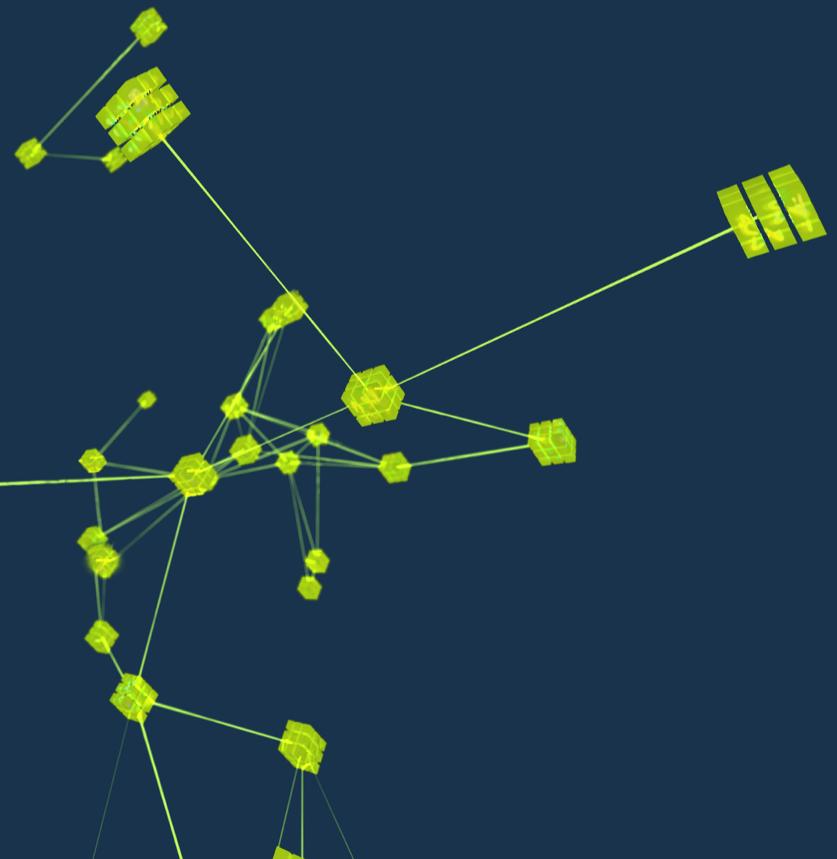


EPICS Continuous Integration: Status and Support

Build and test infrastructure:

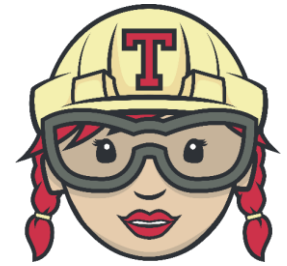
- Number of unit tests in EPICS Base is increasing:

  3.14: **2.6k** → 3.15: **8.5k** → 3.16: **9.2k** → 7.0: **23k**

- Jenkins instance at APS (https://jenkins.aps.anl.gov/)
  - C++ builds
  - Only master branch
  - Host builds: Linux, MacOS, Solaris, Windows
  - Cross builds to many targets (VxWorks and RTEMS)

- Jenkins instance at PSI (*behind institute firewall*)
  - C++ builds
  - Linux host build
  - Cross builds to many VxWorks targets
  - Pull request builds

Build and test infrastructure:

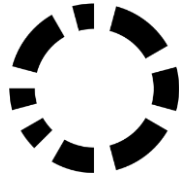


- Travis-CI (https://travis-ci.org/epics-base/epics-base)
  - Host builds on Linux (gcc, clang) and MacOS
  - Cross builds for RTEMS, Windows (WINE)
  - Pull request builds
- AppVeyor (https://ci.appveyor.com/project/epics-base/epics-base)
  - Host builds on Windows (all branches and pull requests)
  - Compilers: VS 2010, 2012, 2013, 2015, 2017, Cygwin, MinGW
  - 32bit or 64bit architecture
  - DLL or static build
  - Debug or optimized build
  - Full matrix: 50 builds, taking ~13 hours (only one builder)
- Evaluating: Azure Pipelines (https://dev.azure.com/epics-base/)
  - Five parallel builders

Still on the wishlist:

- Test coverage
- Static code analysis

Maybe use/integrate a tool like Codacy (https://app.codacy.com/)?



Common CI support for EPICS modules

- Complex helper scripts for Travis-CI & Co. as a Git submodule
- Easy to use:
  1. Add ci-scripts as a Git submodule to your EPICS software module
  2. Copy an example .yml configuration and adapt to your needs
  3. Activate your repository on the CI platform
- Travis-CI: Linux, MacOS, RTEMS-cross and MinGW-cross
- Git submodules always use a specific commit:
  ci-scripts developments don't break your builds

https://github.com/epics-base/ci-scripts

- Travis-CI (Linux, MacOS, RTEMS-cross, WINE-cross) works
- AppVeyor (native Windows builds) is next

EPICS