GSoC-2021

Devansh Shukla

I. Basic Information

Name:	Devansh Shukla
Pronouns:	He/Him/His
Academic Program:	Integrated Master of Science in Physics
Institution:	Sardar Vallabhbhai National Institute of Technology, Surat, India (svnit.ac.in)
GitHub:	@devanshshukla99
Matrix:	@devanshshukla99:matrix.org
E-Mail:	devanshshukla99@gmail.com
Contact No.:	+91 9826887954
Permanent Address:	HNo. 269, Triveni Complex, One-Way Gate, Sagar, MP, India 470002
TimeZone:	IST — UTC +5:30

II. Education Background

Current Affiliation:	Sardar Vallabhbhai National Institute of Technology, Surat, India.
Academic Program:	Integrated Master of Science in Physics.
Expected Year of Graduation:	2023
Curriculum vitae:	Curriculum vitae

III. Experience in Programming

III.I Software Skills

Languages:	Python, C
Platforms:	Linux, Windows
Software & Tools:	LaTeX, conda, WxMaxima, qspectrumanalyzer, 4nec2, GQRX, Mathematica.

III.II Open-Source Projects

- **SAS-RFI**
  - Developed a Python program for Radio Frequency Interference scan at Sardar Vallabhbhai National Institute of Technology, Surat, India.
  - github.com/devanshshukla99/SAS
  - Analysed data and results at github.com/devanshshukla99/Radio-Frequency-Interference-Scan.
- **Juno**
  - Developed a python program for straight-forward terminal-based socket communication with 128-bit AES encryption.
  - github.com/devanshshukla99/Juno-ReDesign-Sockets-Communication/
- **Chromos**
  - Package for getting colored text output in Python.
  - github.com/devanshshukla99/Chromos

IV. SunPy

IV.I Why did I choose this Project: SunPy?

I have been very interested in the Astrophysics and Cosmology domain. SunPy specifically caught my eye due to my course, *Intro. to Space Physics*, which included a seminar on Sun-spots in which I utilized SunPy to generate AIA and HMI Image of the Sun, thereby showing the correlation of sunspots and magnetic field; I have been reading about SunPy since then, slowly learning about the incredible community.

This project brings out the best of both worlds, some Solar Physics and some Programming; for me, it looked like a natural choice in my career to learn and hopefully contribute positively to the community.

More specifically, subsection IV.II points to some benefits of 3D plotting I could see; of course, I'm open to finding more in the future

IV.II Benefits of 3D Plotting

- Super helpful when combining visualizations of multiple objects, for example combining HMI with AIAImage or adding a LASCO Map.
- For Example, PFSSPy, a dependent of SunPy, has a feature of overplotting field lines over an AIA Map; it has an exciting attribute, which can overlap field over the image; this will precisely be the best use case for 3D support in SunPy. (Figure 2).
- Suitable for intuitive understanding and visualization.
- Looks super cool.

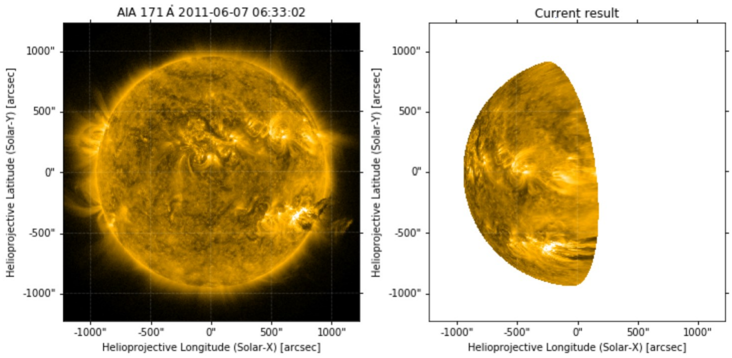


Figure 1: Example for show-casing usefulness of 3D plots, Issue # 3997

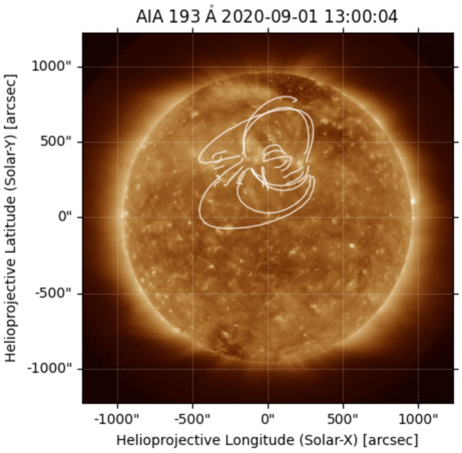


Figure 2: Overplotting field lines over an AIA Map with PFSSPy

IV.III What have other people contributed to this project previously?

Presently, PR #4591 by @dstansby provides a foundation for the project. It's basically a proof-of-concept. It takes a sunpy.map.Map object, converts it to Heliocentric-Inertial Coordinate system, creates a mesh and finally plots in PyVista.

IV.IV My Contributions

PR #5193: Under Review

- Adds support for file-handlers and file-type in Map and TimeSeries.
- github.com/sunpy/sunpy/pull/5193

Issue #3303: Currently working

- Create a Test or a function to report status on all remote servers SunPy pings during a test run.
- Thinking about using a pytest plugin to intercept the requests with a flag but it likly to change. :)
- github.com/sunpy/sunpy/issues/3303

V. Project: 3D Plotting in SunPy

V.I Project Abstract

SunPy currently uses Matplotlib for plots and has limited support for 3D plots owing to PR #4591 .

The project's goal is to pack PR #4591 into a new package, independent of SunPy, and develop a complete API initially interfacing with PyVista but general enough for other backends like plotly too. SunPy has excellent support for 2D plots with Matplotlib, but 3D support is missing; through 3D backing, one can easily combine visualizations of multiple objects such as overlapping HMI on a AIA image in 3D; it'll be a gem for intuitiveness. Moreover, PFSSPy, a dependent package of SunPy, which calculates the Potential-Field Source-surface model for computing the coronal magnetic field features, has an exciting attribute, which can overlap field over the image; this'll precisely be the best use case for 3D support in SunPy. The project is evenly balanced between writing, testing and documenting code.

V.I.I Goals

- Convert code from PR #4591 into a new package under SunPy org, and understand it down it a fiber.
- Add infrastructure to plot astropy coordinate object.
- Add infrastructure to support other data types, too, like for SOHO, RHESSI, etc.
- Hoping for 90%, targeting 100% code-coverage. Benchmarks and extra optimizations, if possible.
- Complete documentation

In all, we have to prepare a pre-release of the package for 3D support.

V.II Time Commitment

I plan to fully commit and immerse the project, with a minimum investment of 20hr/week and with an average of 30hr/week. Honestly, I am not a big fan of assessing based on time investment; my priority would be to make and follow weekly task schedules, with some buffer time for unforeseeable circumstances and additional tests.

V.III Timeline

I have attached a thorough timeline of the proposed project. Regarding my commitments, I'll have my course classes from August 10, so I've shifted some of the workload to July.

I am super eager to publish a weekly blog for the project with all relevant details to help future developers and users; I believe this would be critical in the long run.

- **May 17, 2021 - May 24, 2021**
  - Familiarize with the code and the community.
  - Since I am already familiar with the dev env and test systems used in SunPy(tox, pytest) I'll focus more on the code and how it all is integrated together.

  - Create a separate package with PR #4591 as foundation and set-up all necessary workflows.

- **May 24, 2021 - June 1, 2021**
  - During this time, I'll discuss with my mentors and come with more specific details about the project, like how the public face of the API should behave, etc.
  - I'll learn and try to solve the challenges I'm already aware-of in the project, see autoref
  - Dependencies like AstroPy, plotly, MPL, PyVista.
- **May 24, 2021 - June 1, 2021**
  - Plan for the API.
  - Throughly understand and test the existing code.
  - Dependencies like AstroPy, plotly, MPL, PyVista.
- **June 7, 2021 - June 14, 2021**
  - Define tests and documentation for the existing code.
  - Discuss & Work on the API.
- **June 14, 2021 - June 21, 2021**
  - Add code for plotting sunpy.map.Map
  - Work on the API.
  - Crush the bugs along the way.
- **June 21, 2021 - June 28, 2021**
  - Add tests and documentation, hoping to achieve 75%+ code coverage.
  - Check the Integration with SunPy. – Waypoint - 1 Check
- **June 21, 2021 - June 28, 2021**
  - Crush the bugs.
  - Add tests and documentation, hoping to achieve 90%+ targeting 100% code coverage.
  - Finalizing the API.
- **June 28, 2021 - July 5, 2021**
  - Gather Feedback.
  - Finalizing the API for the evaluation.
  - Crush the Bugs.
- **July 5, 2021 - July 12, 2021**
  - Finalizing the API for the evaluation.
  - Plan for plotting astropy coordinates in 3D infrastructure.
  - Buffer time.
- **July 12, 2021 - July 16, 2021**
  - Buffer time, for incorporating required changes as per feedback.
  - More debugging.
- **June 16, 2021 - July 23, 2021**
  - Work on infrastructure for plotting astropy coordinates in 3D.
  - Discuss on adding infrastructure for other data type.
- **July 23, 2021 - July 30, 2021**
  - Test infrastructure for plotting astropy coordinates in 3D.
  - Plan infrastructure for other data type, SOHO, EUV, RHEISSI, tbd.
  - Buffer time to crush bugs.
- **July 30, 2021 - August 6, 2021**
  - Work on adding infrastructure for other data type, SOHO, EUV etc, tbd.
  - Add tests and documentations, hoping to achieve 75%+.
- **August 6, 2021 - August 13, 2021**
  - Work on support of pyvista for jupyter-notebook.
  - Crush the bugs.

  - Finalize the API.

- **August 13, 2021 - August 20, 2021**
  - Add tests and documentations, hoping to achieve 90%+ targeting 100% code coverage.
  - Finalize the API, prepare for the pre-release.
  - Crush the bugs.
  - Check Documentation for mistakes/typos/bugs.

- **August 13, 2021 - August 16, 2021**
  - Finalize the API, incorporate the feedback from additional review.
  - Crush the bugs.
  - Gather feedback.

- **August 16, 2021 - August 23, 2021**
  - Pre-Release.
  - Crush the Bugs.

- **August 23, 2021 -**
  - Maintain support and work on improving the performance and stuff.
  - Keep crushing the bugs.

VI. GSoC

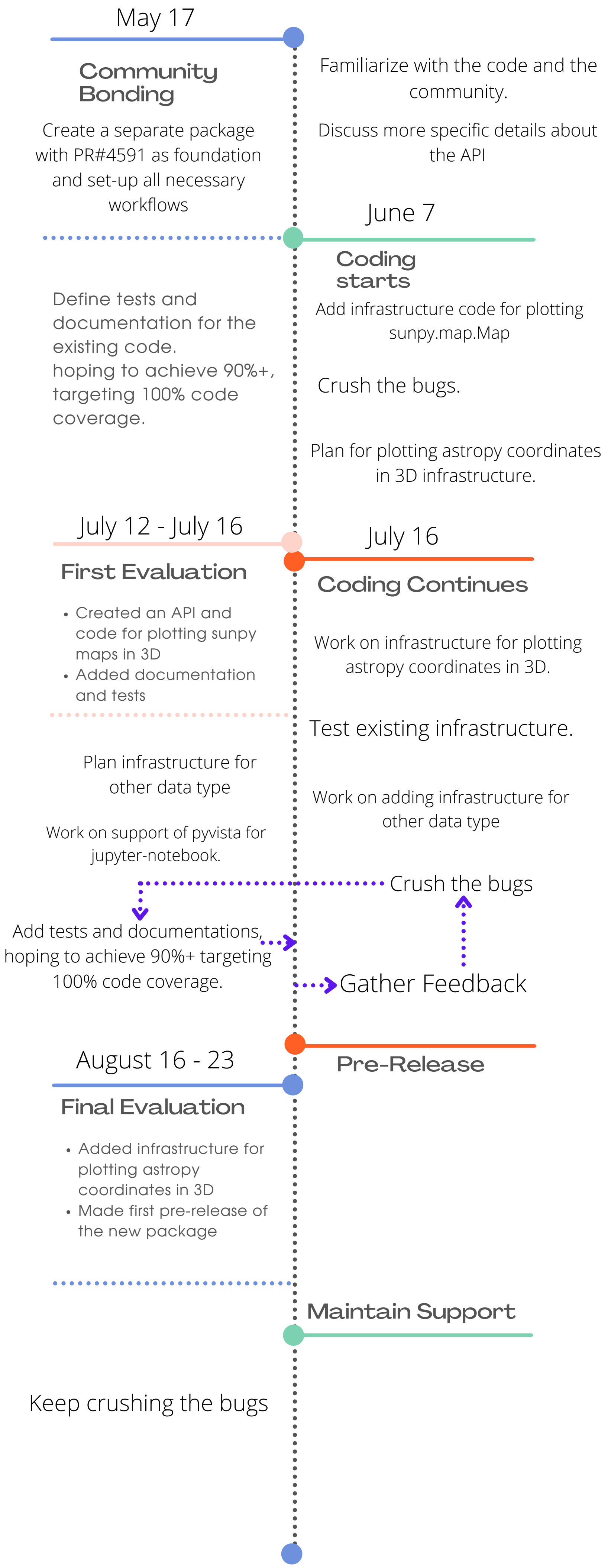
Have you participated previously in GSoC?
This is the first time I'm applying to a project.

Are you also applying to other projects?
This is the only proposal I'm submitting.

How much time do you to invest in the project before, during and after Summer of Code?

- Before - 20hrs/week.
- During - 30hrs/week.
- After - 15hrs/week.

Are you eligible to receive payments from Google?
Yes, I am eligible.



May 17

Community Bonding

Create a separate package with PR#4591 as foundation and set-up all necessary workflows

Familiarize with the code and the community.

Discuss more specific details about the API

June 7

Coding starts

Add infrastructure code for plotting sunpy.map.Map

Define tests and documentation for the existing code. hoping to achieve 90%+, targeting 100% code coverage.

Crush the bugs.

Plan for plotting astropy coordinates in 3D infrastructure.

July 12 - July 16

First Evaluation

- Created an API and code for plotting sunpy maps in 3D
- Added documentation and tests

July 16

Coding Continues

Work on infrastructure for plotting astropy coordinates in 3D.

Test existing infrastructure.

Plan infrastructure for other data type

Work on adding infrastructure for other data type

Work on support of pyvista for jupyter-notebook.

Crush the bugs

Add tests and documentations, hoping to achieve 90%+ targeting 100% code coverage.

Gather Feedback

Pre-Release

August 16 - 23

Final Evaluation

- Added infrastructure for plotting astropy coordinates in 3D
- Made first pre-release of the new package

Maintain Support

Keep crushing the bugs

@devanshshukla99