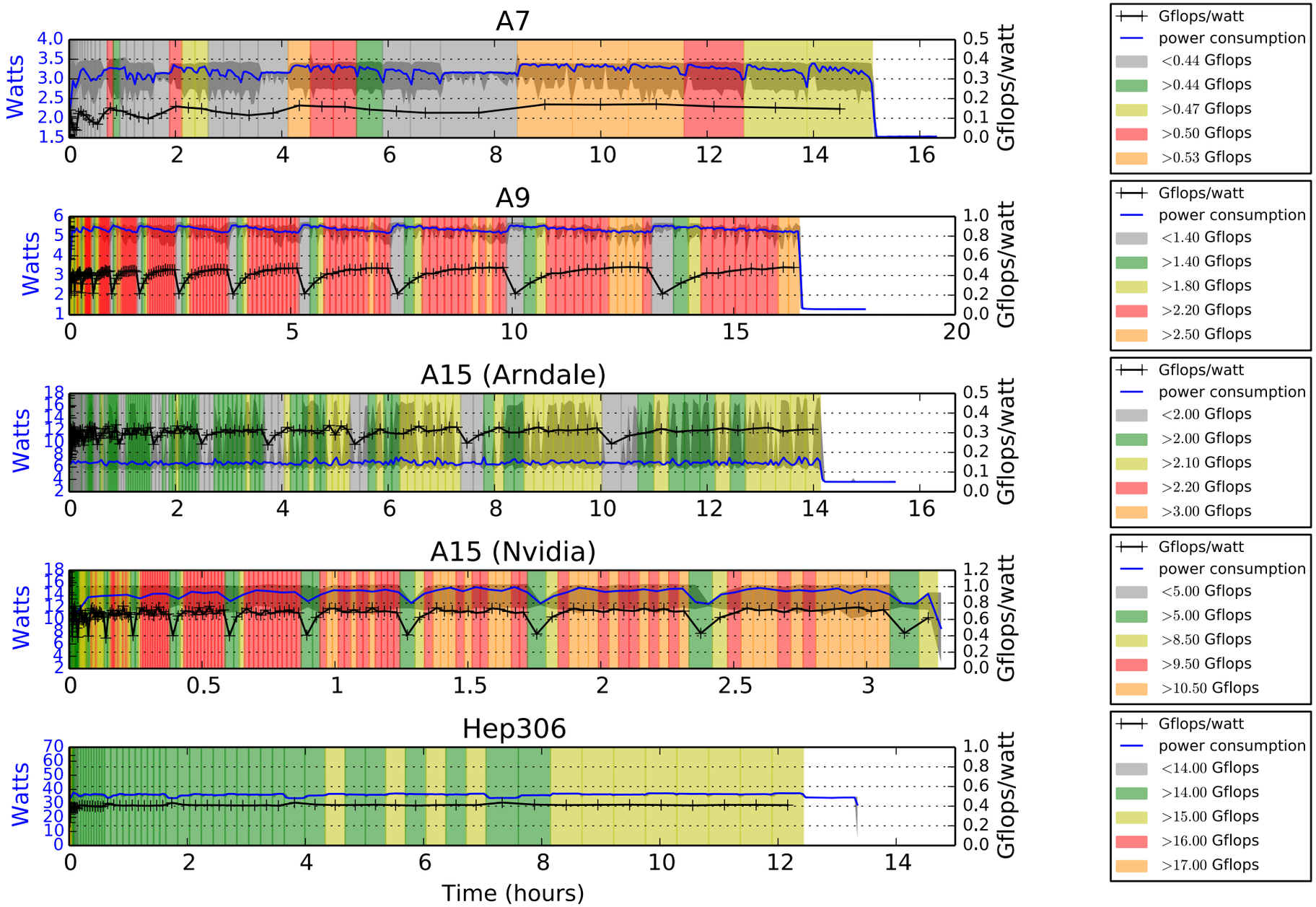
AllWinner A20
A7 Dual Core



High Performance Linpack

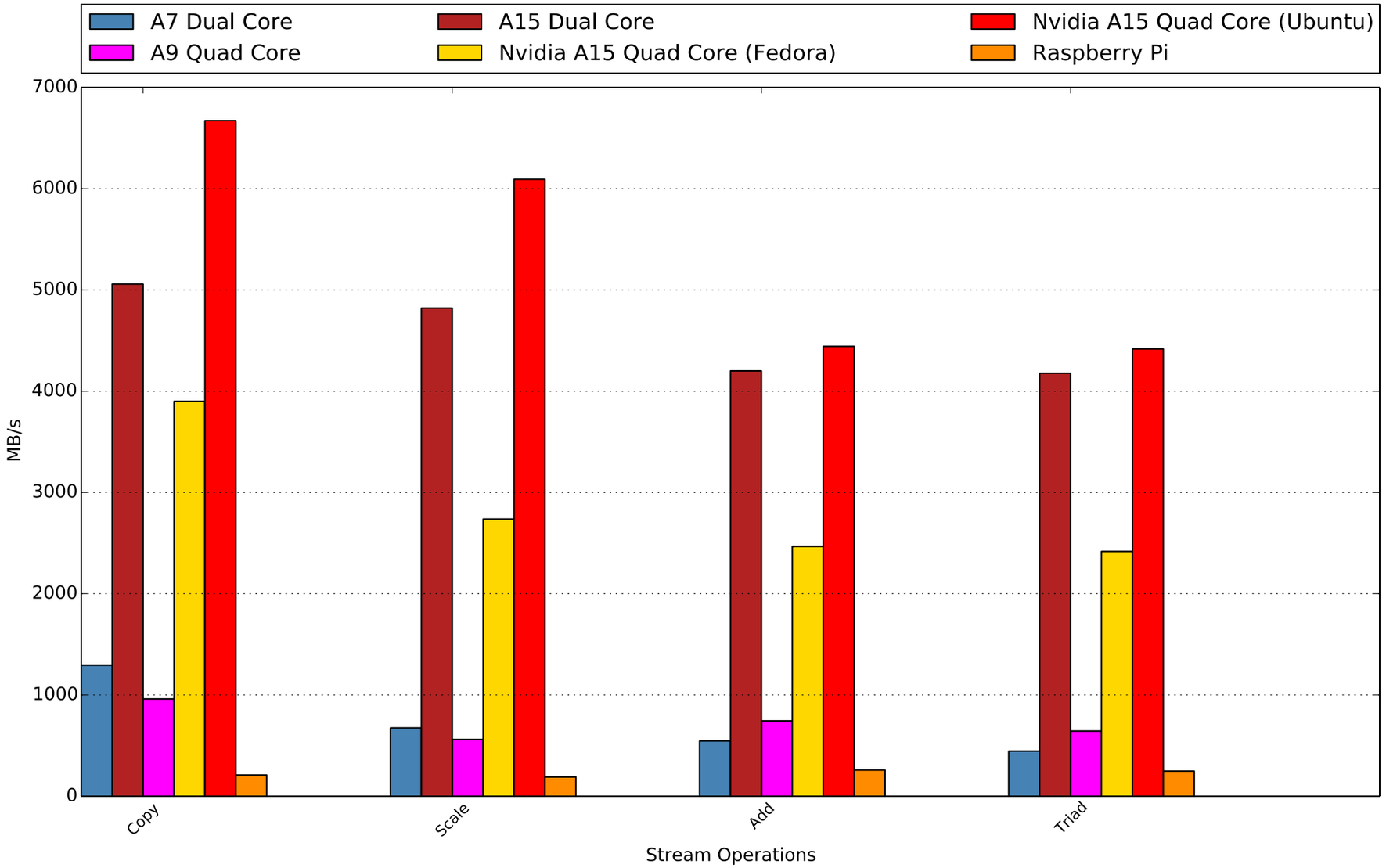
Multimeter
(CPU & all peripherals)

Intel Power Gadget
(only CPU)



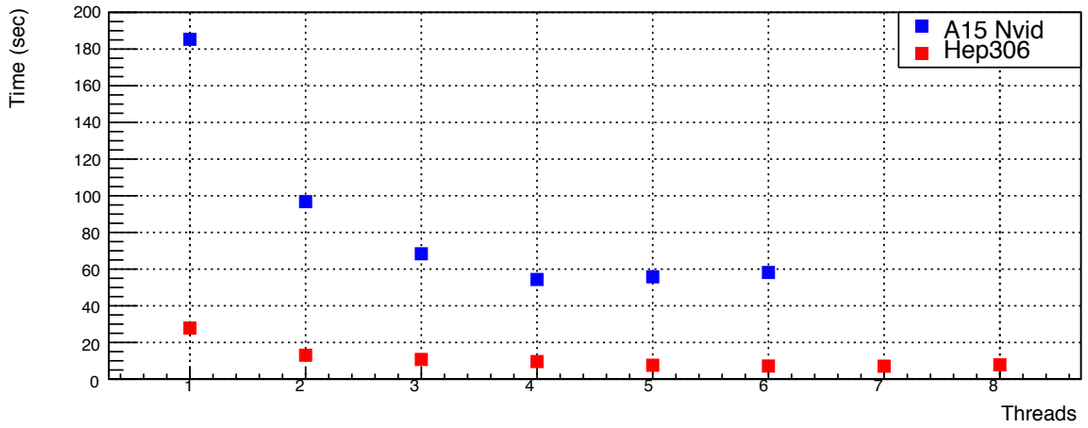
- Increasing matrix and block size (from left to right)

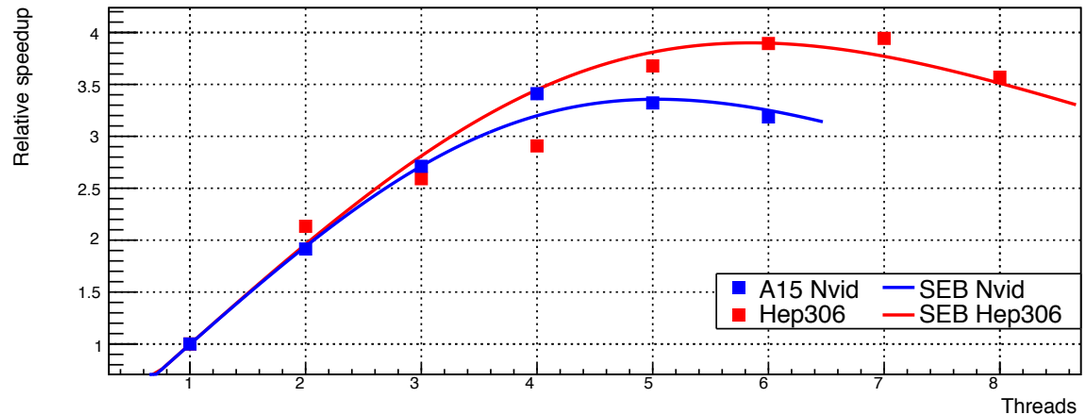
Stream benchmark

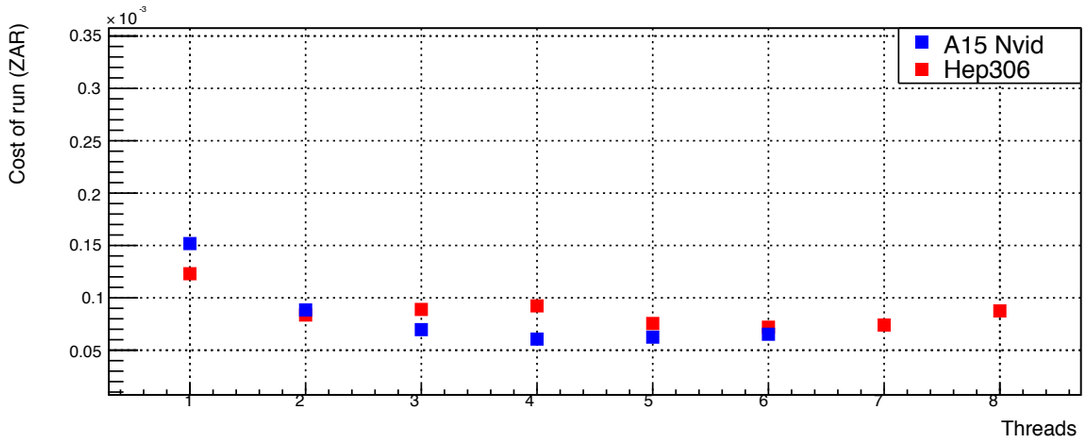


Serial memory bandwidth. Choice of OS clearly plays a large role in performance for A15 quad-core.

E/p analysis







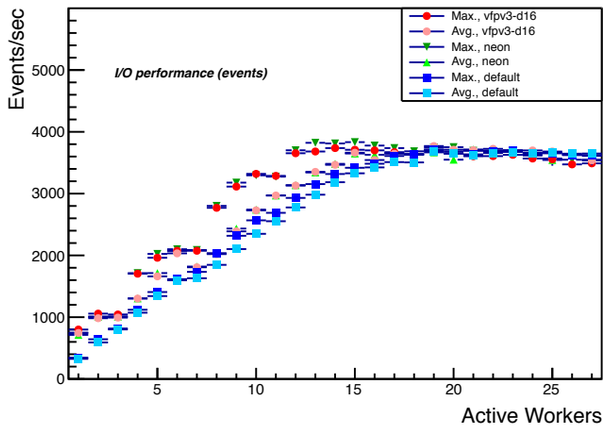
The E/p analysis for the TileCal in ROOT. First practical use of CMT on an ARM board.

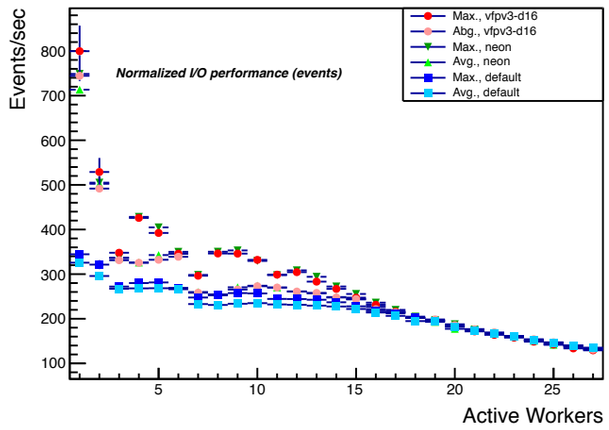
SEB=Strong Scaling, Embarrassingly parallel, Blocking network:

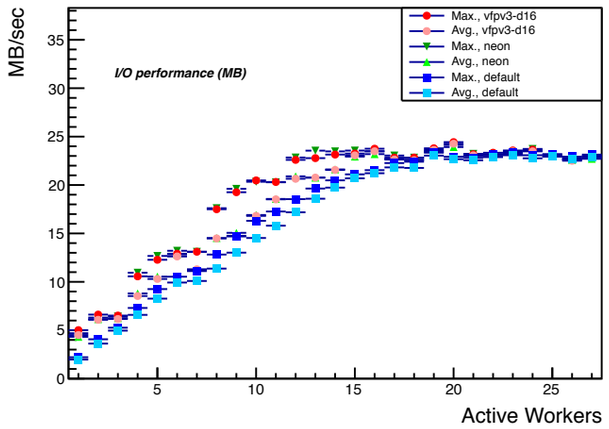
$$\frac{1}{s+\frac{1-s}{n}+n^2\omega}$$

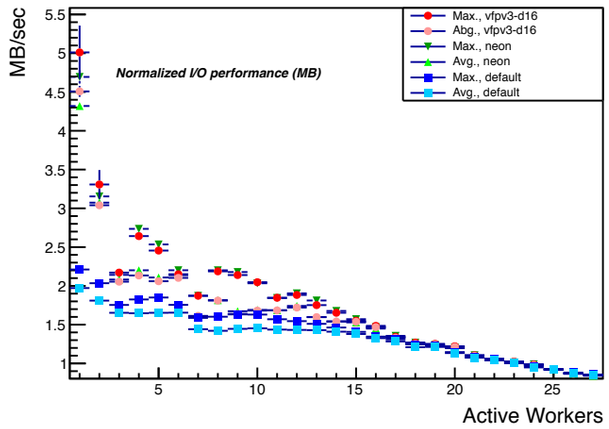
s=serial portion
ω=communication overhead
n=number of processors

A PROOF cluster









I/O benchmark
(NFS causes a bottleneck)

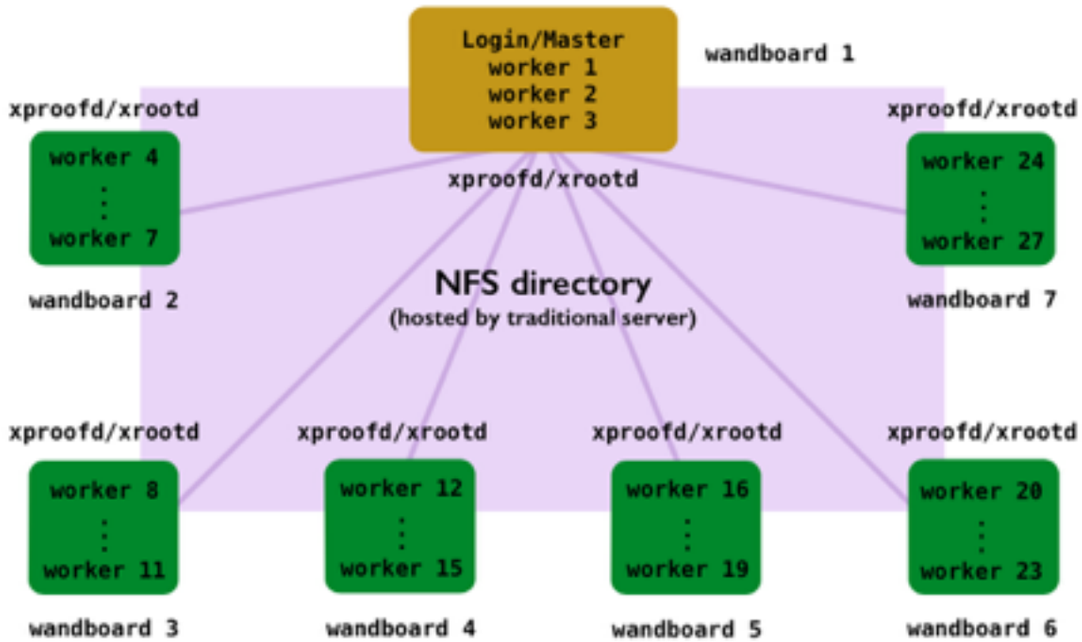
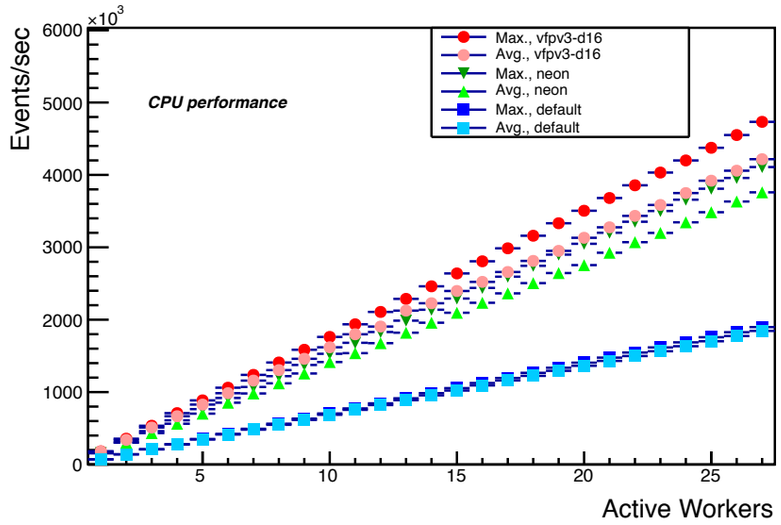
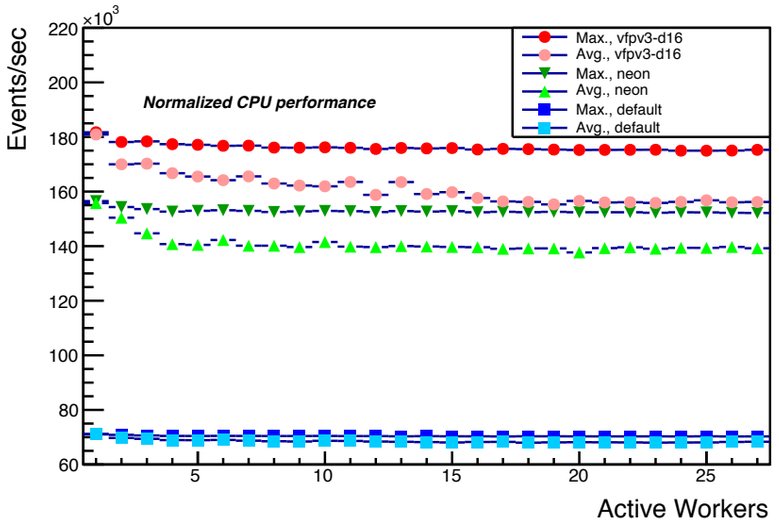


Figure 4.6: The setup of the dedicated PROOF cluster at Wits.

CPU benchmark
(very scalable)



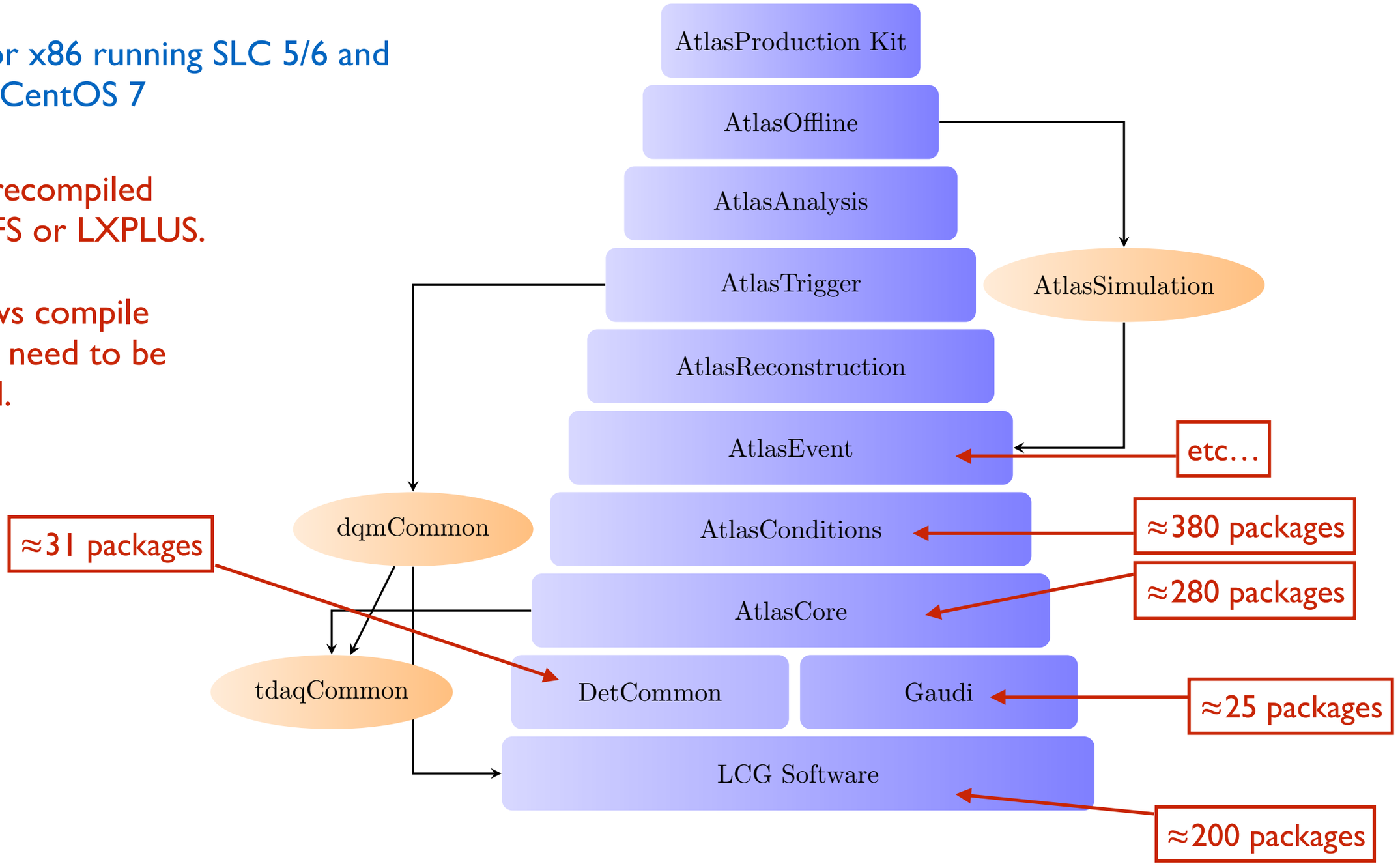


ATLAS Software on ARM

An ATLAS kit (ATHENA) has over 2000 packages
≈ 6.5 Million lines of code

All of it built for x86 running SLC 5/6 and more recently CentOS 7

- Can't use precompiled binaries in AFS or LXPLUS.
- ARMS throws compile errors which need to be fixed by hand.



CMake/CMT

- Package and software managers,
- Both require knowledge of new non-trivial syntax.

CMake

- Minimal dependencies,
- **Cross compilation possible,**
- Generates a cache, so is fast,
- Easily supports parallel builds,
- steer project with a CMakeLists.txt,
- Lots of documentation!
- ATLAS software being migrated to CMake.

CMT

- Still the preferred method,
- **Cross compilation NOT possible,**
- slooooooow,
- Is difficult to learn,
- Only capable of a small amount of parallelization,
- Each individual package needs ambiguous “requirements" file.

ATLAS Software on ARM

Introducing: ANA (ATLAS Nightly on ARM)

- builds software at night, ready for debugging the next day. (Based of ATLAS NICOS).

- CMT environments and directory styles,
- Populate directories,
- Create project.cmt and "steering requirements file",
- Build everything or specific project.
- Apply patches? (Yes/No)

https://github.com/jwsmithers/AtlasOfflineBuild-framework

61 commits | 1 branch | 0 releases | 1 contributor

branch: master · AtlasOfflineBuild-framework / +

more patches added

jwsmithers authored 2 days ago — latest commit 5ddc832a4e

InstallArea	add InstallArea folder which disappeared at some point	2 months ago
PostInstall	more patches added	12 days ago
PreInstall	Eigen and readline sources available in PreIsntall folder	2 days ago
SVN	SVN directory disappeared for a bit, now it's back.	2 months ago
Testing	patch updates	2 months ago
fileChanges	file patches and file changes	8 days ago
logs	Some patches for AtlasConditions	2 days ago
patches	more patches added	2 days ago
scripts	Environment changes	5 days ago
BuildAtlasOffline.sh	more patches added	2 days ago
Environment.sh	Patches added, genCLIDDB finally compiles	14 days ago
Notes.txt	cleaned up a few unnecessary scripts	22 days ago
Projects.txt	Major updates	3 months ago
README.txt	cleaned up a few unnecessary scripts	22 days ago
WelcomeMSG.txt	adding cleaning, patching and file change capabilities...	3 months ago

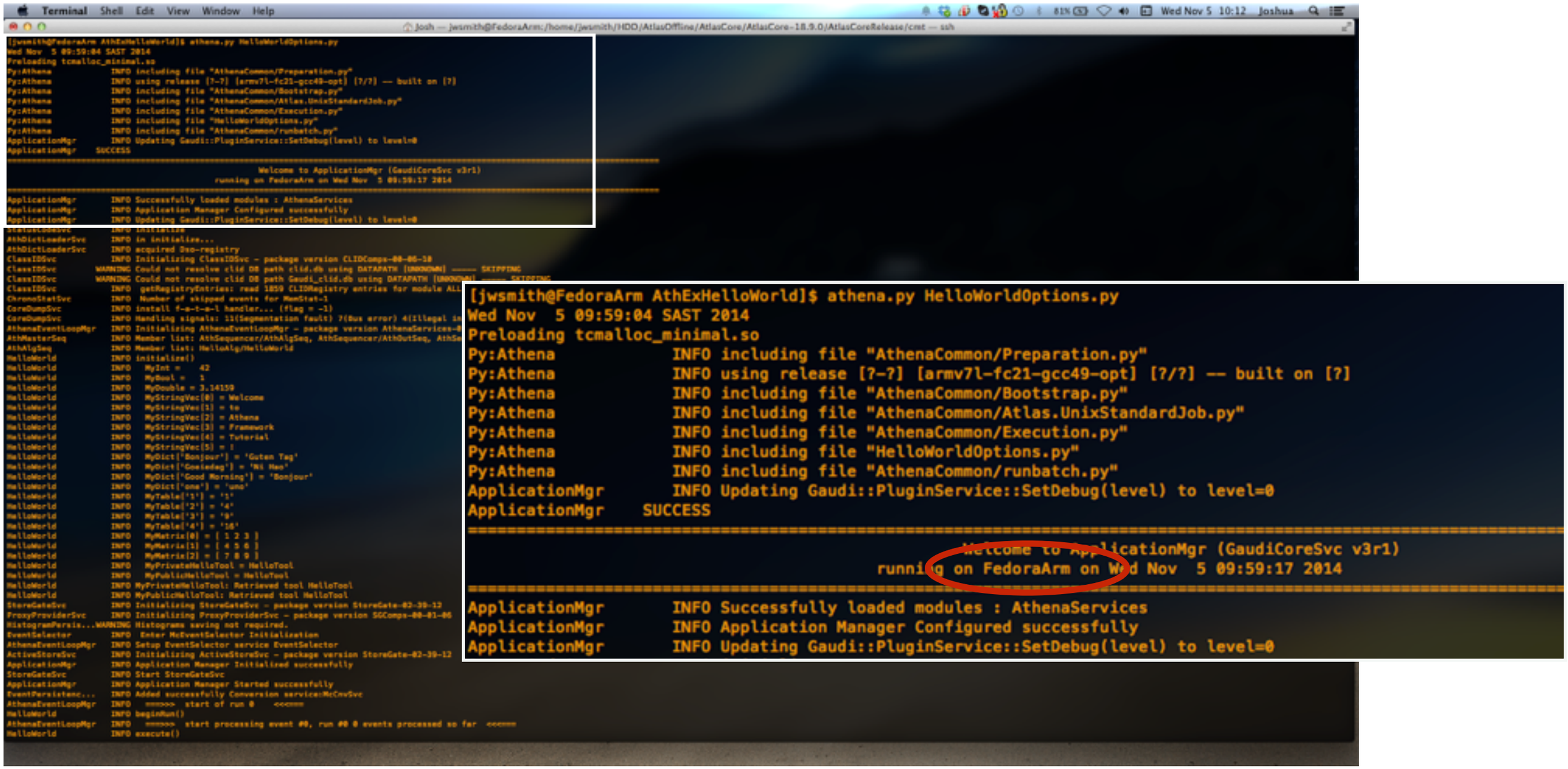
README.txt

```
########################################################################
########################
#                               How to build ATLASOFFLINE for ARM
#
#                               Joshua Wyatt Smith
#
#                               joshua.wyatt.smith@cern.ch
#
########################################################################
########################

This describes how to buld AtlasOffline using this framework..

This assumes that you have downloaded and installed CMT, LCG, Gaudi,
CORAL and COOL. Make sure
that SVN is set it up so that no password is required to check out
packages from svn.cern.ch.
To do this see
https://confluence.slac.stanford.edu/display/Atlas/Avoiding+repeating+pas
swords+for+CVS+and+SVN


To configure everything simply run
>> source BuildAtlasOffline.sh
```

ATLAS Software on ARM

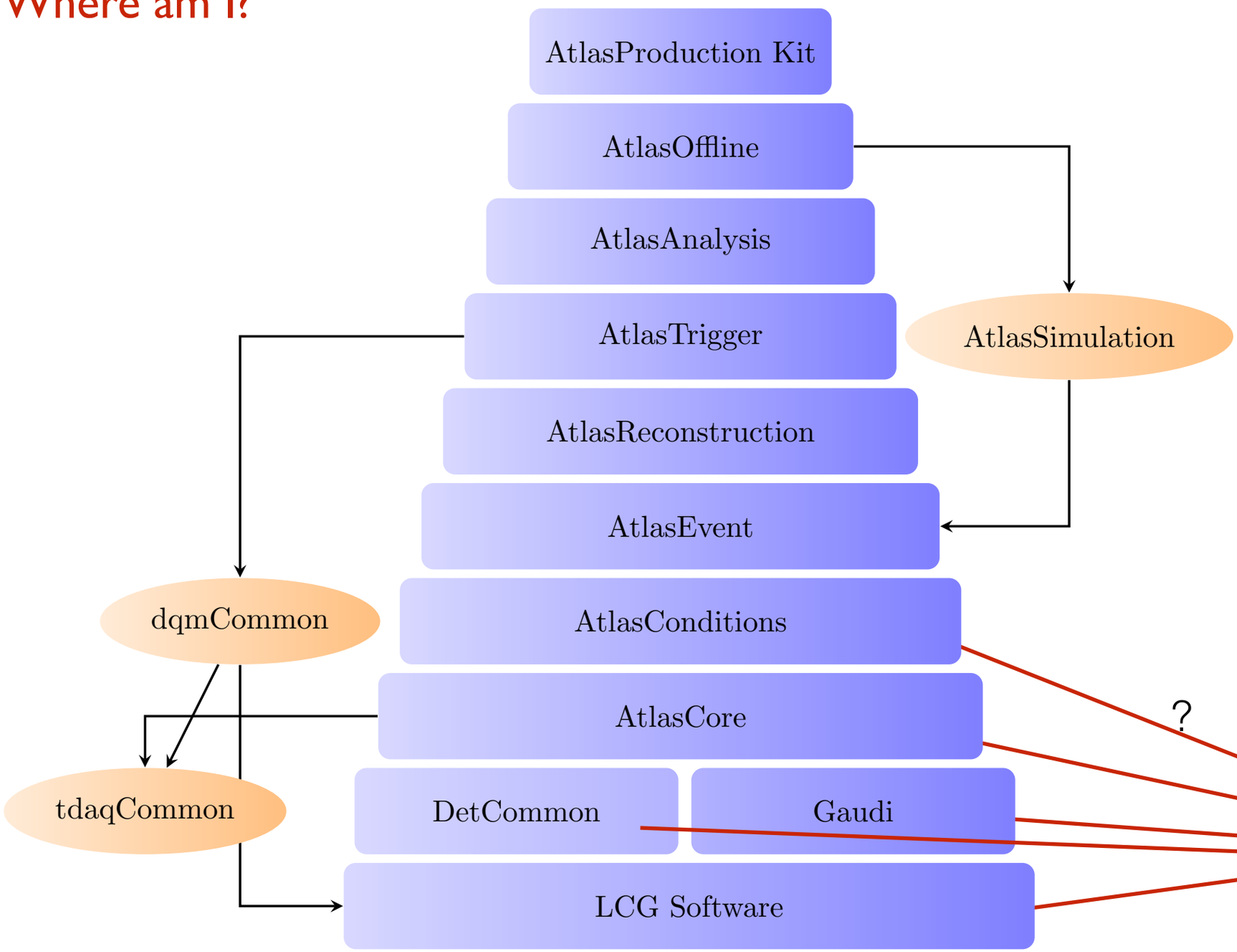
Recently:

First ever ATHENA HelloWorld Example on ARM!



ATLAS Software on ARM

Where am I?



Conclusions

- ARM processors are still slow relative to Intel/AMD. However, they are improving quickly!
- ARM processors use much less power than traditional CPUs.
- For the most part, code written for Intel is capable of being compiled on ARM. It's just that someone has to do it.*
- ARM is a enormous company and offer a lot of freedom when creating SoCs.

* However, finding a lot of "quirks" in code that should work.

Acknowledgements

DEPARTMENT OF PHYSICS
UNIVERSITY OF CAPE TOWN
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

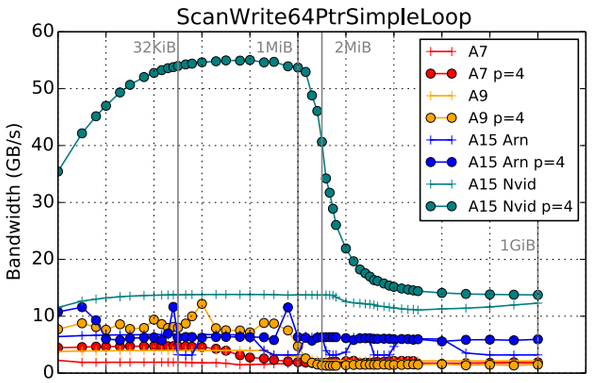
National Research Foundation

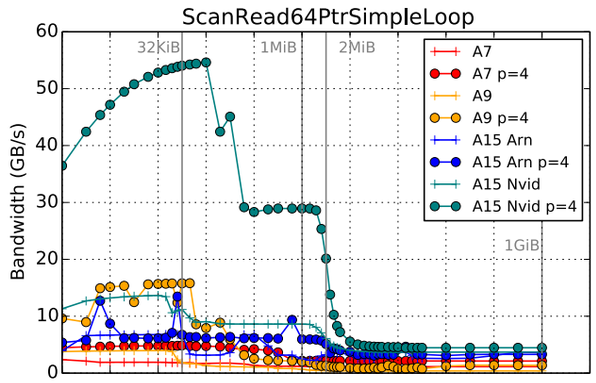
CERN

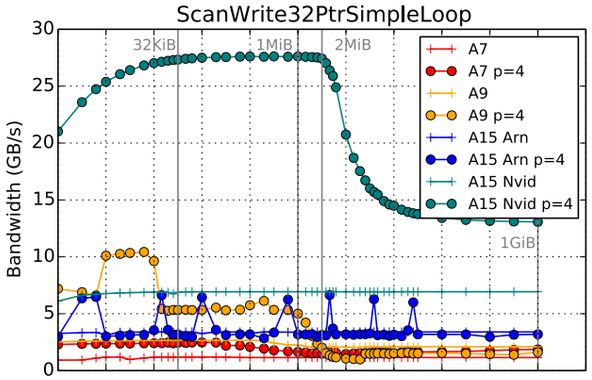
UNIVERSITY OF THE WITWATERSRAND JOHANNESBURG

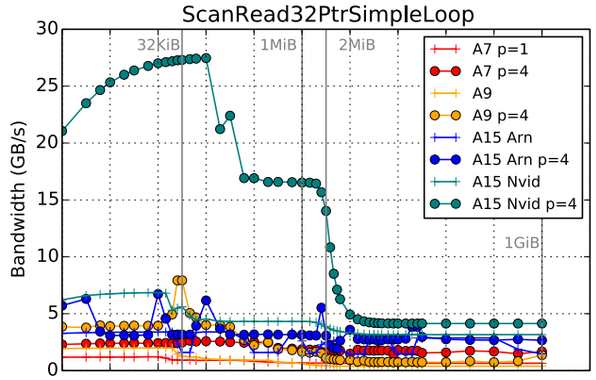
Thank you.

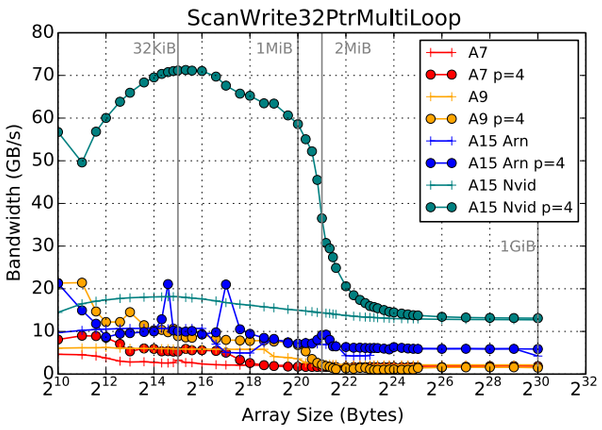
Backup

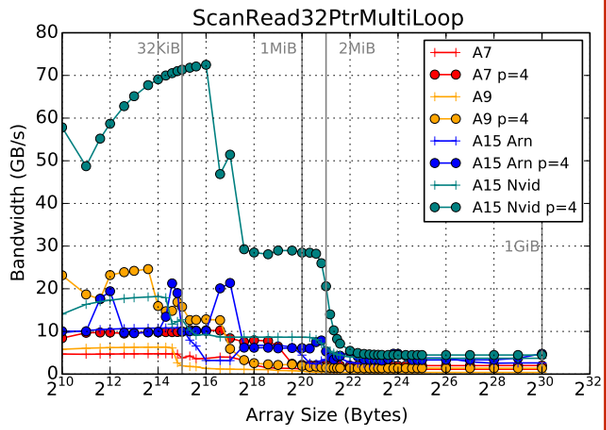


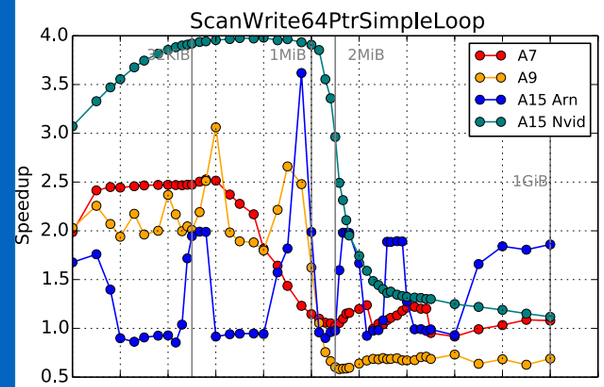


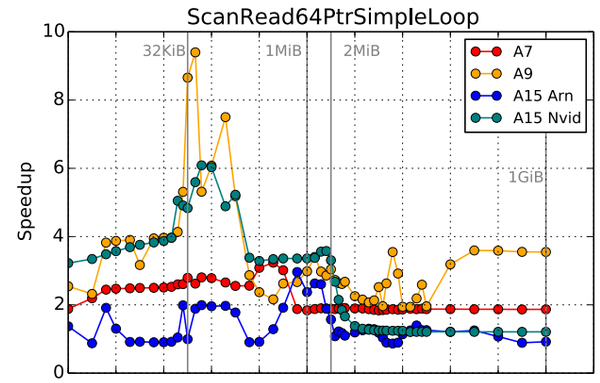


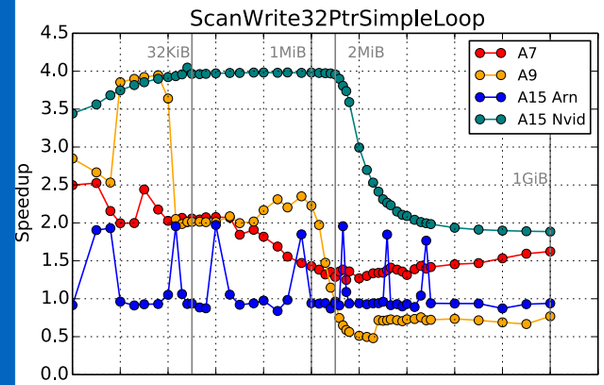


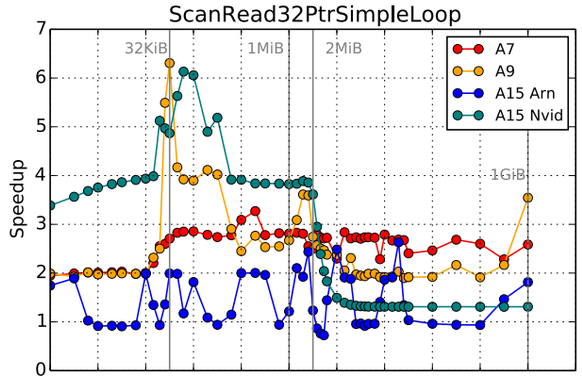


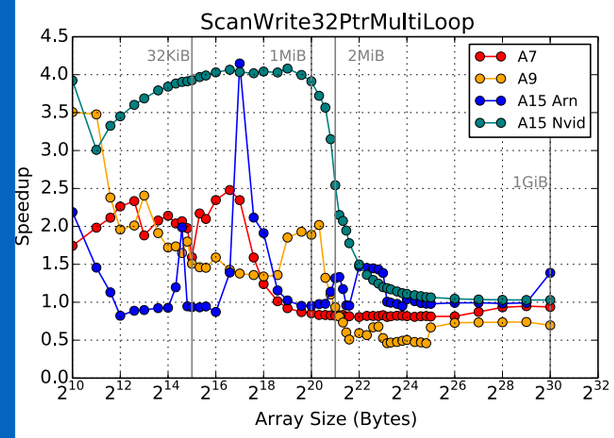


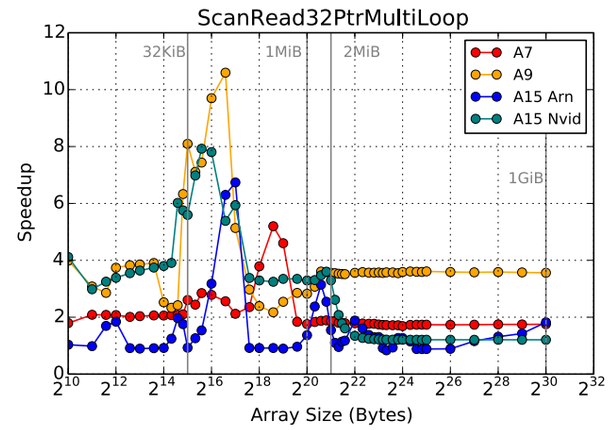












Different scenarios and tests for parallel memory bandwidth on ARM boards.

The speedup from 1 thread to 4 threads for each of the tests.