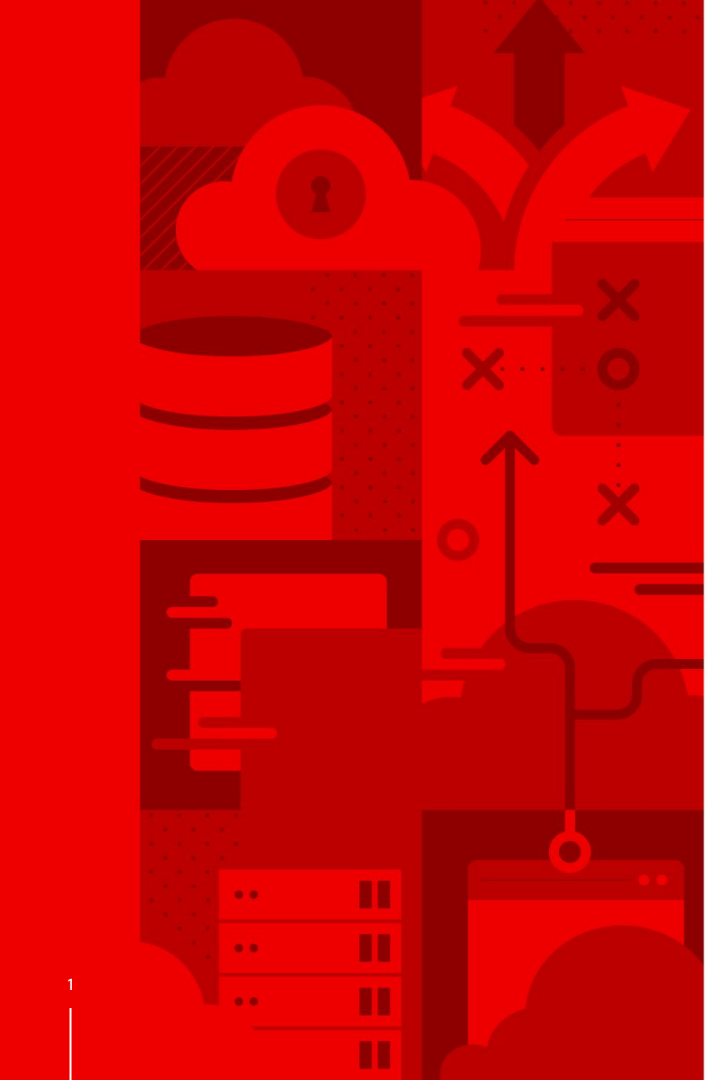


Thoth: Improvements in OpenShift Python s2i

DevConf.US Virtual 2020

Fridolin Pokorny <fridolin@redhat.com>
AICoE, Project Thoth

Red Hat

Agenda

Agenda

1. What is s2i?
2. OpenShift s2i - turn your Git repo into a contianer running in a cluster
3. Existing OpenShift s2i
4. What is Thoth and why do we need it in OpenShift Python s2i?
5. Why Thoth's addition to Python s2i?
6. Thoth s2i

What is s2i?

Source-To-Image

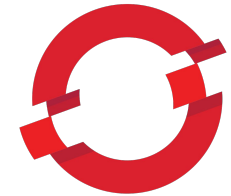
What is s2i?

Source-to-Image (S2I) is a tool for building reproducible, Docker-formatted container images. It produces ready-to-run images by injecting application source into a container image and assembling a new image. The new image incorporates the base image (the builder) and built source and is ready to use with the `buildah run` command. S2I supports incremental builds, which re-use previously downloaded dependencies, previously built artifacts, and so on.

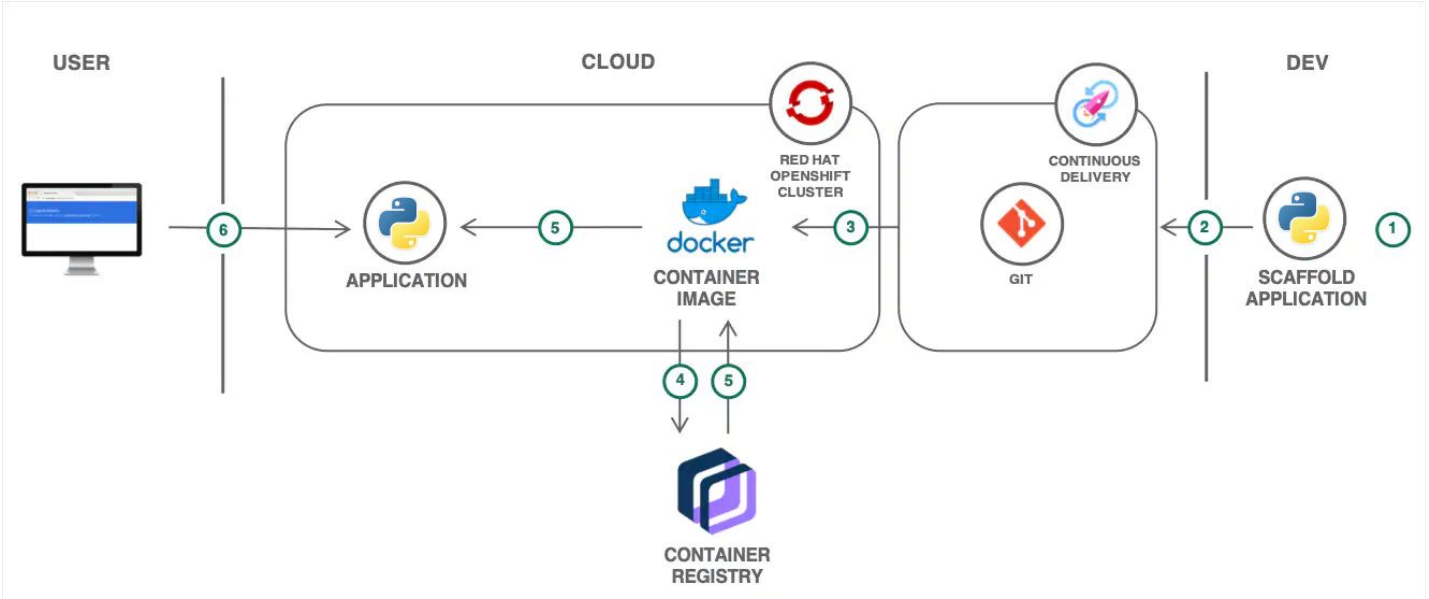
Source: https://docs.openshift.com/container-platform/4.5/builds/understanding-image-builds.html#build-strategy-s2i_understanding-image-builds

What is s2i?

- A way of building container images in OpenShift
- A build process that turns a Git repository into a container image
- Easy to follow conventions
  - Easy to turn any application living in a Git repo into a container image
- Built container images are ready to be deployed into an OpenShift cluster
  - Native support in OpenShift



What is s2i?



Source: https://www.ibm.com/cloud/blog/ibm-cloud-solution-tutorials-2019-in-review (additionally adjusted)

OpenShift Python s2i: Existing solutions

- Fedora based s2i container images:
  - https://github.com/sclorg/s2i-python-container
  - https://registry.fedoraproject.org/
- UBI based s2i container images
  - https://catalog.redhat.com/software/containers/ubi8/python-36/5c839d99d70cc51dd4c42610
  - https://catalog.redhat.com/software/containers/ubi8/python-38/5dde9cacbed8bd164a0af24a
- Thoth s2i container images
  - https://github.com/thoth-station/s2i-thoth
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py36
  - https://quay.io/repository/thoth-station/s2i-thoth-f32-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-f31-py37

OpenShift Python s2i: Existing solutions

- Fedora based s2i container images:
  - https://github.com/sclorg/s2i-python-container
  - https://registry.fedoraproject.org/
- UBI based s2i container images
  - https://catalog.redhat.com/software/containers/ubi8/python-36/5c839d99d70cc51dd4c42610
  - https://catalog.redhat.com/software/containers/ubi8/python-38/5dde9cacbed8bd164a0af24a
- Thoth s2i container images
  - https://github.com/thoth-station/s2i-thoth
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py36
  - https://quay.io/repository/thoth-station/s2i-thoth-f32-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-f31-py37

Why Thoth's addition to Python s2i?

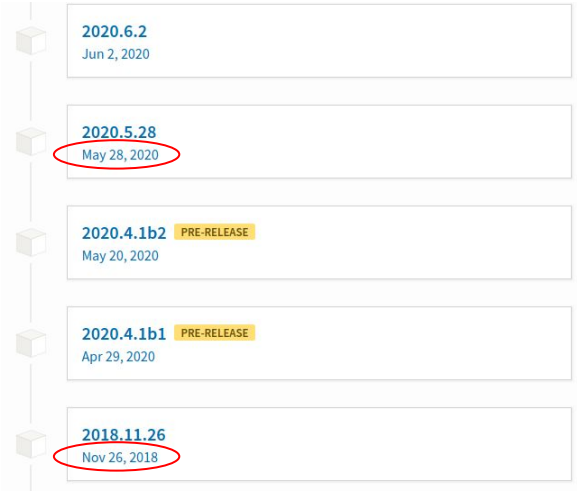
What is Thoth?

- A project in AI Center of Excellence (AICoE), Office of the CTO
- Software stacks are complex and only not-changing when running in production
  - Stacks depend on a lot of components, they keep changing all the time
  - Stacks span a lot of layers
  - Vast amount of platforms available as runtime alternatives
  - Off the shelf builds and upstream libraries do not fit corporate needs
- Make OpenShift a better platform to run AI/ML workloads
  - Python is the driving force for AI/ML applications

Find us on GitHub: http://github.com/thoth-station/

Why Thoth's addition to Python s2i?

- Build container images using OpenShift s2i or using Tekton
- Pinned software stacks (you SHOULD!) - Pipenv
- Issues with Pipenv
  - Pipenv community was innactive and did not release new versions of Pipenv
  - If this project is dead, just tell us · Issue #4058 · pypa/pipenv
  - https://pypi.org/project/pipenv/
  - Installed during the build process from the Internet
  - Forked and maintained our own thoth-pipenv fork
    - https://pypi.org/project/thoth-pipenv/



but...

Why Thoth's addition to Python s2i?

- Do we really need Pipenv when deploying Python applications?
  - Hard to maintain
    - https://github.com/pypa/pipenv/tree/master/pipenv/vendor
  - Quite large project
    - cca 33MiB after installation (Fedora 31, Python 3.6)
    - https://github.com/fridex/s2i-example-micropipenv
  - Python applications are already shipped with a resolved software stack
    - A pinned down list of packages with their versions which should be installed
    - No need to implement resolver, stacks are already resolved

Why Thoth's addition to Python s2i?

... Introducing micropipenv...

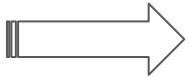
micropipenv is a lightweight wrapper for pip to support requirements.txt, Pipenv and Poetry lock files or converting them to pip-tools compatible output. Designed for containerized Python applications but not limited to them.

https://github.com/thoth-station/micropipenv

https://pypi.org/project/micropipenv/

Why Thoth's addition to Python s2i?

- Main benefits of micropipenv
  - No more vendored dependencies (one optional dependency)
  - One lightweight tool to install dependencies
    - cca 1180 lines in one Python script (841 LOC)
  - Support for:
    - Pipenv lock file
    - Poetry lock file
    - pip-tools style requirements.txt
    - raw requirements.txt
  - Installation log is easy to process by humans or by machines



One single script to rule them all.

micropipenv demo

Why Thoth's addition to Python s2i?

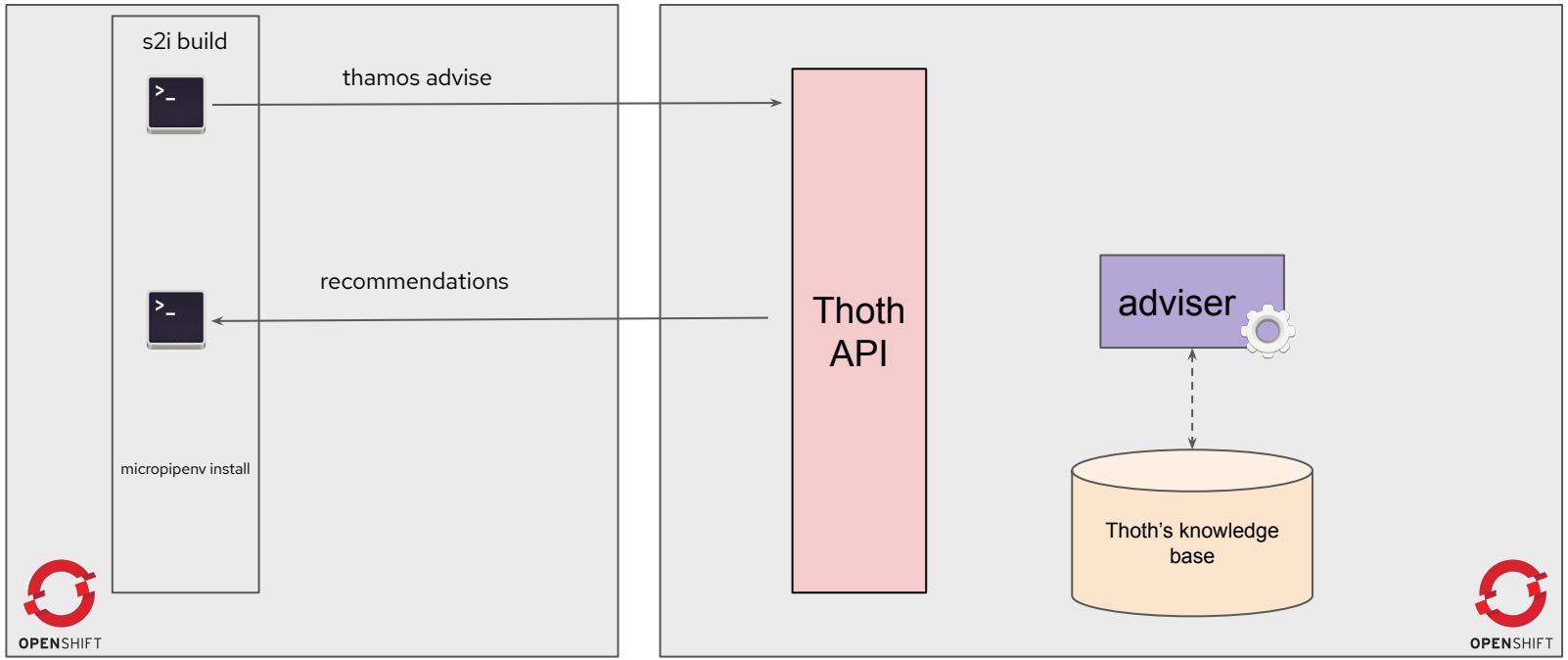
- micropipenv
  - *Designed for containerized Python applications but not limited to them.*
  - https://pypi.org/project/micropipenv/
  - https://github.com/thoth-station/micropipenv/
  - Packaged and maintained in Fedora thanks to Lumir Balhar

```
$ dnf install -y micropipenv

$ pip install micropipenv

$ micropipenv --help
```

Why Thoth's addition to Python s2i?

- Thamos
  - A command line interface (CLI) for communicating with Thoth
  - https://github.com/thoth-station/thamos
- Thoth recommends Python packages to be installed
  - Considers build time and runtime environment
  - Resolves stacks not to latest but to greatest packages
  - Server side resolution
    - See talk "Reinforcement learning based dependency resolution"

Why Thoth's addition to Python s2i?



Why Thoth's addition to Python s2i?

- Thamos as an interface to Thoth
  - https://thoth-station.ninja/docs/developers/adviser/integration.html

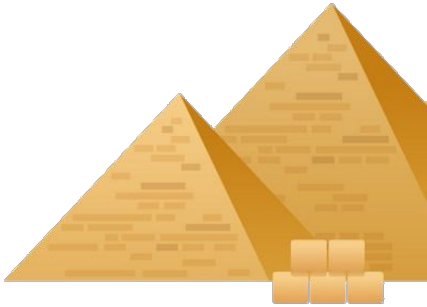
```
$ pip install thamos

$ thamos --help

$ thamos config

$ thamos advise
```

Thoth Python s2i

- Thoth s2i container images
  - https://github.com/thoth-station/s2i-thoth
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-ubi8-py36
  - https://quay.io/repository/thoth-station/s2i-thoth-f32-py38
  - https://quay.io/repository/thoth-station/s2i-thoth-f31-py37
- OpenShift s2i
  - https://thoth-station.ninja/docs/developers/adviser/integration.html#openshift-python-s2i-build-process
- Thoth's homepage:
  - https://thoth-station.ninja/
  - https://github.com/thoth-station/

Project Thoth

- AICoE, Office of the CTO
- GitHub organization
  - http://github.com/thoth-station/
- Twitter account with updates
  - https://twitter.com/thothstation
- YouTube channel
  - https://www.youtube.com/channel/UCIUIDuq_hQ6vlzmqM59B2Lw



Thanks for your attention!

https://github.com/thoth-station

https://twitter.com/thothstation

https://www.youtube.com/channel/UCIUIDuq_hQ6vlzmqM59B2Lw

References

Website https://thoth-station.ninja/

Twitter https://twitter.com/thothstation

GitHub https://github.com/thoth-station


linkedin.com/company/red-hat


youtube.com/user/RedHatVideos


facebook.com/redhatinc


twitter.com/RedHat

Red Hat