“Predicting Developers’ *Negative Feelings* about Code Review”

Prof. Filippo Lanubile

Student:
Chandrika Patibandla.

About

- When developing a *code review* developer examine each other's code to make sure.
  - Their code has *good quality*
  - To *share knowledge*
  - Also, to check the *standards of the coding.*
- In this process they may have *negative feelings* with their peers which can lead to frustration or stress.
- This negative interaction may also affect the *overall result of the project*.
- This study mainly says about those *negative experiences* which is called as ***pushback***.

Introduction

- Actually, code review provides many benefits for S/W organization:
  1. Improve *S/W maintenance*.
  2. Finding *defects*
  3. *Knowledge sharing*.

- This paper focuses on practical measurement of *feelings of pushback*, for that following research questions have been asked....

  RQ1: How frequent are negative experiences with code review?

  RQ2: What factors are associated with pushback occurring?

  RQ3: What metrics detect author-perceived pushback?

Definition

"The perception of unnecessary interpersonal conflict in code review while a reviewer is blocking a change request." - ***Pushback***

Method

Qualitative and quantitative methods using surveys and log data

They have developed 3 log-based metrics

- *14 developers*
- Metrics through a survey with a sample of *"2500" Developers* – covered five feelings of pushback;
- *1317 developers* have completed the survey.

A) Code Review at Google

At Google, code review is *mandatory*.

B) Interviews about Code Review Pushback

14 people have been conducted with an interview about code review pushback covering defining pushback and how they dealt with pushback it when it occurred.

Dealing with Pushback

- Initial reaction.
- Appeal to authority or experts.
- Long-term consequences.
- Investing time to work towards a resolution.

C) Design of Metrics

I) Rounds of a review:

Round of review would be *batch of comments* that an author and reviewer would send back and forth

II) Active reviewing time:

which is the *time that reviewer spends* in the review tool itself looking at the code and providing comments.

III) Active shepherding time:

This is the time that the *author spends* in the code review tool looking at and responding to comments as well as the time that they spent revising the code based on the comments that they receive.

D) Survey Design

They have *surveyed 1317 google* developers to test if their metrics predicated feelings of pushback

The survey has covered:

Survey Overview

- Overall code review perceptions
- Rating of two selected change requests.
- Asking for problematic code reviews

Feelings of Pushback

- Interpersonal conflict,
- Feeling that acceptance was withheld for too long.
- Reviewer asked for excessive changes,
- Feeling negatively about future code reviews
- Frustration

Selecting the Code Reviews for the Survey

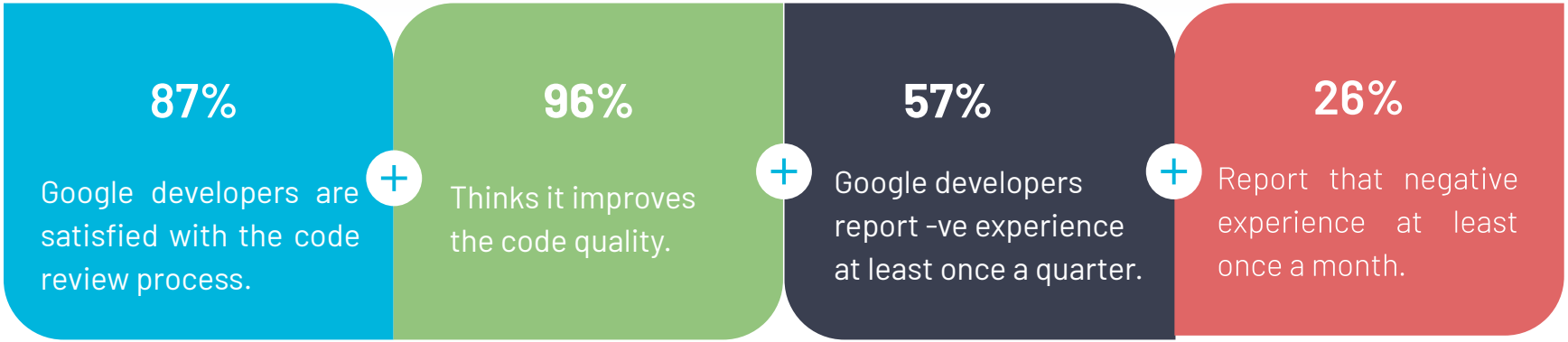
- To evaluate metrics its been compared the CR's Flagged by *one or more of their metrics* to developer feelings of pushback.
- Individual CR's can *fall into any one of the eight* categories with respect to the metrics or metric combinations as shown in the fig beside.

		CR Criteria for Inclusion		
	Surveyed CRs	Flag: Review Time	Flag: Shepherd Time	Flag: Rounds of Review
0 Flags	250			
1 Flag	200	✓		
	200		✓	
	200			✓
2 Flags	100	✓	✓	
	100	✓		✓
	100		✓	✓
3 Flags	100	✓	✓	✓

RESULTS

For the questions:

1) *How frequent are negative experiences with code review?*



2) *What factors are associated with pushback occurring?*

To investigate in which pushback more likely occurs, it is considered both quantitative and qualitative data.

Initially we start looking into code review attributes and later on we look at how developers answer the questions.

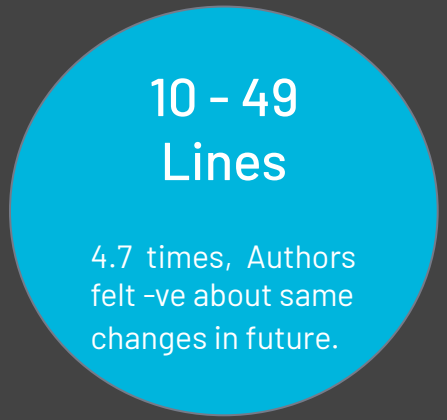
Code Review Attributes:

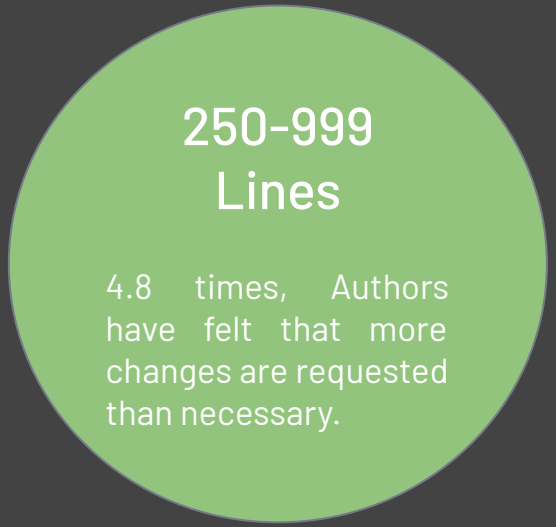
- *Readability*

  It is suggested that achieving readability is very much frustrating based on survey of Google developers. Also review process takes longer.

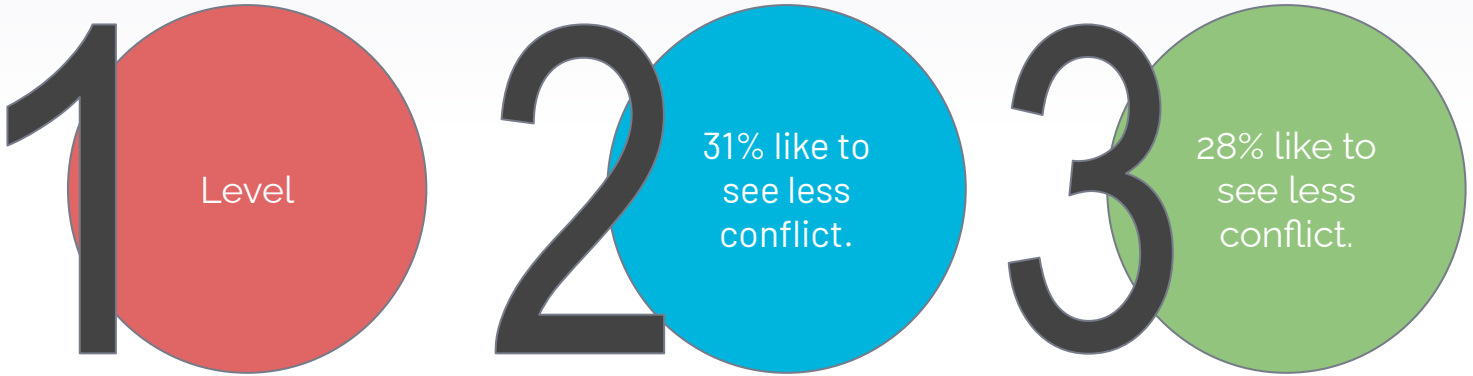
  More frustration - "round trip"

Code Review Size





New employee status & author seniority

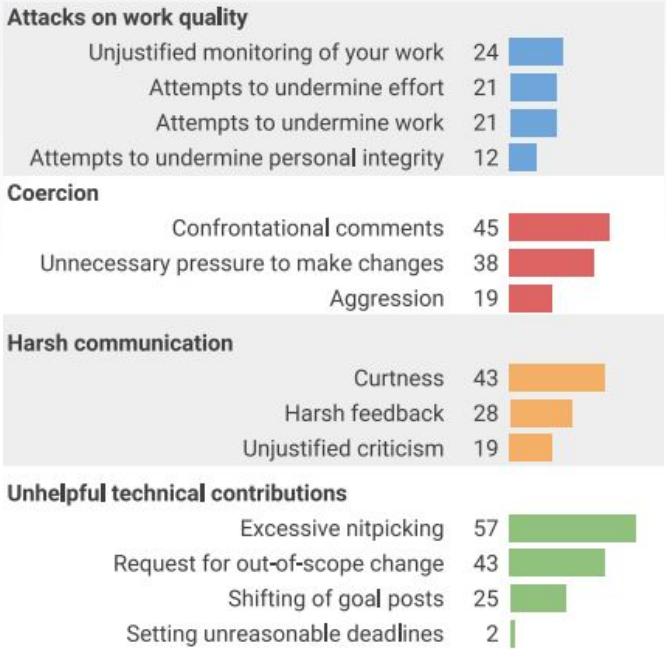


Selected code reviews: *(Emergent themes on why pushback occurs)*

- Why this behavior occurred during CRs.
- Frequencies of codes within open-ended comments for unflagged and flagged change requests shown in the fig beside.

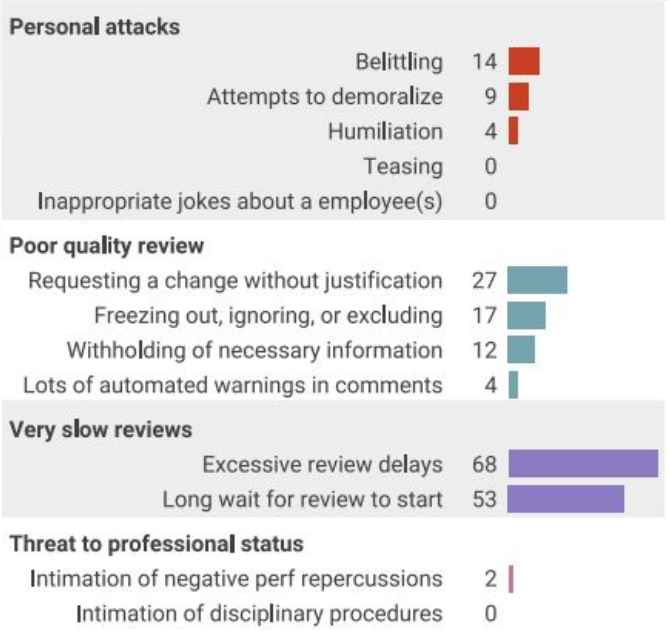
Topic	Total Comments	Unflagged Review Comments	Flagged Review Comments
Time zone & delays	26	1	25
Offline Conversation	19	1	18
Readability	15	0	15
Code review was complex or large	14	0	14
Conflict	12	0	12
Code quality & formatting	12	1	11
Documentation & comment clarity	7	1	6
Seniority, tenure, politics	5	0	5

Volunteered Code Reviews: *(Problematic Behaviors and Emergent Themes on why pushback occurs)*



"Frequency of behaviors indicated by developers in CRs they volunteered in the final section of the survey.

You can observe tables besides."



3) *What metrics detect author-perceived pushback?*

A) Occurrence of undesirable behaviors:

"Incidence rates for each feeling of. pushback as rated by authors of surveyed CRs; comparison of CRs not flagged by any metrics and flagged on all 3 metrics"

Feeling of Pushback	0 Flags	3 Flags
frustration	1%	27%
interpersonal conflict	0%	22%
acceptance was withheld for too long	4%	27%
reviewer asked for excessive changes	1%	16%
feeling negatively about future CRs	4%	8%
any pushback behavior	7%	49%
2+ types of pushback behavior	1%	27%

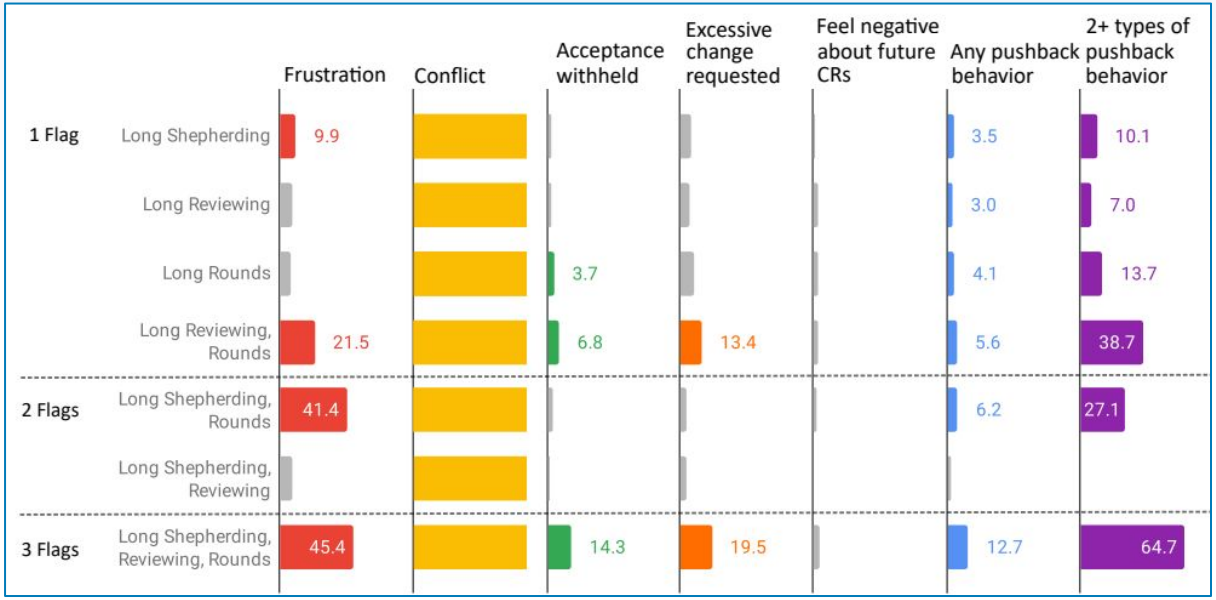
B) Precision & Recall of the Metrics:

"Precision and recall by feeling of pushback"

Pushback Feeling	Precision	Recall
frustration	0.11	0.98
interpersonal conflict	0.10	1.00
acceptance was withheld for too long	0.10	0.93
reviewer asked for excessive changes	0.07	0.97
feeling negatively about future CRs	0.06	0.87

C) Robustness of metrics to confounding factors:

Regression results predicting the likelihood of CR having undesirable, pushback behaviors. Numbers shown are the odds ratio from the fitted logistic regression model where the baseline category is an unflagged CR.



Conclusion

- Pushback is *rare*, but i does occur in code review at google.
- Pushback have *high recall* but *low precisions*.
- *Predictions* are needed to support future changes in design, to help *reduce pushback*.
- *Detecting pushback* also may help in identifying where *pushback is occured* in general /frequent.

THANKS!