

SIMPLIFYING SCIENTIFIC PYTHON PACKAGE INSTALLATION AND ENVIRONMENT MANAGEMENT

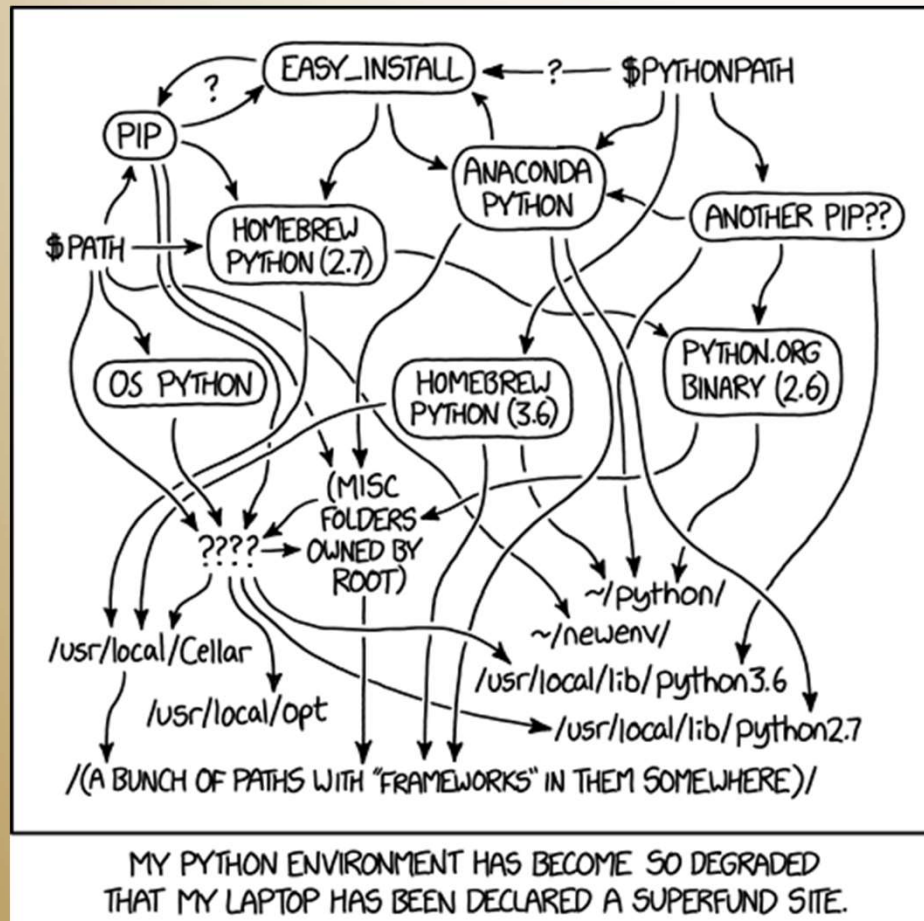
AMIYA K MAJI

BSSw Fellowship Introduction

ITaP Research Computing
Purdue University



MOTIVATION



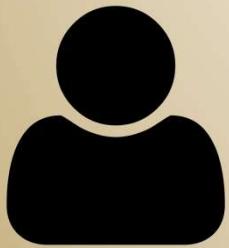
Source: <https://xkcd.com/1987/>

- How can I install tensorflow/pytorch?
- Why can't I use tensorflow in Jupyter?
- Why is my Jupyter session not loading?
- Why is "sudo pip install" asking for password?
- Why can't I import the libraries I installed?
- How can I share my environment with a colleague?
- Various conda related issues ...

CONTRIBUTIONS

- Managing and using virtual environments become too daunting for novice users
- Simplify and streamline installation of Python packages
 - **Simplify** management of virtual environments
 - **Non-interference** with existing packages
 - **Flexible** activation
- Empower interactive Python users

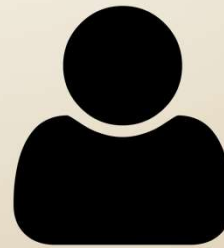
CONDA-ENV-MOD: INSTALL PYTHON PACKAGES



create myenv



- conda create
- create module
- create kernel



module load



- pip install
- conda install

SUCCESS STORIES

- Deep Learning package installation

- 11 applications (multiple versions)
- 3 Python versions
- CPU and GPU version



- Shared Python environment for teaching

- Instructor uses conda-env-mod to install packages
- Students load modules and Jupyter Kernels
- 12+ course, 1500+ students (since Fall 2019)
- Data Science, Atmospheric Science, Molecular Chemistry, Library Science



- Shared Python environment for research groups

- New feature requests

CURRENT FEATURES OF CONDA-ENV-MOD

- Supports conda environments
- Environment creation/deletion
- Lmod module files
- Module file creation/deletion
- Jupyter kernel creation/deletion

FELLOWSHIP OBJECTIVES

- Capture current best practices across HPC centers. Prepare requirements
- Design, implement, and release python-env-mod
- BSSw webinar presentation and community engagement

PROPOSED ADDITIONAL FEATURES

- Add support for multiple environment backends: conda, venv/pipenv
- Add support for different module file formats: Lmod, TCL, Bash/Csh
- Add support for site-customized module templates
- Add support for descriptive help messages
- Add support for creating environment from yaml spec
- Capture compiler/mpi dependencies in environment
- Stretch goals
 - Add support for popular package installation
 - Dependency checking
 - Inform developers about missing dependencies and misconfigurations

CONCLUSION

- Broad impacts
 - Capture best practices for scientific Python application installation
 - Engage the scientific Python community for better packaging
 - Improve scientific productivity, reduce user errors
 - Help interactive Python use
 - JupyterHub
 - Open OnDemand
- Download
 - `https://github.com/amaji/conda-env-mod`

QUESTIONS

ACKNOWLEDGEMENTS

- Lev Gorenstein
- Geoffrey Lentner
- Xiao Zhu
- Preston Smith