

The Merits of a Meritocracy in Open Source Software Ecosystems

Erwin Kaats
Information and Computing
Sciences
Utrecht University
Princetonplein 5, 3584 CC
Utrecht, The Netherlands
E.J.Kaats@students.uu.nl

Evert Eckhardt
Information and Computing
Sciences
Utrecht University
Princetonplein 5, 3584 CC
Utrecht, The Netherlands
A.E.Eckhardt@uu.nl

Slinger Jansen
Information and Computing
Sciences
Utrecht University
Princetonplein 5, 3584 CC
Utrecht, The Netherlands
Slinger.Jansen@uu.nl

Carina Alves
Center of Informatics
Federal University of
Pernambuco
Recife, Brazil
cfa@cin.ufpe.br

ABSTRACT

The Eclipse open source ecosystem has grown from a small internal IBM project to one of the biggest Integrated Development Environments in the market. Open source communities and ecosystems do not follow the standard governance strategies typically used in large organizations. A meritocracy is a frequently occurring form of governance on different levels in open ecosystems. In this paper we investigate how this form of governance influences the health of projects within the Eclipse ecosystem in terms of the amount of commits within each month. We analyzed the hierarchy of Eclipse, how merits are conceptualized within the ecosystem and the effect of the appointments of mentors and project leads on the amount of commits. From our research, we can conclude that this system is not always as fair as it seems; merits are only a benefit in some cases.

Keywords: Software Ecosystems, Open Source, Meritocracy, Ecosystem Health

1. INTRODUCTION

The Eclipse Project started in November 2001 by IBM Canada as a means to create the world's biggest open source Integrated Development Environment (IDE) platform. Unlike its competitors, like Microsoft Visual Studio, the entire project is run as an open source community and is directed by the Eclipse Foundation. Contributors who are active within the ecosystem and produce higher quality work are more likely to be offered to work at higher positions. Within this research we look at the effects of this so called meritocracy. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.
ECSAW, August 25 - 29 2014, Vienna, Austria
Copyright 2014 ACM 978-1-4503-2778-7/14/08...\$15.00.
<http://dx.doi.org/10.1145/2642803.2642810>

racy; and how it influences the health of the Eclipse projects. The term meritocracy is derived from the Latin word *mereo*, which means “earn” and -cracy, derived from Ancient Greek word *kratos* which means “strength, power”. We define meritocracy as *a system in which the talented are chosen and moved ahead based on the basis of their achievement.*¹

Since the founding of the Eclipse Foundation, an ecosystem has been formed by the many organizations and individuals that contribute to the projects governed by the Foundation. A software ecosystem is defined as: “*A set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts.*” (Jansen, Finkelstein, & Brinkkemper, 2009). The Eclipse platform facilitates a shared market for software and services, while managing the relationships between different projects and organizations.

Within the Eclipse Foundation, there are several membership tiers partnering that organizations can join, such as strategic, enterprise, solutions and associate members. Each tier has different benefits for the organizations, for example, voting rights and seats on councils and boards (Aarnoutse, Renes, & Snijders, 2014). Generally, organizations choose to either contribute to projects or base their own products on existing Eclipse projects. The Eclipse Foundation itself is governed by the Board of Directors and the projects and development processes are managed by a body of councils and committees. Within the individual Eclipse projects, there are one or more committers, which are individuals that commit code to projects. Most of these are affiliated or employed by member organizations but there are also individual contributors who voluntarily commit code. New projects also get one or two mentors to advise them during the project incubation phase. These mentors are often well respected veterans within the organization. Our research focuses on the system which governs this hierarchy. Indeed, Eclipse calls it-

¹<http://www.merriam-webster.com/dictionary/meritocracy>

self a meritocracy and as such, productivity should be key to advance in the ranks of the organization. An in-depth look at the inner workings of this system will grant a two-fold contribution: Eclipse may be able to improve their structure and organizations planning on adopting a meritocratic system can learn from the way Eclipse has conceptualized their merits.

While the previously mentioned Eclipse projects on the platform are maintained by the Architecture Council and project leads, the Eclipse marketplace is curated by the community. This marketplace consists of solutions, submitted by contributors with a range of functionality, from themes to complete APIs for different programming languages. While these marketplace solutions are hosted as ready-to-use packages; the repositories are not hosted by Eclipse. For this research we will focus on the Eclipse projects platform instead of the marketplace.

The paper is structured as follows; in section 2 the research approach is described. Section 3 offers an overview of the existing literature, focusing on the hierarchy of the Eclipse ecosystem, its structure and meritocracy. In section 4 the results of the data analysis are presented. Finally in section 5 a conclusion is drawn and the observed results are discussed in section 6.

2. RESEARCH APPROACH

To analyze the aspects of meritocracy in the Eclipse ecosystem, we will investigate if there is a correlation between the appointment of a mentor or project lead to a project on a certain date and the amount of commits to a project. As described by Jansen (2013), the first step of analyzing the health of an ecosystem is to assess the productivity, robustness, and niche creation within the platform. These factors and the definition of a meritocracy will be gathered from the official documents provided by the Eclipse Foundation regarding their governance mechanisms and established by laws. Note that download statistics and other platform and ecosystem wide data are not available within the repositories. We can, however, scrape various statistics from the website of Eclipse. As Eclipse is an open source platform, its repositories are free to be downloaded by anyone. These data storages do not only contain source code but also the metadata we are interested in.

The data was gathered from the dash sub domain of Eclipse, which is an Eclipse-ran database with statistical data on the activity in its own subversion and git repositories. For each project, we retrieved the amount of commits and their lines of code per project, per user, aggregated by month. With the support of a custom Python script, this data was cleaned and saved into a SQLite database. Calculations and plots were made using the SciPy suite packages. The scope for this research is largely restricted to the official Eclipse platform as the marketplace lacks a central source-code repository which makes it hard, if not impossible to scrape. In order to validate our findings and to avoid basing our conclusions on quantitative data alone, we also conducted an expert interview with one of the project leads in the Eclipse Ecosystem.

To analyze the effects of meritocracy on an open source ecosystem like Eclipse, we investigated the following research question:

What are the effects of meritocracy on the health of an open

source ecosystem?

To answer this question we analyzed the different aspects of how the Eclipse ecosystem is structured and how merits are conceptualized by answering the following sub questions:

1. *How is the Eclipse ecosystem structured in terms of hierarchy?*
2. *How does the Eclipse Foundation conceptualize merit?*
3. *Is meritocracy active on all layers of the Eclipse hierarchy?*
4. *What are the effects of appointing a mentor or a project lead to an Eclipse project on the health of that project?*

3. THE ECLIPSE ECOSYSTEM

3.1 The hierarchy of Eclipse

The Eclipse Foundation is governed by the Board of Directors, which consists of six community representatives and ten members from different companies such as IBM, Google and SAP. These companies are members of the Eclipse Foundation. Members determine the policies and the strategic direction of the Eclipse Foundation. The community representatives are elected by the projects committers, whereas the organizational members are appointed by the respective organizations.

The open source projects in Eclipse are guided by two councils: the Planning Council (PC) and the Eclipse Architecture Council (EAC). The Planning Council is responsible for coordinating the Simultaneous Release, which is the annual release of Eclipse. Projects can opt to this release, which results in reducing the redundancy of, for example, the same library used by different projects. The EAC is responsible for several aspects of the Eclipse Development Process², including the long-term technical health. They revise the processes, monitor the project architectures and assist project committers by mentoring them. Currently there are 55 active members in the EAC.

In addition to the project mentors, the projects are guided by the Project Management Committee (PMC) and the project lead. The PMC manages the top level projects and are responsible for the right execution of the Eclipse Development Process. The leads of the PMC have to be approved by the Eclipse Foundation Board and the Eclipse Management Organization (EMO), and PMC members are nominated and voted on by the current Leads and Members of the committee.

Project leads manage the projects and their sub projects. They ensure that their projects are following the Eclipse Development Process and that their users, adopters, committers and contributors create a community. A project lead has to be appointed during the creation process of the project. If a new Lead must be appointed, the committers and contributors of the projects elect a new Lead which subsequently must be approved by the PMC.

Each project consists of a Development Team steered by project leads. The Development Team consists of contributors and committers. The contributors are individuals who work on the project in terms of testing, fixing, documenting

²<http://tinyurl.com/eclipsedevprocess2014>

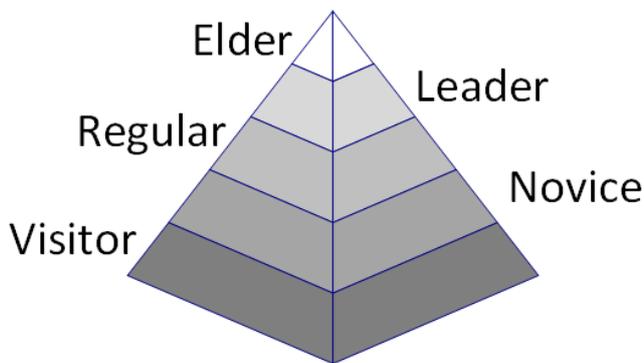


Figure 1: Pyramid of meritocracy and role hierarchy

and contributing code to the project. Committers are individuals who have access to project resources, which include bug tracking, downloads, build servers, etc. As stated in the Eclipse Development Process: *“Becoming a committer is a privilege that is earned by contributing and showing discipline and good judgment. It is a responsibility that should be neither given nor taken lightly, nor is it a right based on employment by an Eclipse Member company or any company employing existing committers”*. This indicates that a meritocracy is used at the very core of the ecosystem, the actual projects.

In order for a contributor to become a committer, an existing committer on the project has to nominate the contributor, which happens if a contributor is active enough within the project. If there are enough positive votes from the other committers within a certain timeframe, the PMC is asked for approval of the promotion. If the PMC approves, the contributor is promoted to committer and granted writing access to the source-code of the project.

All committees, leads and councils are managed by the Eclipse Management Organization (EMO), which selects the chair of both councils and leads in the execution and maintenance of the Eclipse Development Process. With this information, our first sub-question is answered.

3.2 Ecosystem Health

Ecosystem health can be calculated by looking at productivity, robustness and niche creation (Iansiti & Levien, 2004), which was further elaborated in Jansen (2013) by dividing metrics into network and project levels. Eclipse facilitates niche creation in several ways. On a network level, there are several committees and councils which consist of individual and organizational members. At the base of the ecosystem are the projects, which are carried out by committers and contributors. As mentioned in the previous section, the task of the project leads is to create a community, a niche, with their contributors and committers to create and maintain a healthy project.

Another way to stimulate niche on a network level is via organizing conferences all over the world with meetups, speakers and workshops where members from the community can meet each other in real life. These EclipseCons can facilitate the formation of more vibrant communities around projects. In the expert interview, it was indicated that the reason they joined the foundation as a member was that they were convinced during one of these conferences by some of the board members. The networking they have done on conferences like these has helped them further their own company within the Eclipse ecosystem.

On a project level, productivity can be measured in lines

of code and how well the committers are doing according to the project plan for their project. On a network level, the productivity of projects can be measured in the amount of code released to the public. Projects can opt in to new yearly Eclipse releases that are managed by the Planning Committee.

To guarantee the robustness of projects, SonarQube³ is used. SonarQube is an open-source platform on which code quality analysis can be done. The platform takes things like duplication, bugs, standards, complexity and documentation into account to manage the quality of a project’s code. Committers can receive code contributions from Contributors through the standard communication channels such as forums, mailing lists and the bug reporting system. The quality and appropriateness of the code has to be checked before it is added to the project, based on the significance of the contribution which is defined as at least 250 lines of code that adds new functionality to the codebase. If a contribution is significant, it has to be approved by the PMC and the EMO before it is added to the project.

3.3 Meritocracy Factors

O’Mahony and Ferraro (2007) researched the open source Debian community in terms of governance. They state: *“Any examination of meritocracy must develop a context-specific understanding of how merit is conceptualized”*. In this section we answer our second sub-question by analyzing how merit is conceptualized within the Eclipse ecosystem on the different hierarchical levels. The term meritocracy in combination with software ecosystems can be found in Kim (2000). Scacchi (2003) describes the term pyramid meritocracy as a dynamically organized virtual enterprise. The Eclipse ecosystem takes on similar forms with the board of directors as the so called *elders*, the councils and committees as the *leaders* and the committers and contributors as the *regulars*.

As mentioned before, the Eclipse Development Process states that new committers are nominated from a pool of active contributors and elected by the existing committers from the same project. This is where contributors are really selected based on their quality and commitment to the project, which can be seen as the base for the meritocracy. An added benefit that comes with the committer status is that you gain a vote in the Committer Representatives elections.

If we take one step up the organizational ladder and move to the leaders in Scacchi’s pyramid, depicted in Figure 1, there is the appointment of leadership of a project. The project lead is again nominated by the existing committers, who discuss the qualifications of the committer in question in order to determine if he or she is the best fit for leading the project. Within the PMC, new committee members are nominated by the current members and chosen through unanimous votes. For the elders of the pyramid, there are different procedures. The Board of Directors of the Eclipse Foundation consists mostly of allocated seats for the member organizations, where elections are held each year by the organizations themselves. New member nominees are selected based on their capabilities, as stated in the Bylaws of the Eclipse Foundation.⁴ The Community Representatives are selected by the members of the community with vot-

³<http://www.sonarqube.org/>

⁴[http://www.eclipse.org/org/documents/Eclipse BYLAWS](http://www.eclipse.org/org/documents/Eclipse_BYLAWS)

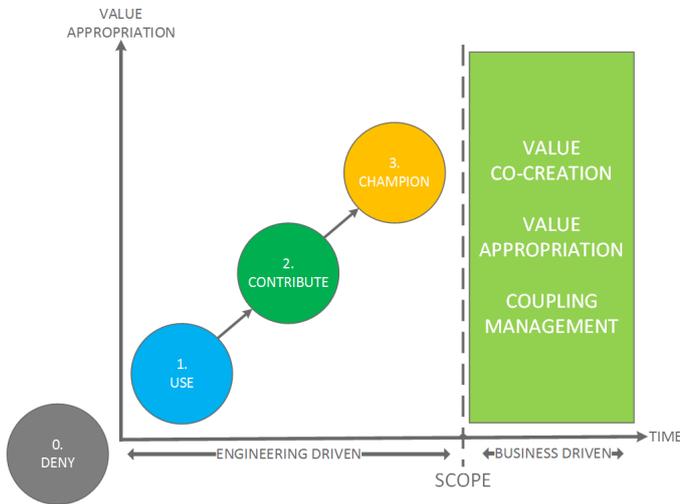


Figure 2: The Eclipse Maturity Model

ing rights, such as the committers and project leads. The steps as described by Scacchi (2003) can also be found in the Eclipse Maturity Model⁵, depicted in Figure 2. In this model, three steps are given: (1) Use, (2) Contribute, (3) Champion. The interviewed expert compared these three steps to contributors, committers and project leads. It shows the growth of a user of the ecosystem in a way a meritocracy works - by actively contributing to a project, a user gains more and more rights and respect in that project and can work his way to the top.

If a project has no active committers and inactive or a non-effective project lead, it is possible for the PMC to take over on behalf of the Eclipse Foundation to manage that project. When committers become inactive for longer than six months, they are removed from the project and given an Alumni status⁶. If a certain committer has contributed greatly to the success of Eclipse, he or she is granted the Committer Emeritus status.

A project lead can nominate a committer for this status when removing the person in question from the project. This is one of the merits Eclipse gives its active contributors, a recognition of their valiant efforts in the flourishing of their ecosystem. Another example of rewarding individuals is the Eclipse Community Awards, a yearly event where titles like 'Top Committer' and 'Top Newcomer Evangelist' are given to active individuals of the community. Awards like these can stimulate contributors and committers to be as productive as possible in the stimulation of the ecosystem.

The Eclipse ecosystem certainly has several aspects of a meritocracy embedded into its roots, such as the elections at the bottom of the meritocracy pyramid. But the higher up the chain of governance we look, the more we see the formation of a more typical form of governance. This means the answer our third sub-question is that a meritocracy is not active on all layers of the hierarchy.

4. RESULTS & ANALYSIS

To analyze the effects of meritocracy on the health of the Eclipse ecosystem, we look at how mentors and project leads influence the health of projects within the Eclipse ecosystem.

2011_08_15_Final.pdf

⁵Copyright 2012, Eclipse Foundation, Inc. Made available under the Eclipse Public Licence 1.0

⁶<http://www.eclipse.org/projects/committers-alumni.php>

A mentor is a member of the Eclipse Architecture Council (EAC) and helps in the steering of the project by tutoring committers when they have questions. Furthermore, they make sure everything is handled according to the project plan. Because mentors are typically users that have been active for a longer period of time and have proven to be capable of giving advice to other people, it would be interesting to see how mentors affect the health of a project after they have been appointed. Project leads are in charge of the daily workings within the project, they are the oil between the cogs and play more often than not, a supporting role. We first take a look at mentors after which we analyze the data on project leads.

4.1 Mentors

Currently there are 55 members in the EAC, of which 36 are mentoring at least one of the Eclipse projects. Most of these members do not have a very specific appointment date, since only the appointed year is given. However, for 14 of these 55 members the assigned month is given.

We assume that the mentors have taken on mentorship of these projects at the time of their appointment or directly at the start of the project in some cases. What is apparent when looking at the amount of commits of these 14 EAC members is that there is not a real visible trend to be found. Some of them have barely committed anything over the course of their activity, while others have committed a significant number of times. For the latter part of the members there is another distinction to be made: while some continue committing regardless of their new position, some of them stagnated after their appointment on the AC.

To gain an overview of the effect of a mentor on a project, we will now look at the average amount of commits before and after the assignment of the mentor to the project. The results are summarized in Table 1. An unexpected result of this analysis is that out of the 37 projects that are mentored by the selected mentors, only 16 of them have actually committed something to the project. Many of the empty projects are in the beginning of their lifecycle and are still in the progress of being set up.

4.2 Project leads

Besides analyzing the mentors, we also looked at the productivity of project leads. A sample from the data is depicted in Table 2, it is based on data obtained from the EMO and sorted according to project name and project lead. Each row has the average amount of commits before, after and during the appointment of a project lead. The after column is only filled if that project lead has ceased his or her activities as lead of the project. The total column reflects on the average amount of commits during the runtime of the projects. The change column shows if the project was less productive ('<'), equally productive('=') and more productive ('>') during the assignment of the project lead.

While mentors only help to start a new project, existing projects always need one or more project leads to function. While this fact by itself is hardly enough evidence to state a project leads' importance, it is an interesting trend. Furthermore, project leads who managed a project poorly tend to have managed other projects within the ecosystem. Those who managed more than three projects tend to do so successfully, however, an ill-performing project could also indicate a lack of tutoring after leaving a project. To answer our

| Projects | Average # of commits (per month before) | Average # of commits (per month after) | Average # of commits over runtime of project |
|------------------------|--|---|---|
| technology.m2e.m2e-wtp | 0 | 60 | 60 |
| technology.tycho | 0 | 224 | 224 |
| technology.handly | 0 | 0 | 0 |
| polarsys.polarsys.3p | 0 | 12 | 12 |
| rt.concierge | 0 | 85 | 85 |
| technology.kura | 0 | 4 | 4 |
| technology.smarthome | 0 | 501 | 501 |
| webtools.vjet | 0 | 3110 | 3110 |
| datatools.incubator | 14 | 6 | 9 |
| modeling.mdt.rmf | 0 | 326 | 326 |
| modeling.sirius | 0 | 890 | 890 |
| modeling.upr | 0 | 214 | 214 |
| technology.xwt | 0 | 131 | 131 |
| technology.skalli | 205 | 42 | 191 |
| technology.tycho | 238 | 71 | 224 |
| modeling.emf-parsley | 0 | 356 | 356 |

Table 1: Projects and the amount of commits before and after the appointment of a mentor

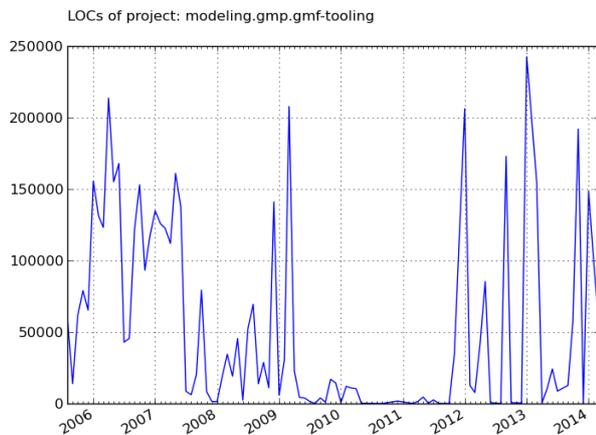


Figure 3: Commits of the Graphical Modeling Project (GMP) project

fourth sub-question, we can say that general trends are difficult to either find or to trace back to the effects of a meritocracy. Projects generally do not express large changes, noise excluded.

Some project like ‘modeling.gmp.gmf-tooling’ (Figure 3.) shows a standstill for a couple of years but whether that is a deliberate move or the remnants of an incompetent project lead is not clear from the numbers. Likewise, the data does not show a correlation between the commits of a project lead and that of the overall project. Manually looking at the data, we did not identify any fluctuations that might indicate a correlation between these data points. The expert interview confirmed our conclusions. According to the expert, most aspects of meritocracy are actually visible before people get to committer status. As anyone can fill out bug reports or submit a patch to be reviewed by a committer, the actual philosophy of a meritocracy is only really visible at this stage. On the other hand, once someone becomes a board or council member or even a project lead, they are likely to stay at that position for as long as they want to

continue in this role. Within a true meritocracy, one would expect people to perform less than their subordinates to be succeeded by them. For project leads this is also partly true. Although our expert source did confirm Eclipse’s own statement that a project lead is a peer-acclaimed title, he could not recall an event in which a project lead actually was forced to step down from their position.

In the interview, the expert also mentioned that people mostly become project lead because they work for an organization that has a certain code base to share and wants to develop it under the umbrella of Eclipse. While one could argue that ‘donating’ this code is the same as submitting code to existing projects, it cannot be seen as peer acclaim as only the councils make the decisions for allowing new projects to form, not the committers.

5. CONCLUSION

After measuring the results of the Eclipse ecosystem - in our case the amount of commits the developers committed to the repository - we came to a conclusion which might not be surprising. In order to assess the merits one cannot simply measure one aspect, just as one cannot measure the health of a tree by simply measuring the amount of water that falls down on its leaves. We noticed that while most project leads and mentors had a good commit-track record, there were also a fair amount that had barely written any code for the Eclipse project, yet those individuals might contribute in other ways. A tree needs sunlight and proper soil as much as it needs water. Likewise, individuals might help the Eclipse ecosystem growth by managing projects or by lobbying. There are a myriad of ways to improve and measure the health of an ecosystem and thus, measuring the merits of a meritocratic project structure by only looking at the leaves sprouting on its trees is not enough to draw a conclusion. Furthermore, the expert interview revealed that the work of the contributors can only be traced by looking at bug reports and patch comments. One might think a meritocracy will ensure project leads and mentors are the ones delivering the most meaningful contributions, but the

| Project | Project lead | Before | After | During | Total | Change |
|----------------------|--------------------|--------|-------|--------|-------|--------|
| ... | ... | ... | ... | ... | ... | ... |
| rt.gemini.blueprint | Glyn Normington | 194 | 33 | 71 | 110 | > |
| rt.gemini.blueprint | Costin Leau | 11 | 60 | 214 | 110 | > |
| rt.gemini.blueprint | Dmitry Sklyut | 123 | | 35 | 110 | = |
| rt.gemini.dbaccess | Mike Keith | 39 | 39 | 20 | 38 | < |
| rt.gemini.dbaccess | Juergen G. Kissner | 34 | | 39 | 38 | = |
| rt.gemini.management | Christopher Frost | 47 | | 26 | 39 | = |
| rt.gemini.management | Peter Peshev | 32 | 26 | 52 | 39 | > |
| rt.gemini.management | Hal Hildebrand | | 40 | 32 | 39 | < |
| rt.gemini.naming | Bob Nettleton | | 27 | 15 | 19 | < |
| rt.gemini.naming | Violeta Georgieva | 15 | | 27 | 19 | = |
| rt.gemini.naming | Mike Keith | 15 | | 27 | 19 | = |
| rt.gemini.web | Glyn Normington | | 28 | 19 | 24 | < |
| rt.gemini.web | Violeta Georgieva | 19 | | 28 | 24 | = |
| rt.gyrex | Gunnar Wagenknecht | 244 | | 161 | 167 | = |
| rt.jetty | Jesse McConnell | 227 | | | 227 | = |
| rt.jetty | Jan Bartel | 442 | | 224 | 227 | = |
| ... | ... | ... | ... | ... | ... | ... |

Table 2: Sample of project lead data with commit averages in months

real proof of work is delivered by contributors before even entering the committers' lists.

6. DISCUSSION AND FUTURE RESEARCH

During our research we found that constructing an overview of the ideal structure of a meritocracy seems quite easy. The Eclipse Foundation is very open on their goals and discussions with an Eclipse member enabled us to get a good idea of how things really work. This, however, are personal opinions and may not tell the complete story. To further research these dynamics, we looked at the data on commits and lines of code added. After all, these are the primary metrics of work within software projects. The data, however, proved to be incomplete and often unreliable. As such, we were not able to validate our findings in a qualitative manner. Future research can be focused on finding more reliable data on the projects and measuring not only the health of projects by the amount of commits, but the health of the entire ecosystem as a whole. With this research, we have provided an overview of the Eclipse ecosystem meritocracy structure as a whole and provided some results to assess the health of this ecosystem. Another possibility for future research can be comparing multiple meritocratic open source ecosystems to gain an overview of the conceptualization of merit, and possibly generalize these merits.

References

- Aarnoutse, M., Renes, C., & Snijders, R. (2014). Loyalty and Migration of Contributors in Software Ecosystems. *Proceedings on workshop Software Ecosystems, Utrecht University.*
- Iansiti, M., & Levien, R. (2004). Strategy as ecology. *Harvard business review*, 82(3), 68–81.

- Jansen, S. (2013). Measuring the Health of Open Source Software Ecosystems: Moving Beyond the Scope of Project Health.
- Jansen, S., Finkelstein, A., & Brinkkemper, S. (2009). A sense of community: A research agenda for software ecosystems. In *Software engineering-companion volume, 2009. icse-companion 2009. 31st international conference on* (pp. 187–190).
- Kim, A. J. (2000). *Community building on the web: Secret strategies for successful online communities*. Addison-Wesley Longman Publishing Co., Inc.
- O'Mahony, S., & Ferraro, F. (2007). The emergence of governance in an open source community. *Academy of Management Journal*, 50(5), 1079–1106.
- Scacchi, W. (2003). Free/open source software development practices in the computer game community.