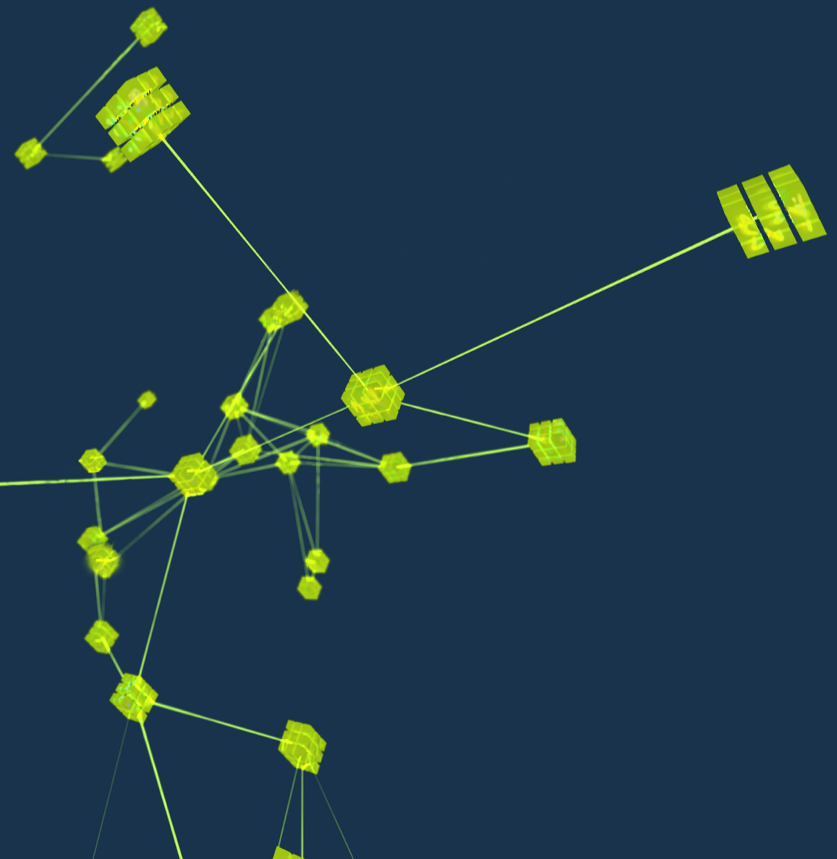

The EPICS Training VM

Ralph Lange

Basic Ideas

- Reproducible platform for hands-on training sessions
  - Training courses are done a few times a year
  - Demonstrations and hands-on exercises are often part of the training
  - Needs to be easily maintainable in a shared fashion
- Virtual machine based setup
  - Runs on may host platforms (Linux, Windows, old Macs)
  - OVA applications can be distributed (but they are huge)
  - Trainees start with a working setup (don’t waste time with the installation)
- Ansible based installation
  - Different parts (training sessions) are available as independent roles
  - Automated installation on the “empty” VM for a specific training course
  - Ansible content are text files (yaml) under Git management

EPICS

The Virtual Machine

- Based on Oracle VirtualBox (7.0)
  - Freely available virtualization platform
  - Good experiences at many labs
- Using Rocky Linux 9.3
  - Best knowledge and most existing Ansible code is based on RHEL
  - Easy installation from Rocky distribution images
  - Training-VM could be extended to also work on Debian-based distros
- Generic “EPICS Developer” (epics-dev) User Account
  - Best practice:
    Read-only shared installation, development under a regular user account
- Personalize your VM (or use your own)!
  - Make yourself comfortable:
    Create your own user, install your favourite IDE and tools, use your own VM

EPICS

The Training-VM Installation

- From sources
  - For the EPICS related parts, the virtual machine contains all source code and all knowledge how things are built and set up
- Simple, following best practice
  - Avoid unnecessary complexity (e.g., containers)
  - Show how a minimal EPICS set-up would look like
- Reproducible
  - Under configuration control (git) and fully scripted
  - Easier to support trainees
- Modular
  - Easy to adapt to specific training events
  - Easy to maintain collaboratively

EPICS

The Training-VM Installation

- Ansible roles control the scope of the installation
  - *epics-modules*
    C/C++ EPICS Support modules: Base, ASYN, Sequencer, AreaDetector, …
    Under `/opt/epics`
  - *epics-tools*
    Java 17, Maven: from installation downloads for a fixed and portable install
    Phoebus: from source
    Under `/opt/epics-tools`
  - *docker*
    Podman and podman-compose: everything to run groups of containers
  - *bluesky*
    Containerized setup for the Bluesky training session
    In container images and under `/opt/bluesky`

Configuring the Training-VM Installation

- The central configuration file is `~/training/local.yml`
  - Enable/disable roles as you need them
  - Define the list of EPICS modules that the *epics-modules* role will install
  - Set http/https proxies (if you need to)
  - Define the settings for a corporate firewall (if you need to)

Applications on the Training-VM

- Under `~/training`
- Apps roughly follow the training sessions
  - The directories under `~/training` mostly contain regular EPICS Modules
  - Configuration against the training VM setup is done through a *single file*, `~/training/RELEASE.local`
  - To compile application modules, follow the usual EPICS approach:
    ```
    $ cd <TOP>
    $ make
    ```
  - To run IOCs, similarly:
    ```
    $ cd <TOP>
    $ ( cd iocBoot/ioc<ioc-name>; \
    ../../bin/linux-x86_64/<IOCbinary> st.cmd )
    ```

EPICS

Update the Training-VM Installation

- The `update.sh` script gets your training-vm up-to-date.
  Call it (best from the `~/training` directory) to:
  - *Update the Ansible configuration and the applications*
    `git pull` the appropriate branch of the `training-collection` meta-repo
    (The training event name is configured in `/etc/epics-training`)
  - *Get the Ansible collections*
    Install the required Ansible collections (equivalent to libraries)
  - *Run Ansible*
    Re-run main Ansible playbook to update the installation
    Ansible is target state-based
    Tasks have been written to minimize run time when nothing needs to be done
- Run it before a session to catch last-minute updates by the trainers
- If you only want to update the application part, it suffices to run
  `git pull --recurse-submodules`
  in the `~/training` directory

EPICS