

Standard Grimoire Report
OPNFV Project
18-Q3



November 20, 2018

This report would not exist without the effort of the people involved in the development of the Grimoire toolset.

(cc) 2018 Bitergia. Some rights reserved.
This work licensed under Creative Commons Attribution-ShareAlike 4.0
Unported License.
To view a copy of full license, see
<http://creativecommons.org/licenses/by-sa/4.0>,
or write to Creative Commons, 559 Nathan Abbott Way, Stanford,
California 94305, USA.

Executive Summary

This report provides a quantitative analysis of the current and past situation of the OPNFV project. All the data presented in it is based on information retrieved from the software development repositories of the project. The analysis includes a summary of the general situation of the project, and specific analysis of some of its development processes and communication channels. Data from previous periods is also shown for comparison.

Contents

1 Project overview	5
2 Activity	6
3 Community	6
4 Process	8
A Metrics Definitions	9

1 Project overview

The report looks at activities across the OPNFV community during 18-Q3 (2018-07-01 to 2018-09-30), comparing it to previous period of analysis.

Data source	Activity last quarter	Change (wrt to prev. quarter)
git	3630 Commits	-36%
mailing lists	0 Sent Emails	-100%
gerrit	1843 Closed reviews	-23%
gerrit	1806 Submitted reviews	-26%
jira_issues	363 Closed tickets	-60%
jira_issues	508 Opened tickets	-34%

Table 1: Activity during the last period of analysis and its evolution

Table 1 shows development activity for each of the analyzed data sources. The activity column displays information about the net activity numbers, while the Change column displays information about the relative difference with respect to the previous period of analysis.

Gerrit data offers an interesting view on how software development process of OPNFV community is working from the point of view of efficiency and process. To analyze this, some metrics based on Gerrit changesets and reviews are presented below. In terms of Gerrit terminology, these metrics are calculated as follows:

- Review Efficiency Index (REI), measured as the number of closed changesets out of the new ones in a given period.
- Time to Merge (TTM), measured as the time since a review is submitted until this is closed.

REI	TTM
1.02	0.55 days

Table 2: Closed out of opened changesets (REI) and median time to merge (TTM)

The rest of the document is divided into three sections with information from the last periods:

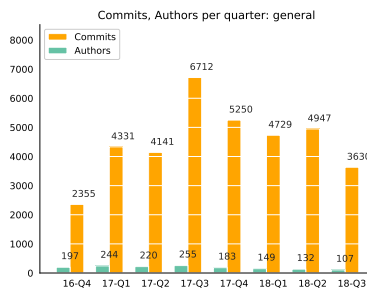
- Activity: focused on contributions.

- Community: focused on contributors.
- Process: focused on efficiency and timing.

2 Activity

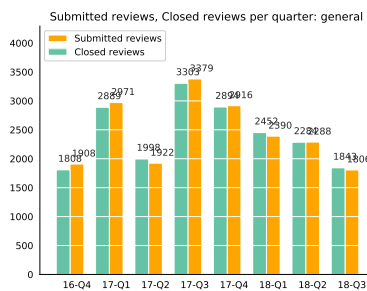
This section covers contributions in the different data sources.

The bar chart below shows the evolution of the number of commits and authors in Git through time, grouped by quarters.



Period	Commits	Authors
16-Q4	2355	197
17-Q1	4331	244
17-Q2	4141	220
17-Q3	6712	255
17-Q4	5250	183
18-Q1	4729	149
18-Q2	4947	132
18-Q3	3630	107

Following chart shows activity in Gerrit data source. It is based on comparing the number of opened and closed reviews grouped by quarters.

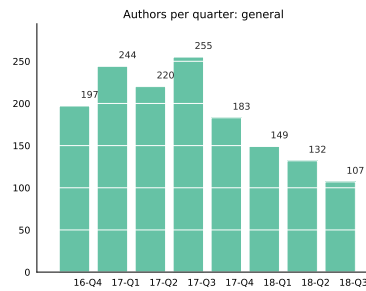


Period	Opened	Closed
16-Q4	1908	1808
17-Q1	2971	2889
17-Q2	1922	1998
17-Q3	3379	3303
17-Q4	2916	2894
18-Q1	2390	2452
18-Q2	2288	2284
18-Q3	1806	1843

3 Community

This section tries to help us to understand the evolution of OPNFV community by looking at active contributors and organizations in the last period of analysis, compared to previous ones.

Number of active authors in Git is shown below, giving us a quick look of contributors evolution in the last quarter compared to previous ones.



Period	Active Authors
16-Q4	197
17-Q1	244
17-Q2	220
17-Q3	255
17-Q4	183
18-Q1	149
18-Q2	132
18-Q3	107

In addition, table below offers a quick glance to the most active authors in Git in the whole period of time shown in the bar chart above.

Author	Commit (s)
Cedric Ollivier	698
Tim Rozet	401
Alexandru Avadanii	368
Trevor Bramwell	168
Fatih Degirmenci	145
Rodolfo Alonso Hernandez	145
Aric Gardner	125
Dan Xu	114
huangxiangyu	90
Michael Polenchuk	89
Yang (Gabriel) Yu	78
Georg Kunz	70
rexlee8776	61
Markos Chandras	59
m00133142	59
wenjuan dong	57
Manuel Buil Mur	53
Serena Feng	38
Jamo Luhrsen	37
Emma Foley	34
Mark Beierl	33

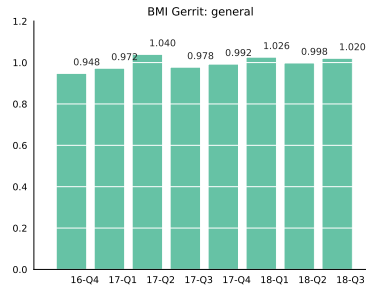
In a similar way, table below shows the same information groped by organization instead of author.

Organization	Commit (s)
Orange	711
Red Hat	474
Huawei	463
ENEA AB	429
Intel	333
Ericsson	295
Linux Foundation	293
Unknown	134
ZTE Corporation	120
Mirantis	89
SUSE	59
Nokia	57
Dell	35
ARM	27
AT&T	22
CableLabs	19
NEC	15
Kontron	14
Cisco	12
Freescale	10
Wipro	9

4 Process

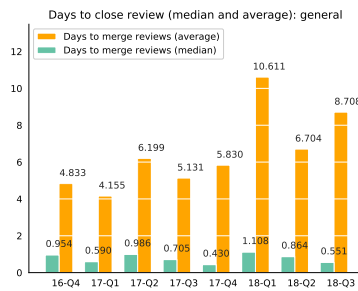
This section intends to show the evolution of efficiency and timing when dealing with tasks related with code review processes.

For Gerrit data source, we use REI, that was defined in section 1, to measure the efficiency in code review process. Chart below shows evolution of REI by quarters in order to visualize how changesets are being managed by the community.



Period	Closed/Subm.
16-Q4	0.95
17-Q1	0.97
17-Q2	1.04
17-Q3	0.98
17-Q4	0.99
18-Q1	1.03
18-Q2	1.00
18-Q3	1.02

In terms of time and again for Gerrit, chart below shows the evolution of mean and median times—in days—to close a review (TTM, defined in section 1).



Period	Median	Mean
16-Q4	0.95	4.83
17-Q1	0.59	4.16
17-Q2	0.99	6.20
17-Q3	0.71	5.13
17-Q4	0.43	5.83
18-Q1	1.11	10.61
18-Q2	0.86	6.70
18-Q3	0.55	8.71

A Metrics Definitions

- **Commit:** this is defined as the action(s) that performs a change in the source code. Bots, merges and other type of automatic activity is removed from the records. In addition, when aggregating several git repositories, this metric only counts unique revisions (unique hashes found in the git repositories). In addition, all branches are aggregated to the analysis.
- **Submitted changesets:** a code review is the process of peer reviewing source code changes. A submitted code is not merged to the master code of a given project till this is approved. A submitted code review is defined as any changeset submitted to the Gerrit system.
- **Authors:** a developer is defined as author if she is the owner of the patchset sent for reviewing and this is merged into the source code.

As previously indicated, automatic commits such bot's are removed from this analysis.

- Efficiency closing changesets: this metric is a derivation of the Backlog Management Index as it is named as Review efficiency index (REI). As similarly used in the BMI index, this metrics measures the number of closed changesets out of the total number of new changesets in a given period.
- Time to Merge: this time consists of the time between the first upload of the first changeset till the last iteration of the code review process is merged into the code. This metric is provided in number of days.
- Developer per period: average of developers per period ignoring bots and merges.
- Emails sent: number of emails sent by people to the several mailing lists. Bots are not registered.
- People sending emails: number of people sending those emails ignoring bots.