Using Docker with HTCondor—Learn It, Live It, Love It

Michael Fienen\textsuperscript{1}, Richard Erickson\textsuperscript{2}, and Grace McCalla\textsuperscript{2}

\textsuperscript{1}U.S. Geological Survey
\textsuperscript{2}Wisconsin Water Science Center, Middleton, Wisconsin
\textsuperscript{2}Upper Midwest Environmental Sciences Center, La Crosse, Wisconsin

HTCondor Week 2017
University of Wisconsin–Madison
Docker has been in HTCondor for a couple years

Containerizing Linux (*and Windows to some extent*)

A Ladder out of Dependency Hell
Introduction: *learn it*

Some Basic Background: *live it*

An Example: *love it*
"Docker is an open-source project that automates the deployment of applications inside software containers. It is promoted by the company Docker, Inc."

— Wikipedia
Container

Containers vs. VMs

Containers are isolated, but share OS and, where appropriate, bins/libraries.

https://insights.sei.cmu.edu/devops/2015/01/devops-and-docker.html
$ docker run -it -v $PWD:/opt/tmp -w /opt/tmp continuumio/miniconda /bin/bash
Configuring a Container: The Seemingly Easiest Way

```
$ docker run -it -v $PWD:/opt/tmp -w /opt/tmp
continuumio/miniconda /bin/bash

root@afa2f9b6f661:/opt/tmp#
```
$ docker run -it -v $PWD:/opt/tmp -w /opt/tmp continuumio/miniconda /bin/bash

root@afa2f9b6f661:/opt/tmp#

root@afa2f9b6f661:/opt/tmp# conda create -n newenv python=3.5 matplotlib numpy pandas
$ docker run -it -v $PWD:/opt/tmp -w /opt/tmp
continuumio/miniconda /bin/bash

root@afa2f9b6f661:/opt/tmp#

root@afa2f9b6f661:/opt/tmp# conda create -n newenv python=3.5 matplotlib numpy pandas

$ docker commit -m "Added python 3 env" -a "Mel Torme" afa2f9b6f661 mtorme/py3_pandas
$ vi pandas.dockerfile
$ vi pandas.dockerfile

FROM continuumio/miniconda
MAINTAINER Mel Torme <mtorme@crooners.org>
ENV WINEPATH z:\\codebase
RUN conda create -n newenv python=3.5 matplotlib
    numpy pandas
$ vi pandas.dockerfile

FROM continuumio/miniconda
MAINTAINER Mel Torme <mtorme@crooners.org>
ENV WINEPATH z:\\codebase
RUN conda create -n newenv python=3.5 matplotlib numpy pandas

docker build -f pandas.dockerfile -t mtorme/py3_pandas .
Configuring a Container: The Boss Way – Dockerfile

$ vi pandas.dockerfile

FROM continuumio/miniconda
MAINTAINER Mel Torme <mtorme@crooners.org>
ENV WINEPATH z:\\codebase
RUN conda create -n newenv python=3.5 matplotlib numpy pandas

docker build -f pandas.dockerfile -t mtorme/py3_pandas .

This is repeatable!
Dockerhub—your happy Docker home online

Like github, bitbucket, etc. online host of Docker containers
Dockerhub—your happy Docker home online

Like github, bitbucket, etc. online host of Docker containers

$ docker push mtorme/py3_pandas

$ docker pull mtorme/py3_pandas
HTCondor is docker/dockerhub aware:

universe=docker
docker_image=mtorme/py3_pandas:v2
HTCondor and Dockerhub

HTCondor is docker/dockerhub aware:

universe=docker
docker_image=mtorme/py3_pandas:v2

will run cached version if available, or will pull from dockerhub
Dockerhub—not perfect

Can be a bottleneck, not always responsive, especially with big containers
Dockerhub—not perfect

Can be a bottleneck, not always responsive, especially with big containers

But, Dockerhub is open source, so can stand up your own
Dockerhub—buyer beware

Even seemingly trustworthy containers might contain bugs/security issues
Even seemingly trustworthy containers might contain bugs/security issues

Good to update packages in your dockerfiles and/or use docker store

https://www.theregister.co.uk/2015/05/28/docker_hub_images_buggy_and_vulnerable_say_researchers/
An Example from USGS

Dependency Hell: Groundwater Monte Carlo
Monte Carlo of Model Parameters and Future Conditions
developer: “the code needs gcc 6.2”
developer: “the code needs gcc 6.2”

me: “that’s cool, but yum install didn’t work”
developer: “the code needs gcc 6.2”

me: “that’s cool, but yum install didn’t work”

developer: “oh, right. you just have to compile the compiler”
GCC 6.2

developer: “the code needs gcc 6.2”

me: “that’s cool, but yum install didn’t work”

developer: “oh, right. you just have to compile the compiler”

me: “huh? seriously?”
## GCC 6.2

<table>
<thead>
<tr>
<th>DEPARTMENT</th>
<th>COURSE</th>
<th>DESCRIPTION</th>
<th>PREREQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPUTER SCIENCE</td>
<td>CPSC 432</td>
<td>INTERMEDIATE COMPILER DESIGN, WITH A FOCUS ON DEPENDENCY RESOLUTION.</td>
<td>CPSC 432</td>
</tr>
</tbody>
</table>

[https://xkcd.com/754/](https://xkcd.com/754/)
Start with a container that has the compiler and dependencies

If only that was the end...
More dependencies

PESTPP only runs if compiled with GCC6.2
More dependencies

PESTPP only runs if compiled with GCC6.2

PESTPP won’t run in WINE
More dependencies

PESTPP only runs if compiled with GCC6.2

PESTPP won’t run in WINE

but....some child codes run Windows only and we don’t have source
More dependencies

PESTPP only runs if compiled with GCC6.2

PESTPP won’t run in WINE

but.....some child codes run Windows only and we don’t have source

and.....some of the exe files are WIN32 but we are running Centos7 (no 32 bit WINE)
A few final thoughts

Docker integration with HTCondor is very straightforward
A few final thoughts

Docker integration with HTCondor is very straightforward

A little care goes a long way in making dockerfiles_containers
A few final thoughts

Docker integration with HTCondor is very straightforward

A little care goes a long way in making dockerfiles/containers

Garbage collection?
**A few final thoughts**

Docker integration with HTCondor is very straightforward

A little care goes a long way in making dockerfiles/containers

Garbage collection?

Inception computing is a little scary, but repeatable and attainable!
A few final thoughts

Docker integration with HTCondor is very straightforward

A little care goes a long way in making dockerfiles/containers

Garbage collection?

Inception computing is a little scary, but repeatable and attainable!

Moving containers remains an issue but
   Dockerhub often works

Use HTCondor to pre-stage

Use HTCondor to push and build Dockerfiles?
Acknowledgements

Funding Provided by the USGS:
Center for Data Integration
Advanced Computing Consortium
Collaborations with Randy Hunt, Janice Gordon, Jeff Falgout
IT Support from Al Barber, Cory Bos, Jim Roys, Mel Bower, Jon Knudsen, Joshua Lee
HTCondor Team

Any Questions?