subMIT: Available Software and Environments

Matthew Heine 1/6/2023

Outline

- Do I need to change my software/code?
- OS & Available Software
- Package & Environment Managers
- Containers
- JupyterHUB
- Summary

If you have never used a cluster or HPC system…

- subMIT offers more computational horsepower than your desktop/laptop
  - more memory
  - many CPUs
  - connected compute resources
  - when you disconnect from subMIT, your jobs can keep running
- Do I need to change my <u>software</u> / <u>code</u> to take advantage of this?
  - Maybe just change your *workflow*
- Simultaneously run independent serial applications on different CPUs
  - investigate dependence of model on input parameters
  - ensemble averaging / Monte Carlo simulations
  - visualization: rendering frames of a movie
- Many 3rd party applications / libraries are already written for HPC
- Resources to help further parallelize your code

Available Software

What comes pre-installed:

Linux Operating System (CentOS 7)

- standard linux packages

Programming

- Python
- C++
- Java
- Fortran
- Julia
- MATLAB
- OpenMPI

But…

You are not limited to pre-installed software

You have the control to install additional software / libraries (the versions you want).

Tools to help you easily do/mange this

- package & environment mangers
- containers
- subMIT help desk

CernVM File System (CVMFS)

- ROOT, GEANT4, Singularity, etc.

Package/Environment Managers (conda, spack)

- Any language (Python, C/C++, Fortran, R, …)
- Easy to find & install packages (software, libraries, …)
  - ~ conda install [name of package]
- Automatically solves & manages dependencies (including automatically installing them)
- Easily update (or roll back) packages
- Easily remove packages & clean house

- Environment Management
  - Easily switch between different sets of consistent packages & dependencies
  - Handle software with conflicting dependencies
  - Multiple versions of a package (e.g. collaborator's code)
  - Helpful for code development & updates
- Reproducibility
- Portability

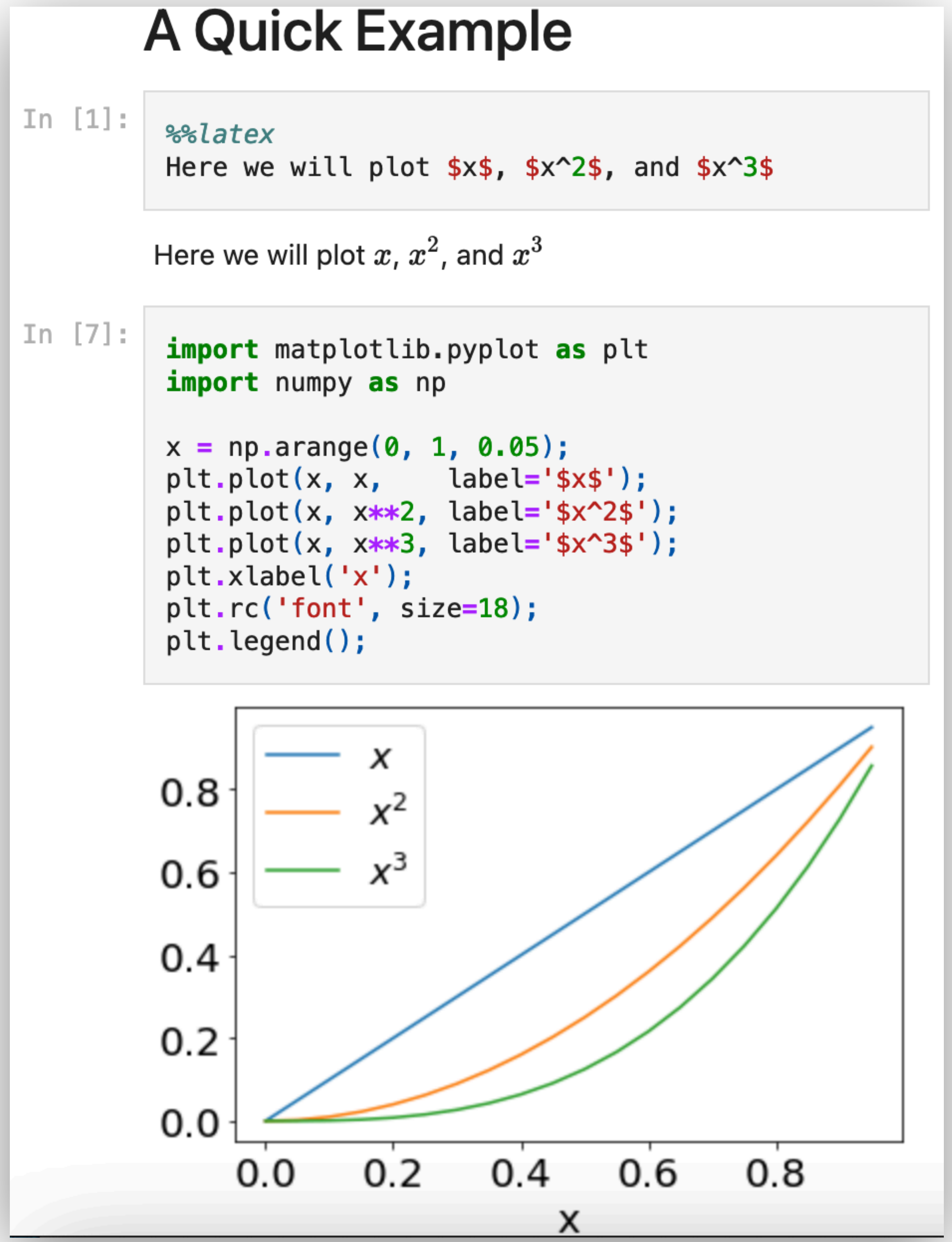
Containers (Docker, Singularity)

- Package into a single container file
    - Environment
    - Code
    - Data
    - User space portion of OS
- Well-suited for HPC
- Reproducibility
- Portability

- Example Use-Cases
    - Run different version of Linux
    - Legacy code/system
    - Time-capsule (writing a paper)
    - Deploy your code/environment to other systems
    - Alternative to installing software
- Ready-made containers
    - https://hub.docker.com/
    - CVMFS
- You can build containers on subMIT —> deploy to other systems

Jupyter: a different way to interact

- Access subMIT from a web browser
- Access subMIT compute horsepower in a more familiar, interactive way
- Jupyter Notebooks: Self-Contained
  - Code
  - Results / Visualization
  - Documentation (Markdown, LaTEX)
- Easily shared
- Example: machine learning or model sensitivity to parameters
  - Interactive
  - Iterative
  - Human-in-the-loop



A Quick Example

In [1]:

```
%%latex
Here we will plot $x$, $x^2$, and $x^3$
```

Here we will plot x, x^2, and x^3

In [7]:

```
import matplotlib.pyplot as plt
import numpy as np

x = np.arange(0, 1, 0.05);
plt.plot(x, x,    label='$x$');
plt.plot(x, x**2, label='$x^2$');
plt.plot(x, x**3, label='$x^3$');
plt.xlabel('x');
plt.rc('font', size=18);
plt.legend();
```

Summary

- subMIT is flexible
- you are in control of your software
    - if we don’t have it, you can bring your own
- package/environment managers (conda, spack) & containers (docker, singularity)
    - install / run new software & libraries on subMIT
    - environment management is powerful
    - make your code / workflow portable (you can build containers on subMIT)
- JupyterHUB
    - Interactive & self-contained

Please ask us if you don’t see how to fill your needs