Wisdoms learnt when contributing to Apache Airflow

An Outreachy intern’s story in contributing to Apache Airflow

Bowrna
Apache Airflow Contributor
Who am I?

- Self-taught programmer
- Software Engineer with 7+ years of experience in building backend systems
- Outreachy Intern for Apache Airflow - Dec 2021 Cohort
- Active Open Source Contributor to Apache Airflow
People who made this talk possible

Outreachy Organisers

My Mentors:
- Jarek Potiuk
- Elad Kalif
- Nasser Kaze

My Cointerns:
- Edith Puclla
- Melodie Ezeanie
My first step in Open source

Ok.. I am going to be an active open source contributor. But where do I start?

But there is no label in issues, no clear documentation to begin contributing

I will pick issues from projects I have used and start working

Ok let me google. Hmm.. I could find good first-issue in many projects
Few weeks later..

With issues in many projects which one should I focus upon?

I'm exhausted and I haven't even begun! May be Open source is not for me

I really wish someone could guide me through this...

Hurray! This Outreachy programme looks so promising. I am definitely trying this
The new beginning

Got selected as an Outreachy intern for Apache Airflow
Rewriting Breeze from BASH to Python

- **Breeze** - Easy to use dev tool for development and testing environment
- Consistent and Common development environment so that it’s easy to replicate the issues in CI pipelines
- Project involves migrating the code from bash script to Python script
CLEAR COMMUNICATION IN WRITTEN FORM IS VERY POWERFUL
Process in getting Pull request merged

- Raise/Update PR
- Read the Conversation
- Take deep dive about it
- Read conversation again
- Refine the code

Diagram:
- Raise/Update the Pull Request for issue
- Read the conversation happening in the issue
- Take deep dive about things mentioned in conversation
- Reread the conversation again
- Refine the code
Example Pull Request

enter the shell breeze2 environment #21145

Bworna commented on 27 Jan

closes: #20960
related: #20960
Breeze2 - enter the shell in breeze2
What I learnt from writing comments in PR

- Writing not only helps to exchange thoughts but also helps to generate new ones.
- Writing makes thought process clearer.
- Writing comments paved path to write blog posts.
- You can check my blog posts in https://bowrna.hashnode.dev/
CODE IS NOT AN ASSET AND DON’T WORRY ABOUT THROWING IT AWAY
Whack the code

Rewrite Breeze setup autocomplete #20066

Bowrna commented on 6 Dec 2021

closes: #19967 (comment)
Breeze rewrite in Python - Sample autocomplete option for breeze commands
Code is not an asset

Initial Thoughts
- Maybe I should have figured out this solution earlier

Realisation
- I wouldn’t have figured out this easy solution if I haven’t worked on the issue for 3 days

Wisdom
- The whole project that I was working on is scraping the old code and rewriting again
DONT FEEL SHY ABOUT ASKING QUESTIONS
The solution is not to ask to ask, but just to ask. Someone who is idling on the channel and only every now and then glances what's going on is unlikely to answer to your "asking to ask" question, but your actual problem description may pique their interest and get them to answer.

So, to summarize, don't ask "Any Java experts around?", but rather ask "How do I do [problem] with Java and [other relevant info]?"
About GIT REBASE

Image by Ryan Hodson
Revised CONTRIBUTING.rst in Apache Airflow

How to rebase PR

A lot of people are unfamiliar with the rebase workflow in Git, but we think it is an excellent workflow, providing a better alternative to the merge workflow. We've therefore written a short guide for those who would like to learn it.

As of February 2022, GitHub introduced the capability of "Update with Rebase" which make it easy to perform rebase straight in the GitHub UI, so in cases when there are no conflicts, rebasing to latest version of `main` can be done very easily following the instructions in the GitHub blog.
Rebasing branch in a button click
OPEN SOURCE IS A RICH ECOSYSTEM
On finding issues in another project

There is a very probable WeakKeyDict bug in Python standard library (to be confirmed and investigated further) that manifests itself in a very rare failure of the test_stacktrace_on_failure_starts_with_task_execute_method.
On fixing issues in another project

Replace key if not identical to old key in dict #31685

This fixes an issue in `dict` and by extension in `weakref.WeakKeyDictionary` which arises when a key that is equal to an existing key is used to set a new value – where the key itself is not replaced.

To appreciate this situation, consider when the keys are type `weakref.ref`.

In this case, the lifetime of the key after replacing the value is now that of the initial key and not the one that's been set subsequently – since the key entry is still the initial object. If the first key falls to refcount zero, the key is removed from the `weakref.WeakKeyDictionary` – which is unexpected because we have set a newer key which might have refcount greater than zero.

In the general case where a value is replaced for the exact same key (object identity), there is no practical difference with this change – and when a new key is used to replace a value, the cost is only a write to memory.

Thanks to @potlu and @asb for helping finding this issue which was discovered during flaky test runs in Apache Airflow.
On contributing to docker documentation
TRIVIAL THINGS BECAME NO MORE TRIVIAL TO ME
Don’t overlook

• Even tiny details are important
• Simple colour coding of messages that is printed for logging purpose is vital
• Little details makes the experience better and saves lot of time
Adding Accessibility

```
Adding Accessibility

> add prophecy 1

Adding Accessibility

```

```shell
$ satishkannan@Satishs-MacBook-Air airflow % breeze config --no-colour

Disable Colour

Current configuration:

* Python: 3.7
* Backend: sqlite
* Postgres version: 10
* MySQL version: 5.7
* MsSQL version: 2017-latest
* ASCIIRT: enabled
* Cheatsheet: enabled

```

```shell
$ satishkannan@Satishs-MacBook-Air airflow % breeze config --colour

Enable Colour

Current configuration:

* Python: 3.7
* Backend: sqlite
* Postgres version: 10
* MySQL version: 5.7
* MsSQL version: 2017-latest
* ASCIIRT: enabled
* Cheatsheet: enabled

* Colour: enabled
```
Thank you!

Contact me:

Github: https://github.com/Bowrna

My blog: https://bowrna.hashnode.dev/

Twitter: https://twitter.com/bowrna_p

LinkedIn: https://www.linkedin.com/in/bowrna/