Building and Packaging **EPICS** Modules with **Conda**

B. Bertrand, A. Harrisson, ESS Lund Sweden, October 7, 2019

Abstract

Conda is an open source package, dependency and environment management system. It runs on Windows, macOS and Linux and can package and distribute software for any language (Python, R, Ruby, C/C++...). It allows one to build a software in a clean and repeatable way. EPICS is made of many different modules that need to be compiled together. Conda makes it easy to define and track dependencies between EPICS base and the different modules (and their versions). Anaconda's new compilers allow conda to build binaries that can run on any modern linux distribution (x86_64). Not relying on any specific OS packages removes issues that can arise when upgrading the OS. At ESS, conda packages are built using gitlab-ci and pushed to a local channel on our Artifactory server. Using conda makes it easy for the users to install the EPICS modules they want, where they want (locally on a machine, in a docker container for testing...). All dependencies and requirements are handled by conda. Conda environments make it possible to work on different versions on the same machine without any conflict.

Conda Concepts

- **Conda Package**
  a compressed tarball file (.tar.bz2) or .conda file that contains a collection of files to be installed and some metadata.
- **Conda Channel**
  a repository of conda packages. Conda packages are downloaded from remote channels, which are URLs to directories containing conda packages.
- **Conda Environment**
  a directory that contains a specific collection of conda packages. A conda environment is a virtual environment that is completely isolated from other environments. Each environment includes the sub-directories /bin, /etc, /include, /lib, /share, mimicking the Linux Filesystem Hierarchy.
- **Conda Recipe**
  a directory that contains at least a meta.yaml file describing the package

Conda EPICS Module Recipe

```
mymodule-recipe/
├── .gitignore
├── .gitlab-ci.yml
├── LICENSE
├── README.md
├── recipe
│   ├── build.sh
│   └── meta.yaml
└── src
    └── Makefile
```

```yaml
---
include: 'https://gitlab.esss.lu.se/ics-infrastructure/gitlab-ci-yml/raw/master/E3CondaBuild.gitlab-ci.yml'
```

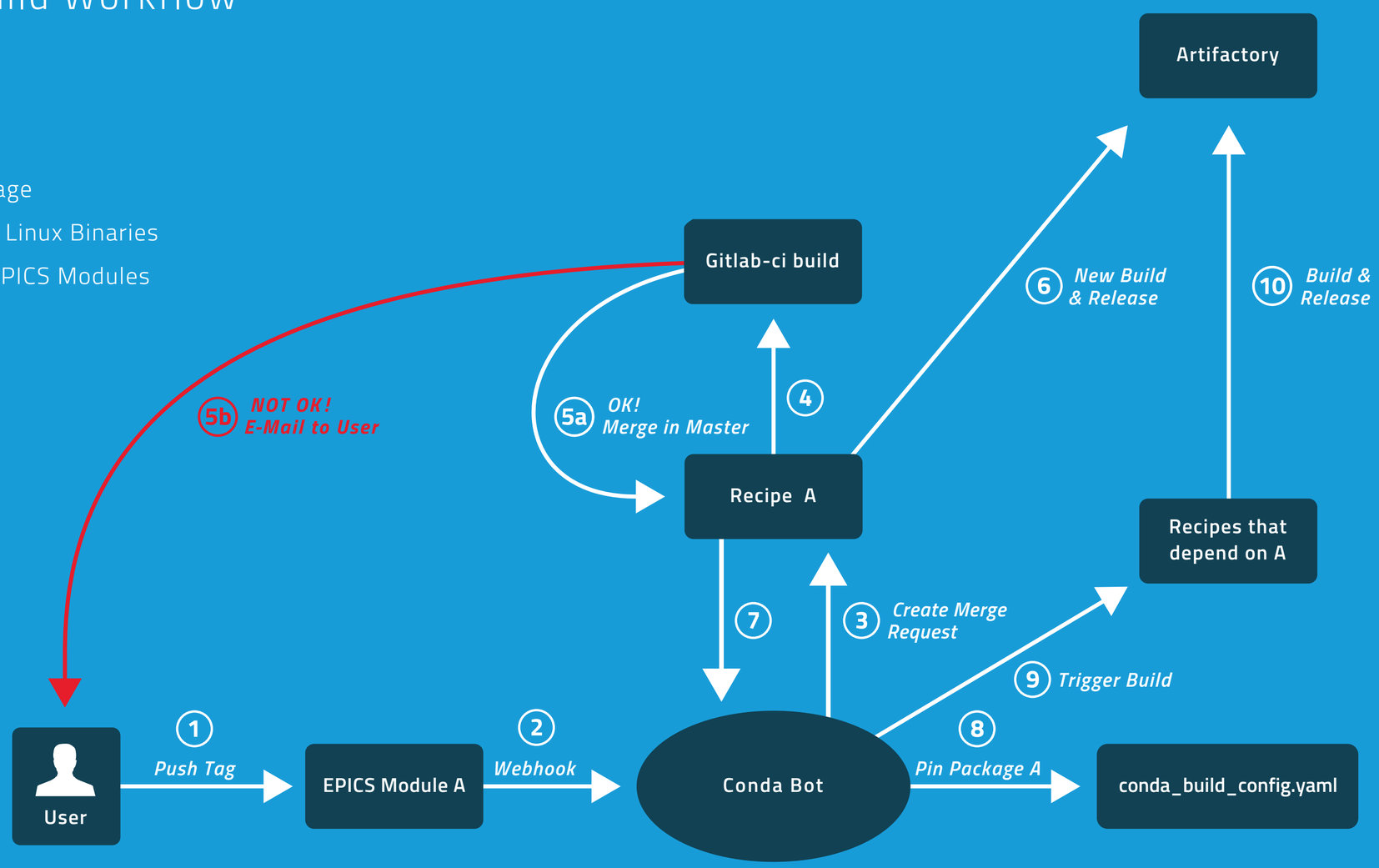
```yaml
build:
  tags:
    - docker
  stage: build
  image: ${CONDA_BUILD_IMAGE}
  script:
    - mkdir pkg
    - >
      conda-build --error-overlinking
      --override-channels -c ${ARTIFACTORY_CONDA_CHANNEL}
      --skip-existing
      --output-folder pkg recipe
  artifacts:
    expire_in: 24h
    paths:
      - pkg/

release:
  tags:
    - docker
  stage: release
  image: registry.esss.lu.se/ics-docker/jfrog:latest
  script:
    - 'echo "conda channel: ${ARTIFACTORY_CONDA_CHANNEL}"'
    - jfrog rt config artifactory --url=${ARTIFACTORY_URL} --apikey=${ARTIFACTORY_API_KEY}
    - jfrog rt upload "pkg/linux-64/*.tar.bz2" ${ARTIFACTORY_CONDA_CHANNEL}/linux-64/
    - jfrog rt upload "pkg/noarch/*.tar.bz2" ${ARTIFACTORY_CONDA_CHANNEL}/noarch/
  only:
    - master
    - /^release-.*$/
```

```bash
#!/bin/bash

# LIBVERSION shall only include MAJOR.MINOR.PATCH for require
LIBVERSION=$(echo ${PKG_VERSION}| cut -d'.' -f1-3)

# Clean between variants builds
make clean

make MODULE=${PKG_NAME} LIBVERSION=${LIBVERSION}
make MODULE=${PKG_NAME} LIBVERSION=${LIBVERSION} db
make MODULE=${PKG_NAME} LIBVERSION=${LIBVERSION} install
```

```yaml
{% set version = "3.11.0" %}

package:
  name: sis8300llrf
  version: "{{ version }}"

source:
  - url: https://gitlab.esss.lu.se/epics-modules/sis8300llrf/-/archive/v{{ version }}/sis8300llrf-v{{ version }}.tar.gz
    sha256: 738348c4d95f2a5fff5a51fd970ca6814e4abd3c66fd861e19db283ae49f0442
  - path: ../src

build:
  number: 0
  run_exports:
    - {{ pin_subpackage('sis8300llrf', max_pin='x.x.x') }}

requirements:
  build:
    - make
    - {{ compiler('c') }}
    - {{ compiler('cxx') }}
  host:
    - epics-base
    - require
    - asyn
    - loki
    - nds
    - libxml2
    - sis8300drv
    - sis8300
    - sis8300llrfdrv

test:
  requires:
    - run-iocsh
  commands:
    - run-iocsh -r sis8300llrf

about:
  home: https://gitlab.esss.lu.se/epics-modules/sis8300llrf
  license: GPL3
  license_file: LICENSE
  summary: "EPICS sis8300llrf module"
```

Automated Recipes Build Workflow

- **Conda Benefits**
  - Install **Binary** Locally
  - Build Different **Variants** of a Package
  - Build **Portable**, Cross-Distribution Linux Binaries
  - Manage **Dependencies** Between EPICS Modules



FEEL FREE TO CONTACT
BENJAMIN.BERTRAND@ESSS.SE