GitHub Data Proposal: Organizational Hierarchy in Open Source Software Development

Chris Liao[*]

September 23, 2024

Project Summary

The objective of this research project is to understand the relationship between the organizational hierarchy of the Open Source Software (OSS) project and the development of open source software. By combining empirical data on the behavior of OSS contributors and the project with economic theory on knowledge-based organizational hierarchies, I can answer two questions.

1. First, how does the OSS organization allocate responsibilities to different OSS contributors.

2. Second, when the OSS development environment changes, how does project leadership respond and how does the behavior of OSS contributors evolve.

Studying the OSS organizational hierarchy is important because OSS development relies on OSS contributors, whose behavior is affected and whose role in the organizational hierarchy is directly determined by OSS project leadership (Lerner and Tirole 2002, Lerner and Tirole 2005). Given the economic importance of OSS, which is used in at least 70% of all software (Perlow 2022), there's great value in studying the organizational hierarchy underlying OSS development. While it is well documented that OSS organizations are organized into hierarchies based on contributor knowledge (Kevin Crowston and Howison 2006, K. Crowston et al. 2006, Lerner and Tirole 2002), my research is the first that seeks to apply microdata on OSS development and an economic model of knowledge hierarchy to quantitatively study OSS development.

[*] cliao@hbs.edu

Analysis

I'm interested in testing two hypotheses using data

- How do OSS organizations and OSS contributors respond to the departure of highly ranked OSS contributors from the project?
- How do OSS organizations and OSS contributors respond to GitHub platform improvements (like issue & pull request templates) that improve communication?

These are hypotheses that economic models of knowledge hierarchies (Garicano 2000) also study. Adapting existing models for OSS development can also illuminate the economic forces driving change in behavior and generalize the settings in my two hypotheses to broader classes of change. Moreover, using the knowledge hierarchies framework makes my results comparable to existing empirical studies of how traditional firms and their employees respond to shocks to their economic environment (Garicano and Zandt 2012, Bloom et al. 2014).

Next, I list the data that I require for this empirical analysis. I require five categories of data: contributor characteristics, OSS project characteristics, data on issues, data on pull requests, and data on pushes. Since I am analyzing contributor behavior over time, it is necessary to have granular, contributor-level data on how their behavior in a project evolves with their role in the project's hierarchy. In particular, the rank variable, which is a key component of my analysis, is not publicly available. Similarly, since I am studying the organizational hierarchy of a project, it is important to also understand the project itself and understand how that might affect the organizational hierarchy. Finally, data on issues, pull requests, and pushes provide measures of OSS development that I will need to objectively describe the behavior of OSS contributors and quantify project outcomes. Some of the variables that I am describing are not publicly available online, so I am willing and happy to work with anonymized data, and I welcome further discussion on data-specific details!

1 Contributor Data

- **Timespan**: 2011-2024
- **Frequency**: Daily (Monthly)
- **Unit of Observation**: Contributor-project level

- **Observation Set**: Popular Python projects (see attached list)
- **Individuals of Interest**: All contributors who interacted with an issue or pull request

Covariates

Each entry is uniquely identified by **Anonymous GitHub User ID**, **Anonymous GitHub Project ID** and **Date**

- **Anonymous GitHub User ID**
- **Anonymous GitHub Project ID**
- **Date** (Month-Year)
- GitHub Project Owner Type (whether it's an organization or not)
- Rank
- # of GitHub Achievements completed
- Country (when listed on their profile)
- # of followers
- # of following
- % of all merged PRs (in project's history) authored by them
- % of all lines of code (in project's history) authored by them
- # months on GitHub
- # months where they have been involved as a contributor on this project
- For the top 10 projects that they have the most PRs in,
  1. Quantity of PRs (all time)
  2. % of all LOC (in project's history) authored by them
  3. # of Stars
- Anonymous GitHub User ID of the contributor who promoted them/invited them

2 Project Data

- **Timespan**: 2011-2024
- **Frequency**: Monthly
- **Unit of Observation**: Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **GitHub Project ID** and **Date**. This matches to the data described in Section 1 (Contributor Data) through **Anonymous GitHub Project ID**

- **Anonymous GitHub Project ID**
- **Date** (Month-Year)
- For each of the top 5 programming languages used in the project, the % of the codebase that relies on that language
- Project topics
- Project license type
- # of Forks (cumulative)
- # of Stars (cumulative)
- # of Watchers (cumulative)
- # Number of dependencies, cumulative
- # of Dependents, cumulative
- # of Releases (cumulative)
- # of Tags (cumulative)
- Whether reviews are required on the PR
- Whether auto-merge is on
- Whether the project has a CONTRIBUTING.md or similar file
- Whether the substring “contribut” shows up in the README

3 Issue Data

- **Timespan**: 2011-2024
- **Unit of Observation**: Issue-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID** and **Anonymous Issue ID**. This matches to the data described in Section 2 (Project Data) through **GitHub Project ID**

- **Anonymous GitHub Project ID**
- **Anonymous Issue ID** (anonymized version of Issue # on GitHub)
- Issue Opener Anonymous GitHub User ID
- Issue Closer Anonymous GitHub User ID
- Date & Time of issue opening (first time)
- Date & Time of issue closing (first time)
- Whether issue was ever reopened
- Issue current status (open or closed)
- A dictionary of all the times the issue tags for an issue were changed/assigned, with the date and time the issue tags were changed, the new cumulative tags assigned and the tag assignee Anonymous GitHub User ID.
- A dictionary of all the times the assignees for an issue were changed/assigned, with the date & time the assignees were changed/assigned, the new cumulative group of assignees and the assigner. All assignees and assigners are identified by their Anonymous GitHub User ID
- Linked Anonymous Pull Request ID, if available
- Date & Time Pull Request was linked

- Whether an issue development branch was created inside the project and if so, the anonymous branch ID
- Number of characters in issue text (in the latest edit)
- Latest edit date & time of issue
- Each Anonymous GitHub User ID that was mentioned in the issue text
- Whether an Issue Template was used (if there are multiple issue templates, an anonymous id for the issue template based on the issue template filename)

Issue Comment Data

- **Timespan**: 2011-2024
- **Unit of Observation**: Issue Comment-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID**, **Anonymous Issue ID** and **Anonymous Issue ID Comment #**. This matches to the data described in Section 3 (Issue Data) through **GitHub Project ID** and **Issue #**

- **Anonymous GitHub Project ID**
- **Anonymous Issue ID**
- **Anonymous Issue ID Comment #**
- Issue Commenter Anonymous GitHub User ID
- Date & Time of Issue Comment
- Number of characters in issue comment text (latest edit)
- Latest edit date & time of issue
- Each Anonymous GitHub User ID that was mentioned in the issue text
- Reactions on Issue Comment (as of data collection)

4 Pull Request Data

- **Timespan**: 2011-2024
- **Unit of Observation**: Pull Request-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID**, **Anonymous Pull Request ID** This matches to the data described in Section 2 (Project Data) through **Anonymous GitHub Project ID**

- **Anonymous GitHub Project ID**
- **Anonymous Pull Request ID**
- Anonymous Issue ID
- Pull Request Opener Anonymous GitHub User ID
- Pull Request Merger/Closer Anonymous GitHub User ID
- Date & Time of Pull Request opening (first time)
- Date & Time of Pull Request closing or being merged (first time)
- Date & Time the issue was linked
- Pull Request current status: Open, Closed or Merged
- A dictionary of all the times the tags for a pull request were changed/assigned, the new cumulative tags assigned and the tag assignee Anonymous GitHub User ID.
- A dictionary of all the times the assignees for a pull request were changed/assigned, with the date & time the assignees were changed/assigned, the new cumulative group of assignees and the assigner. All assignees and assigners are identified by their Anonymous GitHub User ID

- A dictionary of all the times the reviewers for a pull request were changed/assigned, with the date & time the reviewers were changed/assigned, the new cumulative group of reviewers and the assigner. All reviewers and assigners are identified by their Anonymous GitHub User ID

- Whether the Pull Request took place on a branch in the project and if so, the anonymous branch ID

- How many GitHub Actions checks were run and how many passed (latest run)

- Number of characters in the Pull Request Text (in the latest edit)

- Latest edit date & time of Pull Request Text

- Each Anonymous GitHub User ID that was mentioned in the issue text

- The # of other Pull Requests that were mentioned in the issue text

- The # of other issues that were mentioned in the Pull Request text

- Number of commits prior to Pull Request being opened

- Number of commits after Pull Request was opened

- # LOC added & removed in commits prior to Pull Request being opened

- # LOC added & removed in commits after Pull Request was opened

- # files added & removed in commits prior to Pull Request being opened

- # files added & removed in commits after Pull Request was opened

- Date and Time of First Commit

- Date and Time of Last Commit (prior to Pull Request being opened)

- Date and Time of Last Commit (prior to merge or closing)

- Total number of PR review comments

- Total number of PR comments

- For each committer (identified by Anonymous GitHub User ID when possible), the number of commits, the # of LOC added & the # of LOC removed

- For each reviewer (referenced by Anonymous GitHub User ID), whether they approved and how many times they requested changes
- Pull Request Target Branch: If Main/master, list that; otherwise, refer to the anonymous branch ID

Pull Request Comment Data

- **Timespan**: 2011-2024
- **Unit of Observation**: Pull Request Comment-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID**, **Anonymous Pull Request ID** and **Anonymous Pull Request Comment ID** This matches to the data described in Section 4 (Pull Request Data) through **Anonymous GitHub Project ID** and **Anonymous Pull Request ID**. In this dataset, only pull request comments are described (not pull request review comments).

- **Anonymous GitHub Project ID**
- **Anonymous Pull Request ID**
- **Anonymous Pull Request ID Comment #**
- Pull Request Commenter Anonymous GitHub User ID
- Date & Time of Pull Request Comment
- Number of characters in pull request comment text (in the latest edit)
- Latest edit date & time of pull request comment
- Each Anonymous GitHub User ID that was mentioned in the pull request comment text
- Reactions on Pull Request Comment Text (as of data collection)

Pull Request Review Data

- **Timespan**: 2011-2024
- **Unit of Observation**: Pull Request Review-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID**, **Anonymous Pull Request ID** and **Anonymous Pull Request Review #** This matches to the data described in Section 4 (Pull Request Data) through **Anonymous GitHub Project ID** and **Anonymous Pull Request ID**. In this dataset, I describe aggregated pull request reviews

- Anonymous GitHub Project ID
- Anonymous Pull Request ID
- Anonymous Pull Request Review #
- Pull Request Reviewer
- Date & Time of Pull Request Review Completion
- Number of characters in pull request review
- Number of comments in pull request review
- # of pull request comments in pull request review
- Pull Request Review Result
- Each Anonymous GitHub User ID that was mentioned in the pull request review text

5 Push Data

- **Timespan**: 2011-2024
- **Frequency**: Monthly

- **Unit of Observation**: Branch-Project level
- **Observation Set**: All popular Python projects (see list)

Covariates

Each entry is uniquely identified by **Anonymous GitHub Project ID**, **Anonymous Branch ID** and **Date** This matches to the data described in Section 2 (Project Data) through **Anonymous GitHub Project ID** and **Date**. In this dataset, I describe aggregated pull request reviews

- **Anonymous GitHub Project ID**
- **Master, Main, or Anonymous Branch ID**
- **Date (Month-Year)**
- For each **Anonymous GitHub User ID** who committed to the branch in the past month,
  - Number of commits
  - Number of LOC added/deleted (cumulative)
  - Number of files added, deleted, changed & removed
- Whether the branch is protected