SnakeWrangler: A Multi-User IPython Notebook Management System

Peter Bui, Haley Muotka, Trey Zahradka
Computer Science
{buipj,muotkaht,zahradtj}@uwec.edu

1 Project Proposal

In recent years, Python[1] has become one of the most popular introductory programming languages[2] for teaching both Computer Science majors and non-majors. Its wide adoption by both industry and academia is in part due to the availability of powerful tools such as NumPy[4], SciPy[5], and Matplotlib[6] which allow users to quickly and easily generate, modify, analyze, and visualize data from a variety of domains.

At the University of Wisconsin - Eau Claire, IPython, an interactive platform that combines the software tools mentioned previously, has been used as the main teaching platform for CS 170 Computing for the Sciences and Mathematics for the last two years and has received positive reviews from both students and faculty (in and outside of Computer Science).

Unfortunately, while using IPython has been a positive experience for the students, maintaining the infrastructure for the software has not. Currently students access their IPython notebooks via a web browser, which is convenient for end users since no special software has to be installed, but is also a maintenance burden for the instructor who must now monitor and maintain a set of web servers.

The goal of this project is to alleviate this burden of running, monitoring, and maintaining these IPython notebooks by providing a straightforward software platform called SnakeWrangler that automates the configuration and deployment of the IPython notebook web servers for the instructor.

2 Methodology

In the past, the management of the IPython notebook has been performed in two ways:

1. **Decentralized**: In this approach, the instructor provides a script or tool to the students that will automatically configure and run their IPython notebook. This means that whenever the student wants to use their IPython notebook, they must first login to a Linux machine and execute this script.

The disadvantage of this approach is that accessing the command line in order to run a web server is a daunting task for many students, particularly the non-CS majors that typically take CS 170 and thus is an unnecessary obstacle for beginning learners. Moreover, because each student is responsible for their own server, it is difficult to standardize and enforce policies such as security and network resource usage. Finally, this approach also complicates monitoring and troubleshooting errors since each user is responsible for their own environment.

2. **Centralized**: Because of the disadvantages of the previous approach, we tried a centralized method by having the instructor configure and maintain all of the notebooks. This alleviates the burden of managing of the notebooks from the students by shifting it to the instructor who is better equipped to deal with these sorts of issues.

   Unfortunately, this approach also has some drawbacks. First, if an IPython notebook server is unresponsive, then the student must contact the instructor and wait for the notebook to be restarted. Additionally, because a larger number of notebooks has to be managed, a system for monitoring and provisioning is required. The current setup utilizes manual checks and ad hoc scripts that are fragile and lacking in robustness.

Our project aims to combine the two approaches by maintaining a centralized hosting arrangement to ensure consistent enforcement of policies while also given individual users the ability to stop and restart their own notebooks. To do this, we will take the existing centralized system and layer an automated management interface that will monitor a number of IPython notebooks on the behalf of the instructor. This new system will also provide a web portal that will allow individual users to view statistics regarding their notebooks and to manually start and restart the system in case of problems.

3   Context and Significance

As noted previously, the use of IPython and Python in general is growing in both Computer Science and other STEM fields. In fact, because of the use of Python, CS 170 has been included as one of the courses to form the basis of the new Geospatial Education Initiative that is a part of the UW Growth Agenda. Therefore, ensuring that students and instructors have an easy-to-use platform for learning and teaching Python is crucial.

This project aims to provide a tool that is missing in the current education environment: a multi-user IPython Notebook management system. Currently, IPython is a single-user system and requires a

separate web server instance for each user (which is why managing a set of notebooks for a whole class is so cumbersome). Most instructors simply utilize the decentralized approach as described above [7], while others use a centralized cloud solution such as SageMathCloud [8]. Our solution attempts bring the two approaches together while maintaining local control and the ability to run on local resources (unlike the cloud-based solutions).

4   History

This project has not been funded by ORSP previously, but it is a continuation of previous work as noted above.

5   Dissemination

We hope to present the results of our research in the following venues:

1. A paper at the Midwest Computing and Instruction Symposium (MICS).

2. A poster or presentation at the Celebration of Excellence in Research and Creative Activity (CERCA).

If appropriate, we will also submit a poster or paper to other Computer Science workshops or conferences. Additionally, we plan on open sourcing our software so the other members of the Computer Science education community can utilize our tools. Moreover, we plan on using this platform in future offerings of CS 170 Computing for the Sciences and Mathematics.

References

[1] Python. https://www.python.org/.

[2] Philip Guo. "Python is Now the Most Popular Introductory Teaching Language at Top U.S. Universities". Communications of the ACM. 2014, July 7.

[3] IPython Interactive Computing. http://ipython.org/.

[4] NumPy. http://www.numpy.org/.

[5] SciPy. http://www.scipy.org/.

[6] Matplotlib. `http://matplotlib.org/`.

[7] Titus Brown. "IPython Notebook – screencast". `http://ivory.idyll.org/blog/ipython-notebook-intro-screencast.html`

[8] SageMathCloud. `https://cloud.sagemath.com/`.