High-availability on-site deployment to heterogeneous architectures for Project 8 and ADMX

November 7, 2019

Benjamin LaRoque
The ADMX Collaboration gratefully acknowledges support from the US Dept. of Energy, High Energy Physics DE-SC0011665, DE-SC0010280, & DE-AC52-07NA27344. ADMX also receives support from the LLNL and PNNL LDRD programs and R&D support from the Heising-Simons institute.
The Project 8 Collaboration

Case Western Reserve University
• Laura Gladstone, Benjamin Monreal, Yu-Hao Sun

Harvard-Smithsonian Center for Astrophysics
• Sheperd Doeleman, Jonathan Weintroub, André Young

Johannes Gutenberg-Universität Mainz
• Sebastian Böser, Christine Claessens, Martin Fertl, Michael Gödel, Alec Lindman

Karlsruher Institut für Technologie
• Thomas Thümmler

Lawrence Livermore National Laboratory
• Kareem Kazkaz, Lucie Tvrznikova

Massachusetts Institute of Technology
• Zachary Bogorad, Nicholas Buzinsky, Joseph Formaggio, Evan Zayas, Talia Weiss

Pacific Northwest National Laboratory
• Mark Jones, Benjamin LaRoque, Erin Morrison, Noah Oblath, Malachi Schram, Jonathan Tedeschi, Brent VanDevender, Mathew Thomas, Maurio Grando

Pennsylvania State University
• Luiz de Viveiros, Timothy Wendler, Andrew Ziegler

University of Washington
• Ali Ashtari Esfahani, Raphael Cervantes, Peter Doe, Eric Machado, Elise Novitski, Walter Pettus, Hamish Robertson, Leslie Rosenberg, Gray Rybka

Yale University
• Karsten Heeger, James Nikkel, Luis Saldaña, Penny Slocum, Pranava Teja Surukuchi, Arina Telles

This work was supported by the US DOE Office of Nuclear Physics, the US NSF, the PRISMA+ Cluster of Excellence at the University of Mainz, and internal investments at all collaborating institutions.
But what of the on-site cloud?

Project 8
- 3 Dell blade servers
  - 10 Gb direct data link
- 1 ROACH2 blade + ADC
  - 10 Gb direct data link
- 5 “gaming hardware”*
- 2 Dell desktops
- 4 Raspberry Pi
  - GPIO attached instruments

ADMX
- 2 Dell blade servers
  - PCIe Digitizers
- 3 “gaming hardware”*
- 1 Dell desktop
- 2 Raspberry Pi
  - GPIO attached instruments

*this means separately purchased motherboards, processors, etc. with parts typically used for home computer games
Traditional challenges for projects at this scale

• Few dedicated development platforms; anything used for testing becomes the production system once initial tests pass
• Hardware largely unmatched
• No pure worker nodes, many have specialized hardware
• Application and dependency installation is often done interactively and over time (system state is hard to reproduce)
• Application deployments are often in user space (terminal multiplexers), cron jobs, or possibly custom system daemons
• Updates, especially hotfixes, often introduce build/deployment bugs on other systems
Opportunities for success

• Use Raspberry Pi for integrated controls

• Leverage container and CI/CD ecosystems to ensure reproducible builds/installations and execution environments

• Leverage cloud computing ecosystem (Kubernetes and helm) to ensure consistent and version-controlled deployments
Why use a Raspberry Pi?

- Full Linux environment: can run whatever software we need, so long as it compiles for arm
- Easy GPIO programming: access to both logic outputs and I2C programming
- Variety of ADCs, sensors, etc. with existing libraries and user community
- Form factor and price makes it easy to integrate and deploy
Automating building and testing across architectures

- Containers provide consistent and reproducible environment
- OCI Image Index unifies deployment target across architectures
- Continuous Integration (we use travis-ci.com) ensures all architectures are always built
- QEMU (qemu.org) allows us to produce arm-compatible build artifacts on x86 hardware
K8s & helm

• Using k8s “Deployment” objects provides process monitoring and restarts

• NodeSelectors and Taints account for specialized hardware resources

• Resource definitions can specify an Image Index, allowing non-specialized workloads to float between arm and x86 hardware

• A few charts with many releases provide a clear process for upgrades and enables keeping fully version controlled deployment configurations.
Bonus: free monitoring tools

- The k8s ecosystem already maintains Prometheus and Grafana for cluster monitoring (below)
- Same tools also monitor sensors logs (right)
Summary

• Medium-scale experiments often operate only a handful of nodes, but with each having distinct hardware configuration and/or specialized resources

• Cloud computing practices (containerization, configuration as code, etc.) are highly valuable in smaller on-premises environments

• Leveraging the cloud ecosystem also gives access to monitoring and alerting tools, backed by large active communities