On the analysis of contributions from privileged users in virtual open communities

Felipe Ortega, Daniel Izquierdo-Cortazar, Jesus M. Gonzalez-Barahona and Gregorio Robles
GSyC/LibreSoft
Universidad Rey Juan Carlos
Tulipan sn, 28933, Mostoles, Madrid. (SPAIN)
{jfelipe,dizquierdo,jgb,grex}@gsyc.es

Abstract

Collaborative projects built around virtual communities on the Internet have gained momentum over the last decade. Nevertheless, their rapid growth rate raises some questions: which is the most effective approach to manage and organize their content creation process? Can these communities scale, controlling their projects as their size continues to grow over time? To answer these questions, we undertake a quantitative analysis of privileged users in FLOSS development projects and in Wikipedia. From our results, we conclude that the inequality level of user contributions in both types of initiatives is remarkably distinct, even though both communities present almost identical patterns regarding the number of distinct contributors per file (in FLOSS projects) or per article (in Wikipedia). As a result, totally open projects like Wikipedia can effectively deal with faster growing rates, while FLOSS projects may be affected by bottlenecks on committers who play critical roles.

1. Introduction

Over the last decade, we have witnessed the advent of radically new trends regarding the way virtual communities start up and evolve on the Internet. The so-called Web 2.0 technologies [27] have definitely changed the way we conceive and implement collaborative projects. Forums, blogs, RSS channels, mashups, wikis, etc. have moved the focus of the content creation process from webmasters and service providers to final users, unleashing the real power of a worldwide network connecting a global village.

Collaborative organization has been a key feature of many of the most important collective initiatives and projects along the Internet history. Almost all of them, though, have been focused on the development of Free, Libre and Open Source Software (FLOSS): GNU/FSF, Linux, Apache, Mozilla, etc. are prominent examples of these open, collaborative initiatives, whose working philosophy was described in detail in a seminal work by Raymond [30], and further explored by Coffin [9]. However, Web 2.0 technologies expanded the range of these collective initiatives to include virtually any kind of intangible content that we can represent in the digital world: images, sounds and music clips, video content, information repositories, help forums, personal blogs, and even universal encyclopaedias like Wikipedia (http://www.wikipedia.org).

Some authors [14], [4] defend the thesis that this innovative approach in collaborative virtual communities will continue to increase its influence in due course. James Surowiecki explores in [37] the implications of collaborative content creation initiatives in the way we understand collective intelligence, and how it shapes and affects many aspects of our new networked reality. For sure, large-scale projects such as Wikipedia, involving millions of users contributing in many different languages, compiling more than 8 million articles in its top-ten language editions, challenge any limits previously foreseen by the original creator of the wiki concept [22].

Presently, open content initiatives face some of the same evolution paths already followed by FLOSS development projects some years ago. In this context, some natural questions show up: can we find any similarities between FLOSS development projects and collaborative, open content creation initiatives? Does the non-technical nature of open content cause the attraction of a higher number of contributors? Does it affect the frequency and distribution of contributions from project participants? Can these projects scale effectively, in order to avoid becoming victims of their own success?

In this paper, we tackle these questions analyzing contributions from privileged users in FLOSS development projects, as well as in Wikipedia, the epitome of open content creation initiatives at present time. With privileged
users, we mean contributors that have shown a high level of commitment to these collective initiatives, and who have been recognized as such by their respective communities, following what Coffin identifies as “a hybrid political system based upon meritocracy” [9].

Modelling contribution patterns from these high reputation authors, we present an initial comparison of both kinds of collaborative projects. This will help to better describe this new wave of virtual initiatives, in which final users become the key factor of the content creation process. It is expected that some differences will be observed in activity patterns, due to the different focus and goals (development of software on the one side, and creation of content on the other side). However, by understanding how virtual communities like Wikipedia, attracting millions of users, can scale to supervise their own workflow, we can provide useful clues about strategies that successful FLOSS development projects may try to manage larger communities of contributors.

We conclude this section presenting the organization of this paper. In the first place, we summarize some previous research works that analyze virtual communities in FLOSS projects, as well as in Wikipedia. In the second place, we describe our own quantitative methodology to compare privileged contributors in FLOSS and open content development projects. Then, we present our results and discuss its implications to answer the research questions we stated above. Finally, we summarize the conclusions that we can infer from this quantitative analysis, as well as possible future lines of work on this research area.

2. Previous research

In this section, we present some previous research works concerning communities of users in FLOSS development projects and collaborative open content creation initiatives, with special attention to those publications analyzing contributions from privileged users in both areas.

Most of times, libre software community-driven projects are lead by volunteers. They have a common goal and, generally speaking, their main purpose is to attract people to the community. The onion-model described by Crowston et al. [12]. [11] shows a first approach to describe the structure of libre software communities, in which the size of their core developer team, developers team and community of users increase by a factor of an order of magnitude in each layer.

Focusing on social network analysis, Lopez et al. [23] study the network formed by authors at the module level. In this model, an arc between two authors represents that they have worked in the same module, and the links weight accounts the number of commits in the same module. We will follow a similar methodology in this paper, but with a finer granularity level.

Also, general books related to libre software projects describe the management tasks in these group teams. On one hand, a classic approach is given by DeMarco [13]. On the other hand, we find the FLOSS point of view, such as Fogel [16] or Sandred [35].

Focusing on specific libre software projects, Michlmayr analyzes how the Debian project manages volunteer activity [25], whereas German [17] shows how developers work in a libre software project. In a different research work [18], German also shows empirically that there exist a common contribution pattern in FLOSS development projects, where developers tend to contribute to specific pieces of software. This behavioral pattern has been named “developer territoriality” since, like many other species in nature, developers maintain their own “territory”. Moreover, regarding team management Robles et al. find in [33] that there exist a periodic relay of developers in FLOSS projects, showing a pattern where generations of developers successively take the lead of the project for a timespan ranging from two to four years.

Concerning previous research studies about the Wikipedia community of authors, one of the first contributions on this topic is the quantitative analysis on the German Wikipedia, undertaken by Jakob Voss in [40]. In this paper, Voss finds that the number of different authors per article follows a power law, while the number of distinct articles per author follows Lotka’s law. This law is a subclass of Zipf’s law, describing the frequency of publications made by authors in a certain field [24]. Anthony et al. find in [2] two different types of authors who contribute high quality content to the project. On one hand, we have zealots, registered users frequently contributing to Wikipedia with high quality content, possibly with the main purpose of gaining a higher reputation from other community members. On the other hand, we have good Samaritans, casual contributors that only made sporadic edits to Wikipedia articles, but high quality ones.

Kittur et al. present in [21] a detailed study of contribution patterns of frequent and infrequent editors to the English language edition of Wikipedia, including a quantitative analysis of contributions coming from Wikipedia sysops (privileged users with a special status granted by the rest of community members). Ortega et al. further explore in [28] this line of research, finding that Wikipedia administrators do not present such a high activity level like the one maintained by other contributors, at least regarding edits to Wikipedia articles. Moreover, they find that there is a strong correlation in each language edition between the frequency of contributions from administrators and the policy established to select and renew the sysop status in that language edition.

Some authors have focused on the analysis of the rep-
utation level of authors (a measure for the trustability of their contributions to the project). In this way, Adler and de Alfaro present in [1] an automatic tool to infer the reputation level of Wikipedia editors, based on the content that have been added by those authors, and have not been later changed or removed by subsequent authors.

Another interesting research line is the analysis of role distribution among Wikipedia community members. Ciffolilli finds in [8] evidences of a self-selective recruitment process applicable to Wikipedia community members, as well as clear patterns for retaining interested contributors, who usually focus on specialized project tasks. Spek et al. also find in [36] the presence of a clear trend of self-selection of contributors roles in Wikipedia. This hypothesis has been confirmed by Wattenberg et al. in [42], using a visual technique based on chromograms. These graphs color users contributions over time according to the specific tasks (content addition, correcting typos, arbitration, discussion on talk pages, reverting vandalism acts...) undertaken in each of them.

Wikipedia has been also analyzed from a social network perspective. Nevertheless, most of the previous research publications in this area are focused on the study of Wikipedia web graph, that is, the construction of a network graph whose nodes are Wikipedia articles and whose links represents hyperlinks connecting those articles. This is the methodology followed in [3], [6] and [5].

Finally, we might question if there exists a theoretical framework supporting the comparison of two apparently different kinds of initiatives like FLOSS development projects and open content development projects (represented by Wikipedia in this paper). Coffin provides in [9] a thorough list of traits characterizing open initiatives involving virtual communities, disregarding the nature of their final products. It is interesting to notice how basic organizational elements such as geographically distributed, asynchronous networked collaboration, project transparency and openness, production of living documents and project artifacts, the community-wide sense of project ownership, the use of consensus as the principal decision-making tool and even the presence of a trusted benevolent dictator [32] are all shared by initiatives falling in both areas. An empirical study [34] demonstrate that the activity level of contributors in FLOSS projects is directly influenced by their level of exposure to the community activities (disregarding possible effects of personal preferences, as we might expect from intuition). As we stated previously, in [28] we can find evidences of a clear correlation between the activity level of Wikipedia sysops and certain policies established to incentivize their level of contribution to the project in order to maintain their status. Therefore, comparing activity patterns identified in FLOSS development projects and Wikipedia we can gain knowledge about different strategies to manage distinct evolution scenarios over time.

In the following section, we present details about the methodology we have followed to implement our quantitative analysis of privileged users in FLOSS development projects and Wikipedia.

3. Methodology

In the following, we describe the data sources we have used to implement this research work, as well as our tools for automating the data retrieval process and its classification. Then, we present some definitions and assumptions that we have considered in this study.

3.1 Data Sources and Tools

To implement the quantitative analysis in this paper, we have utilized two different data sources. On one hand, we have obtained the data for libre software projects from the FLOSSMETRICS database (http://fm3.libresoft.es/retrieval_system/). In this paper we will use the term “libre software” to refer to any code that conforms either to the definition of “free software” (according to the Free Software Foundation) or “open source software” (according to the Open Source Initiative). The main goal of this database is to provide to the scientific and industrial community a public platform with information and metrics about libre software development projects. On the other hand, we have extracted Wikipedia data from the official database dumps, publicly available on the Wikipedia web download center (http://download.wikimedia.org). These files contain a complete, compressed dump of the whole data history for every wiki page in a certain language edition. For this study, we have downloaded the most up to date version of the database dump of each language edition as of March 31, 2008.

The FLOSSMETRICS data repository already provides quantitative data from FLOSS development projects in an organized way, suitable for being employed in research works like this one straightforwardly. However, database dumps from the Wikipedia public repository need to be first decompressed and processed to extract and organize quantitative results about its workflow and content in a similar way. WikiXRay is a Python software tool developed at our research group (http://developer.berlios.de/projects/wikiXRay/) to automate this process. It takes the compressed database dump for any Wikipedia language edition, and creates a database with quantitative information describing it.

With respect to the analysis of these data, in FLOSS projects only committers are authorized to upload changes to the project repository, so we have considered the whole
set of changes stored in the Code Management System (CVS or SVN) used in each one. In Wikipedia, though, all users can edit articles disregarding their privilege status. Even anonymous contributors may change the content of Wikipedia articles. Therefore, in order to utilize comparable sets of contributions, we have filtered out all edits in each Wikipedia language edition that did not come from sysops.

Finally, to obtain statistical results and graphs summarizing our quantitative analysis, we have used GNU R [29], a well-known, powerful and libre statistical software package, providing many routines and additional packages from contributors all over the world, accessible from the Comprehensive R Archive Network web page (http://lib.stat.cmu.edu/R/CRAN/).

3.2 Definitions, Assumptions and Hypothesis

Some of the analyses that we implement in this research paper involves the study of the distribution (or dispersion) of contributions from users to these open projects. To summarize this information, we use the Gini coefficient, first proposed by Corrado Gini in [19], which is a very well-known measure of the inequality of distribution of income and other quantitative factors among the members of a certain population. It has been successfully applied to measure inequality in different research areas such as Economics [15], [26], Education [38], and Health Sciences [41], [7]. It is known that the Gini coefficient, as well as other alternative statistical measures of inequality in populations (like the skewness coefficient or the coefficient of variation), presents some undesirable properties. In spite of this, these properties are usually irrelevant when we apply them on real data, following the usual two-parameter lognormal model for the distribution of the variable under study [31].

Its values are restricted to the closed interval [0, 1]. A Gini coefficient \( g = 0 \) represents perfect equality, that is, a situation in which the contribution from every member is exactly the same as the contribution from the previous one. A Gini coefficient \( g = 1 \) represents the extreme inequality (only one member concentrates all contributions, while the rest ones do not contribute at all). Therefore, Gini coefficients next to 0 shows a high equality level, while Gini coefficients tending to 1 presents a high level of inequality in the distribution of a certain parameter.

Nevertheless, we have to be very careful when we apply this coefficient to small populations, since in these cases the value we obtain can be inaccurate. Therefore, we have restricted our analysis to those FLOSS projects and Wikipedia language editions with a community of privileged contributors with (strictly) more than 30 members. This pre-condition restricts this analysis to 16 language editions of Wikipedia (including the top-ten, regarding their total number of articles) and 123 FLOSS development projects from the FLOSSMETRICS database.

We have considered the following definitions to implement our quantitative analysis of privileged users in FLOSS development projects and Wikipedia:

- **Committer**: The committer is a privileged user in a FLOSS development project, who has access to the libre software repository and can modify the source code and other files.
- **Commit**: A commit registers the changes made by a committer in the source code or other files.
- **Sysop**: The sysop (also known as administrator, or admin) is a privileged user in a Wikipedia language edition who has special permissions to control the content creation process on Wikipedia articles. In particular, they can delete pages, block authors and undo both actions.
- **Edit**: For this quantitative study, we consider as an edit any change made in a single file of a FLOSS development project, or any change on the content of a single article in a Wikipedia language edition.

On top of that, the following assumptions have been taken into account:

- **Assumption #1**: We have considered files and articles as the fundamental unit of content. Thus, there is an analogy between files in libre software projects and articles in Wikipedia, in the sense that an edit in a FLOSS file can be identified with an edit in an Wikipedia article.
- **Assumption #2**: There is an analogy between committers in libre software projects and sysops in Wikipedia. We should take into account that not only the libre software committers are the authors of a project. There are dozens of people who send patches to the project, but they do not have commit permission and they are not registered, although they are authors of that code. Thus, the committers are those who filter all the contributions given by authors who have not reached the committer status. Regarding the Wikipedia, sysops are those contributors in Wikipedia specialized in reviewing articles, reverting vandalism, and controlling articles content and contributions from standard users. Thus, they act as a filter for all the contributions made by other non-sysops authors.
- **Assumption #3**: Considering assumption #2, only those edits made by committers in FLOSS development projects, and made by sysops in Wikipedia, will
be included in the quantitative analysis that we present in this paper.

- **Assumption #4**: Considering assumptions #2 and #3, as we stated above we have to filter out all FLOSS projects and Wikipedia language editions which do not have strictly more than 30 privileged users, if we want to accurately measure the inequality level of contributions by means of the Gini coefficient.

In this research work, we are not trying to formally validate any specific hypothesis regarding the populations under study. All the same, the overall statistics from Wikipedia and FLOSS development projects tell us that Wikipedia must deal with a user community several orders of magnitude larger than that from the FLOSS development field. In this sense, we are particularly interested in characterizing the activity patterns of both types of projects in terms of the distribution of work among privileged users. Wikipedia seems to be capable of creating a larger group of outcome artifacts (articles in this case) due to the main goal pursued by its community (broadening the coverage of the encyclopaedia as much as possible). In recent years, we have seen many examples of FLOSS development projects threatened by their own level of success, struggling to absorb peaks of workload within an already overloaded group of project committers. Activity patterns identified in both types of projects may spot some light about the most appropriate approaches to deal with virtual communities growing at faster rates.

4. Results

We present here the results of our quantitative analysis of privileged users in FLOSS development projects and Wikipedia. To start with, as we stated above we have included in this analysis only those FLOSS projects and Wikipedia language editions with, strictly, more than 30 privileged users. However, it is interesting to analyze the distribution of the number of privileged users in these collaborative development initiatives. Figure 1 presents the Kernel Density Estimation of this parameter in both types of projects. The KDE is a convenient approximation for the histogram graph of a certain statistic variable for those cases in which we want to compare two or more curves in the same plot [10]. The KDE can be informally considered as an interpolation of the histogram graph, in which the KDE curve is depicted joining the middle points of each probability density bar in the histogram plot. Thus, for example both KDE curves should start at \( \# \text{Privileged users} = 30 \), which corresponds to the highest peak of both graphs on the left side of the plot. The remaining part of the curve on the left of that value is just a product of the construction algorithm for the KDE curve in GNU R (for additional details about the implementation of this algorithm see [39]).

As we can see, both KDE curves show that the majority of FLOSS projects (dashed line) and Wikipedias (solid line) present less than 200 privileged users. However, we can also identify emerging subpopulations of projects that are starting to leave the main body of both populations. The peak on the far right side of the solid line curve corresponds (as we have labeled in the graph) to the English Wikipedia. This language edition has achieved to attract 1,456 sysops over its 7 years of history. Therefore, we expect it to behave in a much different way than other collaborative initiatives, since at present time, it has a “critical mass” of users much larger than any other project. If we focus now on the tail of both curves on the left side of the graph, we appreciate that some language editions in Wikipedia and several FLOSS projects are beginning to acquire enough “critical mass” as to start to separate themselves for the rest of the “standard population” of projects. That is the case of the German Wikipedia (corresponding to the peak separating from the tail of the main body of the Wikipedia density curve) and some FLOSS projects like gcc and mono, which are beginning to split from the main curve of the FLOSS projects KDE graph as well, towards the right side.

We now turn to the analysis of the inequality level of contributions from privileged user in FLOSS projects and Wikipedia. Figure 2 shows the KDE curve representing the probability distribution of Gini coefficients in FLOSS projects and Wikipedia.

Here, we can visually compare the values of the Gini coefficients calculated in both types of projects. Surprisingly, both curves present us a completely different pattern for the
KDE comparison of Gini coeffs. in FLOSS projects and Wikipedias

Figure 2. Comparison of KDE of Gini coefficients for privileged contributors in FLOSS development projects and Wikipedias

Lorenz Curve and Gini coeff. for admins de

Figure 3. The Lorenz curve and Gini coefficient value for contributions from privileged users in the German edition of Wikipedia

distribution of contributions. The left side curve (solid line) shows us that contributions from Wikipedia privileged users have a high equality level (most Gini coefficients ranging from $g = 0.2$ to $g = 0.4$). As a result, all privileged contributors collaborate with a similar level of effort to the project (regarding the number of edits they have made). On the contrary, the right side curve (dashed line), corresponding to contributions from committers in FLOSS development projects, reveals us a high level of inequality in their contributions (most Gini coefficients ranging from $g = 0.7$ to $g = 0.9$). Even in a language edition like the German Wikipedia, with a high number of privileged users, we obtain an egalitarian level of contribution, graphically represented in Figure 3, by means of the Lorenz curve. This curve depicts the cumulative percentage of contributions from each privileged user. In this graph, the straight line represents perfect equality (the case in which every user would contribute the same as the previous one). The Gini coefficient actually computes the area between the Lorenz curve and the line of perfect equality: the larger the area, the higher the inequality level of contributions from privileged authors. As the Lorenz curve in this case is very close to the line of perfect equality, we can infer that Wikipedia sysops exhibit a similar workload pattern regarding their number of contributions to the project.

In order to put in context our results about the inequality of contributions from privileged users in FLOSS development projects and Wikipedia, we also have to analyze the total number edits made by these users, as well as the total number of files and articles receiving these contributions in each project. Figure 4 and Figure 5 depict the total number of edits/changes and the total number of articles/files in every FLOSS project and Wikipedia language edition considered in our study. As we may have expected a priori, the English and German language editions of Wikipedia hold both the highest number of articles and the highest number of edits from privileged users. Regarding FLOSS projects, we have *mono* and *gcc* clearly standing out from the rest of cases, according to their number of files. If we turn to the number of edits from committers, we can identify four clear outliers: *mono*, *gcc*, *crystal* and *emacs*. According to this results, whereas in Wikipedia these two outliers are clearly separating themselves from the rest of their population, according to their total number of sysops, the four outliers in FLOSS projects also presents a specially high number of committers (more than 300 in each case) which indicates that those projects are included in the group of FLOSS initiatives that is also beginning to acquire a “critical mass” of contributors.

We must also notice the big difference between both types of initiatives in quantitative terms. The Wikipedia language editions have to maintain a larger number of articles and receives a larger number of contributions as well than FLOSS development projects. Tables 1 and 2 present summary data corresponding to the boxplots of Figures 4 and 5.

To conclude with the results of our descriptive quantitative analysis, in Figure 6 we present the histograms of the number of distinct committers per file in FLOSS projects and the number of distinct sysops per Wikipedia article. This is a measure of the “territoriality” level of contributors from both types of initiatives. A low number of distinct contributors per file/article indicates that committers and sysops tend to concentrate their effort in a low number of targets, while a high number of contributors per file/article indicates the opposite situation. As we can see, both types of projects
follow the former behavioral pattern, presenting a very low number of different contributors per file/article.

5. Discussion

We can infer several interesting conclusions from this quantitative analysis of privileged contributors in FLOSS projects and Wikipedia. In the first place, our results show that Wikipedia receive many more contributions from privileged users than FLOSS projects, even though only committers can access code repositories in FLOSS projects, and any user can access Wikipedia articles (not only privileged ones). This is an obvious consequence of the inherent organizational structure of both types of information systems. Wikipedia articles store knowledge, in the form of encyclopedic entries, while FLOSS development projects store (mainly) source code. As a consequence of that, the entry barrier to contribute to FLOSS projects is higher than in Wikipedia, since FLOSS committers need first to demonstrate their technical skills and abilities to contribute to a...
certain project, while Wikipedia authors do not need to fulfill any previous requirement to contribute to an article.

Nevertheless, these numbers provide an empirical characterization of each type of initiative. According to them, software development projects are not expected to receive a high number of contributions until they reach some popularity. On the contrary, distributed information systems focused on content creation, and open to receive contributions from any user, are expected to support a higher number of contributions. This is an important implication for system designers and administrators when they allocate resources for a new networked distributed collaborative project. But this is only an initial aspect of a set of more interesting conclusions confirmed by the rest of results.

The Gini coefficients also offer us valuable information about the inner contribution mechanisms of both types of collaborative initiatives. While FLOSS projects receive a higher number of contributions from a small subset of committers (a Gini coefficient of $g = 0.8$ tell us that 20% of the total number of users is committing 80% of the total number of changes in the project), article editions from Wikipedia sysops are much more balanced over the group of privileged users. Hence, these results suggest the presence of radically different organizational patterns. On one hand, the role of sysops in Wikipedia is very homogeneous, and most of them contribute with the same intensity to the project. On the other hand, committers with special roles, like those who maintain critical pieces of code, or those who are in charge of overall supervision tasks, may concentrate a higher number of contributions.

From an organizational point of view, this has direct implications for the successful management of open initiatives build around virtual communities. According to [20], both the skill level and skill heterogeneity positively affect the survival and performances of FLOSS development projects, though following a non-linear relationship. Moreover, these authors also give empirical proof that design modularity is also positively associated with the project performance (if we can estimate that projects are comparable regarding other relevant parameters). Our results show that activity patterns of privileged users in open content development projects (like Wikipedia) are more balanced than those of committers in FLOSS development projects. Thus the effort distribution is more egalitarian, an this approach allows Wikipedia sysops to surveil a broader number of articles and maintain an higher growing rate in their global level of contribution. The clear inequality level exhibited by committers in FLOSS development projects may eventually act as a bottleneck, preventing committers to attend critical tasks for the project due to an excessive workload.

Besides that, Figure 1 shows that the English and German Wikipedias are clearly separating themselves from the rest of language editions, regarding their number of sysops. The case of the English language edition is extremely unusual, since it is now quite far from any other language edition in Wikipedia, or any other FLOSS project. This has important implications with regards to the workload organization of these information repositories. Wikipedia seems to be able to scale better than FLOSS projects, rapidly attracting a higher number of users and “sysops” that can effectively contribute to the project. Nonetheless, many FLOSS projects that have a longer history than any language edition in Wikipedia cannot compete with the total number of edits received in the encyclopaedias. These results set the question about the capacity of FLOSS development projects to scale so rapidly and effectively as open content development projects like Wikipedia. Despite the low entry barrier in the Wikipedias, and the obvious advantage of attracting English writers from many countries around the world, this is something that should be further analyzed in future research works.

In contrast with these results, we find an almost identical territoriality pattern, when we measure the number of different committers per file in FLOSS projects and the number of distinct sysops in Wikipedia articles. Considering that contributions from sysops in Wikipedia exhibit a quite egalitarian level, and taking into account that commits from committers in FLOSS development projects are concentrated on a small subset of those privileged users, we can extract some important conclusions about the distinct organizational approaches followed by these types of initiatives. If a low number of committers are responsible for a higher number of changes, given that the number of contributions per artifact (article/file) is the same in both fields, we can infer that committers in FLOSS development projects must be responsible for a higher number of files. On the other side, Wikipedia sysops provide a more egalitarian distribution of their contributions, and given the same distribution of contributions per artifact, each of them will monitor a lower number of articles.

This is the reasoning behind the attraction of a higher number of privileged users. We need a higher number of sysops in each language edition of Wikipedia to manage the faster growing number of articles produced. The more sysops we are able to attract, the more egalitarian will be the distribution of effort among sysops, given a fixed number of privileged users per article. In FLOSS development projects, increasing code modularity (that is, splitting the code over a higher number of source code files) will lead to a more equal distribution of committers’ effort, though we would need to attract a higher number of committers with varied technical skills to take in charge of new modules as well. In this way, we can identify a vertical growing structure in FLOSS development projects, concentrating the effort of a small number of committers who have to monitor a higher number of files. On the other hand, open content de-
velopment projects like Wikipedia show a horizontal growing structure, focusing on broadening the number of artifacts produced by the project, and trying to ensure that the number of sysops controlling those artifacts is high enough to distribute their effort in a more egalitarian way.

All these results have direct impact in our understanding of how FLOSS development projects and Wikipedia (as an example of open content development projects) work. The main objective of the Wikipedia project is to broaden the number of encyclopaedic entries covered in each language edition. The inner behavioral patterns followed by the Wikipedias favors the achievement of this goal. On the other side, the focus of FLOSS development projects is to maintain the source code, ensuring the functionality and stability of the software, and progressively eliminating bugs. Since code changes are controlled by committers, they cannot be implemented at the same rate that Wikipedia changes, and they are focused in a small subset of files. Again, the question of whether FLOSS projects could scale maintaining their working philosophy, if the number of source code files increases. Some projects seem to be finding a good answer, raising the number of committers, and starting to stand out from other projects, in order to effectively respond to these evolution challenges.

Distributed software development teams and offshoring software development projects can benefit from these insights. Depending on their focus, the type of artifacts they want to create and the expect growing rate they would like to achieve for their communities of users, project managers and information systems designers will have to choose between a horizontal or vertical approach for their evolution over time. This decision has proven to be critical in terms of the direct impact for the future performance of project and the workload that privileged contributors within these projects may have to support.

6. Conclusions and Future Research

In this paper, we have presented a quantitative analysis to describe contribution patterns of privileged users both in FLOSS development projects and in Wikipedia, the most important example of libre content development projects nowadays. Focusing on privileged users, we have drawn some noteworthy conclusions which can be extrapolated to the organization and management of other collaborative initiatives, constructed around virtual communities on web information systems. In the first place, there are several FLOSS projects and Wikipedia language editions, specially the English and German Wikipedias, that are increasing their number of privileged users to effectively control the progress of content and source code. We have also concluded that the “territoriality” of contributions from privileged users in both types of projects is almost identical, proving that volunteer supervisors tend to concentrate their surveillance and management work on a small subset of files/articles, disregarding the nature of the content stored in the information system.

We also conclude that Wikipedia communities are able to scale more effectively to control changes in a larger number of articles, rapidly increasing the number of privileged users who survey article content, and who contribute to the project with a similar level of effort. In contrast with that results, we have found a much higher inequality level in contributions from FLOSS committers, which indicates that a small number of editors may become overloaded in due course, if the size of the project grows. Eventually, it would be interesting to further explore the evolution over time of the number of Wikipedia sysops and FLOSS committers, comparing both to have a clearer picture of how these virtual communities solve the problem of managing growing information system repositories. Likewise, the size of edits and the type of actions undertaken in each one (content additions, deletion, arrangement...) should be explored to characterize role specialization in these communities of privileged users, and analyze which of those tasks play a more critical role in the project evolution.

7 Acknowledgments

This work has been funded in part by the European Commission, under the FLOSSMETRICS project (FP6-IST-5-033547) and by the Spanish CICyT, under the SobreSalto project (TIN2007-66172).

References


[34] F. Rullani. Dragging developers towards the core. how the free/libre/open source software community enhances developers’ contribution. LEM Papers Series 2006/22, Laboratory of Economics and Management (LEM), Sant’Anna School of Advanced Studies, Pisa, Italy, Sep 2006.


