

## Communication when Submitting Patches ----Part I

1. What kind of contributor are you?

Option	Count	Ratio
Only write code	13	52.00%
Review code	1	4.00%
Write code and review code	11	44.00%

2. Start communication at right time. (Reserve sufficient review time for the patch (except fixing bug) before merge window closed.)

Question\Option	1	2	3	4	5	mean
Importance	2(7.69%)	1(3.85%)	4(15.38%)	10(38.46%)	9(34.62%)	3.88
Difficulty	8(30.77%)	4(15.38%)	6(23.08%)	6(23.08%)	2(7.69%)	2.62

3. Separate each logical change into a separate patch.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	0(0%)	7(26.92%)	19(73.08%)	4.73
Difficulty	3(11.54%)	9(34.62%)	5(19.23%)	7(26.92%)	2(7.69%)	2.85

4. When describing what you did, use imperative mood (e.g. don't use 'This patch ...').

Question\Option	1	2	3	4	5	mean
Importance	4(15.38%)	5(19.23%)	6(23.08%)	7(26.92%)	4(15.38%)	3.08
Difficulty	9(34.62%)	10(38.46%)	6(23.08%)	1(3.85%)	0(0%)	1.96

5. Pay attention to the rules in the community, such as the formatting requirement.

Question\Option	1	2	3	4	5	mean
-----------------	---	---	---	---	---	------

Importance	1(3.85%)	0(0%)	5(19.23%)	6(23.08%)	14(53.85%)	4.23
Difficulty	6(23.08%)	8(30.77%)	8(30.77%)	0(0%)	4(15.38%)	2.54

6. Try your best to describe your motivation, including why you did it and what you did.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	4(15.38%)	10(38.46%)	12(46.15%)	4.31
Difficulty	5(19.23%)	4(15.38%)	11(42.31%)	5(19.23%)	1(3.85%)	2.73

7. If the implementation is complex and it requires reviewer to spend a lot of time to understand it, briefly describe the implementation details.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	4(15.38%)	7(26.92%)	15(57.69%)	4.42
Difficulty	4(15.38%)	4(15.38%)	8(30.77%)	8(30.77%)	2(7.69%)	3

8. If there is an improvement after applying the patch, clearly describe the improvement.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	0(0%)	12(46.15%)	14(53.85%)	4.54
Difficulty	7(26.92%)	7(26.92%)	6(23.08%)	5(19.23%)	1(3.85%)	2.46

9. If there is a cost after applying the patch, provide the trade-off.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	5(19.23%)	6(23.08%)	15(57.69%)	4.38
Difficulty	4(15.38%)	3(11.54%)	9(34.62%)	8(30.77%)	2(7.69%)	3.04

10. If the patch refers other information, e.g., commits, email content, the view of the expert, or links of related information, include it.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	2(7.69%)	7(26.92%)	6(23.08%)	11(42.31%)	4
Difficulty	4(15.38%)	12(46.15%)	8(30.77%)	2(7.69%)	0(0%)	2.31

11. If the patch is a prototype or on-going, include the limitation of the current version.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	3(11.54%)	10(38.46%)	13(50%)	4.38
Difficulty	6(23.08%)	10(38.46%)	8(30.77%)	0(0%)	2(7.69%)	2.31

12. If the patch has been tested or reviewed by others, use tags such as ‘Reviewed-by’, ‘Tested-by’.

Question\Option	1	2	3	4	5	mean
Importance	1(3.85%)	1(3.85%)	6(23.08%)	2(7.69%)	16(61.54%)	4.19
Difficulty	14(53.85%)	7(26.92%)	4(15.38%)	1(3.85%)	0(0%)	1.69

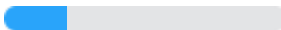
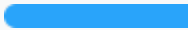

13. If there is more than one way to implement it, showing the superiority of your solution is much better.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	4(15.38%)	7(26.92%)	8(30.77%)	7(26.92%)	3.69
Difficulty	2(7.69%)	5(19.23%)	9(34.62%)	5(19.23%)	5(19.23%)	3.23

14. If the feature is relative complex or it is very important but has not been recognized by the community, provide sufficient reasons why this new feature is required.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	7(26.92%)	3(11.54%)	16(61.54%)	4.35
Difficulty	2(7.69%)	0(0%)	10(38.46%)	7(26.92%)	7(26.92%)	3.65

15. If the feature has been discussed previously, or the feature is a widely recognized feature, you may omit providing the reason. What do you think of this practice?

Option	Count	Ratio
It's significant, because it can save developers' effort.	5	 20.00%
It's not significant.	15	 60.00%
It's meaningless.	5	 20.00%

16. If the new feature includes several functions, list all of the functions or give a brief introduction.

Question\Option	1	2	3	4	5	mean
Importance	2(7.69%)	3(11.54%)	6(23.08%)	11(42.31%)	4(15.38%)	3.46
Difficulty	7(26.92%)	7(26.92%)	10(38.46%)	1(3.85%)	1(3.85%)	2.31

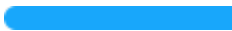


17. If the patch is bug fix, describe the scenario of bug or how to reproduce it, or the issue report.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	2(7.69%)	1(3.85%)	6(23.08%)	17(65.38%)	4.46
Difficulty	3(11.54%)	10(38.46%)	9(34.62%)	3(11.54%)	1(3.85%)	2.58

18. Especially if the bug has serious consequence, e.g. memory leak, system crash, including the seriousness of bug can bring more attention from reviewer.

Question\Option	1	2	3	4	5	mean
Importance	0(0%)	0(0%)	4(15.38%)	6(23.08%)	16(61.54%)	4.46
Difficulty	3(11.54%)	10(38.46%)	10(38.46%)	0(0%)	3(11.54%)	2.62

19. For the patches such as: easy case, fix bug, simplify code, add driver, because the reason for submitting these types of patch is very clear, just express what you did which has already reflected the reasons. What do you think of this practice?

Option	Count	Ratio
It's significant, because it can save developers' effort.	22	 88.00%
It's not significant.	3	 12.00%
It's meaningless.	0	 0%

20. Are there any other practices that you consider important but we missed? If there are, please provide.

No answer.

## Communication when Submitting Patches ----Part II

2. What do you think of the importance of communication in the open source community?

Option	Count	Ratio
Not important	0	<div></div> 0%
Little important	1	<div></div> 4.00%
Important	8	<div></div> 32.00%
Very important	17	<div></div> 68.00%

3. Have you ever encountered communication problems?

Option	Count	Ratio
Yes, I have	20	<div></div> 80.00%
No, I haven't	5	<div></div> 20.00%


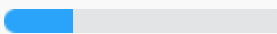
4. How often do you experience communication problems?

Option	Count	Ratio
Often	8	<div></div> 32.00%
Not often	17	<div></div> 68.00%


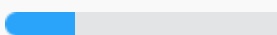
5. What do you think of the importance of communication during patch review via email?

Option	Count	Ratio
Not important	0	<div></div> 0%
Little important	2	<div></div> 8.00%
Important	15	<div></div> 60.00%
Very important	8	<div></div> 32.00%

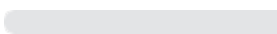
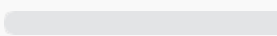


6. Have you ever encountered that your patch was rejected because of description?

Option	Count	Ratio
Yes, I have	15	 62.50%
No, I haven't	9	 37.50%


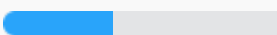
7. Have you ever rejected the patch because of its bad description during code review?

Option	Count	Ratio
Yes, I have	9	 75%
No, I haven't	3	 25%

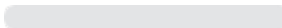
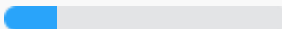

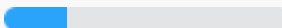
8. Do you think the description of the patch is very important?

Option	Count	Ratio
Not important	0	 0%
Little important	0	 0%
Important	14	 56.00%
Very important	11	 44.00%

9. When you submit your patch via email, do you feel it is difficult to write a good description?

Option	Count	Ratio
Yes, I do	13	 52.00%
No, I don't	12	 48.00%

10. If there is guidance to tell developers in different circumstances how to write the description of the patch, do you think it is necessary?

Option	Count	Ratio
Not necessary	0	 0%
Little necessary	3	 12.00%
Necessary	13	 52.00%
Very necessary	6	 24.00%

9. What problems of the patch description make you reject the patch?

ID	ANSWER
2	Various reasons. A good patch description describes: - What is the problem, and why this is an issue, - How it is solved (unless that's obvious from the code), - Why it is solved this way.
4	Bad or missing explanation of the problem.
5	Coding style problems, misunderstanding of the high level design
6	- no description - a description that didn't accurately describe the problem.
8	Incomplete description. Or just too little justification for the patch.
9	Description simply described the implementation, which the diff itself does. Descriptions should address everything else: motivation for the patch, discussion of the design, reasoning for not taking alternative approaches that reviewers might consider
10	If people do not describe the problem clearly, eg. it is fixing a problem, but he/she describe another problem maybe related but it is not the key aspect.
11	As above. Also for code failing to scale, or failing to handle errors well
12	Incomplete or incorrect information, or garbled language. Poor attention to detail in the description makes me wonder whether about the attention to detail in the patch itself.
13	Not understanding what is the goal.
14	Various reasons. A good patch description describes: - What is the problem, and why this is an issue, - How it is solved (unless that's obvious from the code), - Why it is solved this way.
15	Bad or missing explanation of the problem.
16	Coding style problems, misunderstanding of the high level design
17	- no description - a description that didn't accurately describe the problem.
18	Incomplete description. Or just too little justification for the patch.
19	Description simply described the implementation, which the diff itself does. Descriptions should address everything else: motivation for the patch, discussion of the design, reasoning for not taking alternative approaches that reviewers might consider
20	If people do not describe the problem clearly, eg. it is fixing a problem, but he/she describe another problem maybe related but it is not the key aspect.
21	As above. Also for code failing to scale, or failing to handle errors well



<b>24</b>	Incomplete or incorrect information, or garbled language. Poor attention to detail in the description makes me wonder whether about the attention to detail in the patch itself.
<b>25</b>	Not understanding what is the goal.