

Defect Handling in Medium and Large Open Source Projects

A. Güneş Koru and Jeff Tian

Summary

- _ Goal: As a start point for better understand the defect characteristics and improve quality for a wider variety of software products??
- _ Defect includes failures, faults, changes, new requirements, new functionalities, ideas, and tasks.
- _ Code is the main subject of defect records, followed by design documents, requirement documents, and user documentation.
- _ Developers and testers record both pre-release and post-release defects.
- _ Developers and testers record defects fairly consistently and keep fairly complete defect records

Discussion

- _ Your experience about defect handling in any FLOSS or commercial project
- _ What is the real challenge in handling defects? Why?
 - Record, track, fix, or ??
- _ Do you agree with the conclusion or method used?
- _ What can we learn from this paper if we want to do similar research?
- _ Is FLOSS cheaper ?? more efficient ??

The Survey Criterion

- _ Criterion for selecting project
 - Generally recognized as a success ??
 - Publicly accessible defect database & defect-tracking tool
 - Prior knowledge ??
- _ Target population
 - Development, management, documentation, testing, defect fixing
 - No general users ??

The Survey Questions, Tool, Method

Questions:

- Section A: Directly related to main characteristics of defect handling approach.
- Section B: About general product and project characteristics.
- Section C: Related to defect categorization for quality improvement purposes.

Tool: **Survey** (<http://opensource.isc.vt.edu/products/survey>)

Method:

- Subscribed related mailing list
- Send email and weekly reminders

Result

119 individual; 49.3% of 75 projects (high or low typical ??). 15 more projects

The number and percentages of the respondents assuming each role

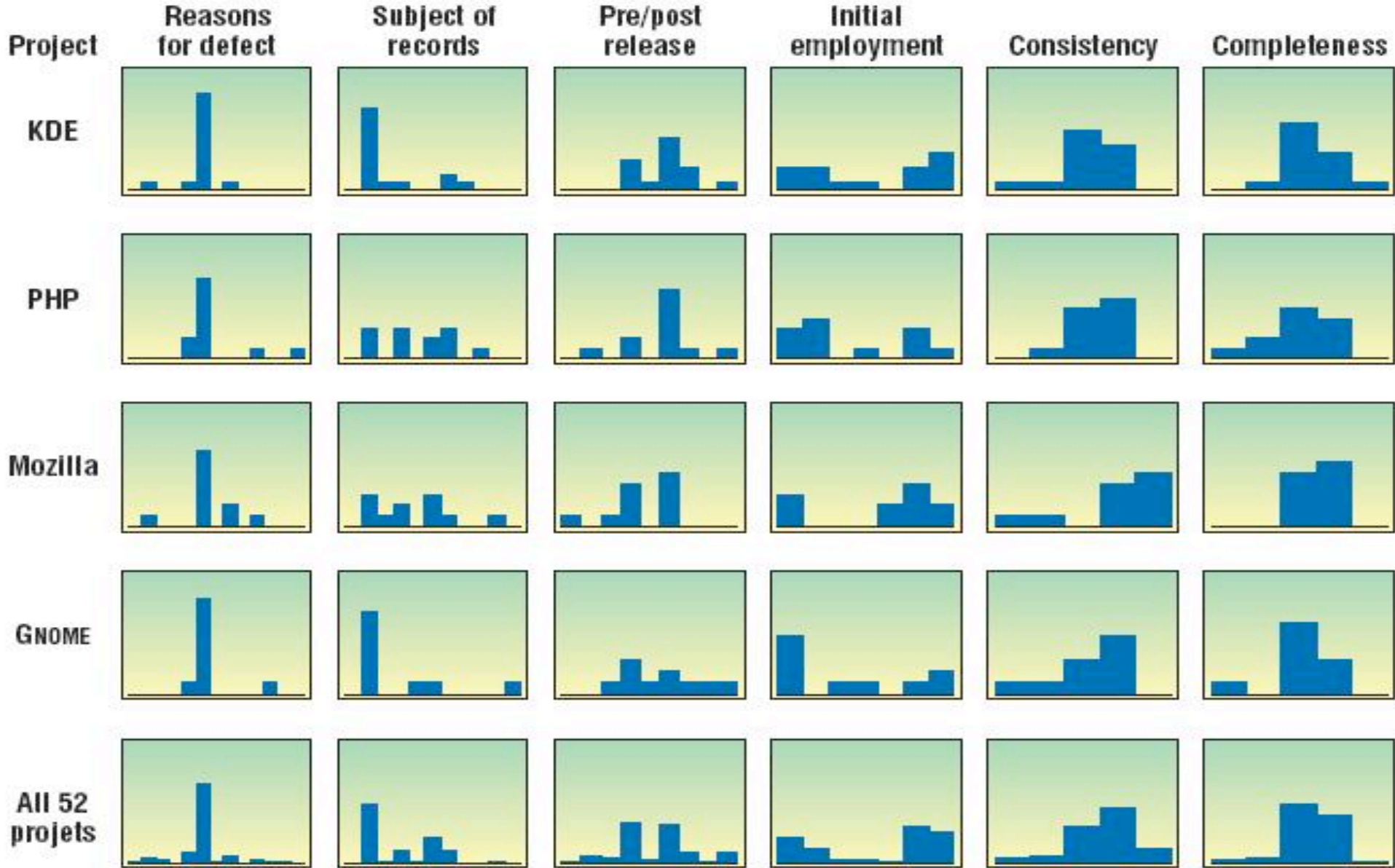
Respondent role	Number	Percentage
Defect database owner	16	13.45
Project manager	25	21.01
Developer	86	72.27
Tester	69	57.98
Defect fixer	66	55.46
Documentation	30	25.21
Other	4	3.36

Result

General results for Questions 4 through 9

Choice order	Question 4 Reasons for defect reports (choose all that apply)	Question 5 Subjects of defect records (choose all that apply)	Question 6 Pre/post release (choose all that apply)	Question 7 Initial employment (choose one)	Question 8 Consistency (choose one)	Question 9 Completeness (choose one)
1	Any change 100 (84.03%)	Source code 116 (97.48%)	Prerelease 31 (26.05%)	Inherited 31 (26.05%)	Very 14 (11.77%)	Always 2 (1.68%)
2	Problem/failure 116 (97.48%)	Design documents 44 (36.97%)	Post-release 36 (30.25%)	Project start 37 (31.09%)	Almost 55 (46.22%)	Almost always 48 (40.34%)
3	Incorrect work 108 (90.76%)	Requirements documents 32 (26.89%)	Both & no distinction 55 (46.22%)	After design 2 (1.68%)	Not very 37 (31.09%)	Sometimes 59 (49.58%)
4	False positives 6 (5.04%)	User documents 16 (13.45%)	Both & distinguished 61 (51.26%)	After coding 4 (3.36%)	Not 8 (6.72%)	Rarely 6 (5.04%)
5	Feature enhancement 11 (9.24%)	Other 1 (0.84%)	Other —	Testing 4 (3.36%)	Other —	Other —
6	Tasks 2 (1.68%)	—	—	Post-release 15 (12.61%)	—	—
7	Other 1 (0.84%)	—	—	Other —	—	—
No answer	—	1 (0.84%)	—	26 (21.85%)	5 (4.20%)	4 (3.36%)

Individual Project Comparisons



Lesson Learned

- _ Multiple choice question + other choices
 - Easy and consistent to answer
 - Easy to statically analyze
 - Allow unexpected answers
- _ Table and Figure
 - Table is clear
 - Figure less detail
- _ Need more detailed & deep information about the result analysis.
- _ Survey =? Quantitative

Is FLOSS cheaper, more efficient ??

- _ Real effort didn't count.
- _ Effort overlap.
- _ Products are duplicated. Waste?
- _ It is good for human society from long period.
- _ Merge with commercial management??
- _ ??