National Synchrotron Light Source II

Brookhaven National Laboratory

U.S. DEPARTMENT OF ENERGY

EPICS Deployment at NSLS-II

EPICS Collaboration Meeting

Anton A. Derbenev, Group Lead, NSLS-II DSSI

September 22, 2022

The Change of Landscape

NSLS-II Controls Package Repository

This repository contains installable software packages in the Debian Linux format (.deb)

Packages in this repository were developed by EPICS community members and are used

- NSLS-II was a Debian shop since construction
- We were building, testing, publishing packages for debianized code – collaboration effort at **github.com/epicsdeb** and internal
- Some time ago, our OS standard has been changed to RHEL
- Debian repo is still up but deprecated at **epicsdeb.bnl.gov**
- We did a massive migration of our IOCs – which required new packaging!

From .deb to .rpm

- Debian packaging approach: follow strict practices, tight about signing and versioning, base and modules are split, multiple architectures, manage dependency trees
- RHEL packaging approach: make it work good fast easy
- **Jenkins** automates build and package:
  - **A specialized in-house build tool** pulls, configures, builds, restructures all required modules and base using customizable build configuration (heavy lifting is done here)
  - **rpmbuild** produces a single .srpm from a relatively unsophisticated .spec (artifacts are pre-structured)
  - Packaging pipeline builds and publishes to the **internal repo** (with basic versioning)
- **RH Satellite/Automation Platform** handles install and update:
  - Centralized, **automated install** on relevant machines (IOC servers, CA client workstations)
  - The package comes from internal repo, all installs go in **/usr/lib/epics**
  - From IOC perspective, very similar to how it was done on Debian (none or minor reconfiguration needed)

Looking Forward

- More flexible customization and versioning of the package, consider split in a collection of base/module/driver packages
- Replace Jenkins with GH Actions, do more advanced testing (e.g., with a “virtual beamline”)
- Externally accessible repo – perhaps from **epicsrpm** collaboration effort?
- Automated deployment pipeline for IOCs