University of Toronto – Scarborough

CSCD01 – Engineering Large Software Systems

**Deliverable 04 – Report**

Prepared by ***GoonSquad*** Team

Swarnajyoti Datta

Nikki L. Quibin

Junil Patel

Beiyang Liu

Laine London

Hajoon Choi

Leo Li

Date: March 16, 2018

**Table of Contents**

[**Bug #1: Legend Annotate (Fixed)** 1](#_Toc508883223)

[**Bug #2: Bbox Tight Legend (Fixed)** 3](#_Toc508883224)

[**Bug #3: Logscale (Fixed)** 3](#_Toc508883225)

[**Lessons Learned** 3](#_Toc508883226)

# **Bug #1: Legend Annotate (Fixed)**

**Bug/Issue:** [Legend does not show ‘annotate’ #8236](https://github.com/matplotlib/matplotlib/issues/8236)

**Estimated Hours:**

* Explore and create a solution (5 h)
* Implement solution (20 h)
* Testing/validation (8 h)
* Code Review (2 h)
* Documentation (1 h)

**Description:**

Legend items for annotations are currently not operational despite inputs currently being legal for them. Boxing the bi-directional arrows above into a legend to denote amplitude and wavelength is an example of a desirable use case.

When the legend() method on an Axes object gets called, the program eventually adds items to a list of handles to be inserted into the Legend for that Axes (see legend.py:1308, 1313). Annotations, stored in the Axes field texts (as Annotations are a sub-class of the Text class), are not currently added to this list. Additionally, the handler to construct the legend items for Annotations and Texts do not currently exist in the file legend\_handler.py (and are subsequently not mapped in legend.py:805).

**Solution:**

The affected files are ***legend.py***, and ***legend\_handler.py*** where the updated files are found in ***solutions/legend\_annotate/***.

[Add solution description]

**Testing:**

To test that the solution works, image comparison tests were used, as suggested by matplotlib when testing for changes to the graph figure. The associated tests are in ***solutions/legend\_annotate/tests/***. The file ***test\_legend\_annotate.py*** contains the specific test cases for annotations appearing in the legend. The result images are found in the folder ***/test\_legend/*** and should be copied over to ***lib/matplotlib/tests/baseline\_images/test\_legend*** in the actual matplotlib source directory when running tests. In addition, ***test\_legend.py*** is the formal way to test the legend, thus it includes the existing tests for legend and as well as the newly added test cases in ***test\_legend\_annotate.py***. As a result, when testing, this ***test\_legend.py*** should replace the existing matplotlib file.

Since the issue pertains to annotations not appearing in the legend, it is a rendering issue, so image comparison tests are used. Moreover, existing legend tests used image comparison tests for testing the labels, for example, ***test\_various\_labels()***.

There were six tests used. Below are the images used as the baseline images. The test cases cover all the existing linestyles, arrowstyles, couple of colours and texts, no arrow, text, or both, and a practical example of using annotations. Also, the existing tests were ran and have passed.

|  |  |
| --- | --- |
| *test\_all\_arrowstyles()* | *test\_all\_linestyles()* |
| *test\_annotation\_colours()* | *test\_annotation\_text()* |
| *test\_simple\_annotation()* | *test\_annotation\_no\_line\_text()* |

**Confidence in Solution:**

The proposed solution works very well since existing and new tests have passed. In addition, there are only two affected files and are only contained with respect to the legend, thus other parts of the matplotlib code are unaffected. Moreover, the implemented solution is a valid and practical fix as opposed to a “hack” fix. The solution consists of using the ***legend\_handler*** class to add the annotation to the legend, which was also the procedure for adding existing labels. Guidelines for contributing to matplotlib were also followed accordingly, such as proper documentation and following PEP8 guidelines. All in all, there is high confidence that the proposed solution works.

# **Bug #2: Bbox Tight Legend (Fixed)**

**Bug/Issue:** [Legend is not present in the generated image if I use “tight” for bbox\_inches #10194](https://github.com/matplotlib/matplotlib/issues/10194)

# **Bug #3: Logscale (Fixed)**

**Bug/Issue:** [Minor ticks on log-scale colorbar are not cleared #8358](https://github.com/matplotlib/matplotlib/issues/8358)

# **Lessons Learned**

**Swarnajyoti Datta**

**Nikki L. Quibin**

* Working on an open source and large-scale project requires a lot of coordination. Therefore, it’s imperative that matplotlib had a guideline and standard for contributing.
* Quality of tests are important to validate that a solution works as opposed to just hoping it works.
* Fixing bugs in an open source project shouldn’t be a full-time thing, rather, something that should be considered a hobby or honing programming skills.
* Something that looks simple to fix can be really complicated; looks can be deceiving, just like the matplotlib code.

**Junil Patel**

**Beiyang Liu**

**Laine London**

**Hajoon Choi**

**Leo Li**