Can Peer-Code Reviews be Exploited for Later Information Needs?

Andrew Sutherland [U. Saskatchewan] and **Gina Venolia** [Microsoft Research] – HiP Human Interactions in Programming group

Code reviews are a focal point where knowledge about code is made explicit

- This knowledge can be used to better inform design decisions and changes
- Tool support may structure reviews and retain the data for later reuse

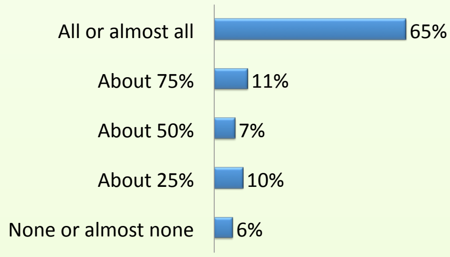
Field Study of Current Practice

Methodology

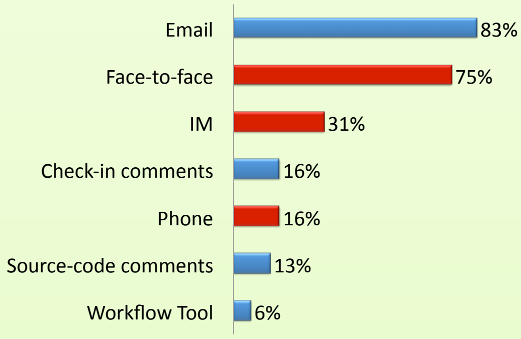
- Formal survey (157 SDEs, 134 SDETs)
- Interviews
- Email thread analysis
- Prototyping and feedback

Observations

- Almost all checked-in code is reviewed



- Equal reliance on async and sync modes
  - Diff packs and email
  - Over the shoulder
  - Could benefit from workflow tool

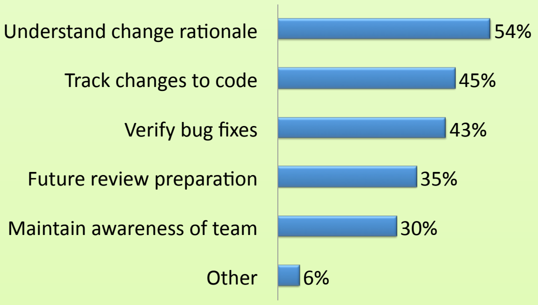


- But it’s not systematically retained

  31% of async reviews are retained for reviewers or team

  17% of sync reviews are retained for reviewers or team
- Desire to use knowledge captured during review later

  63% would be “likely” to use a tool for this
- Many reasons to refer later to reviews

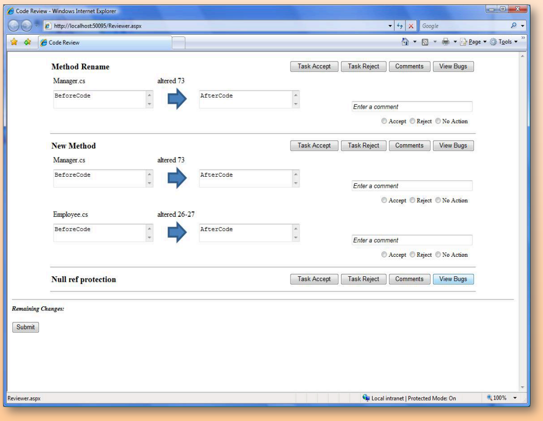


Guidelines for Tool Support

- Tools should support the existing author-reviewer ***workflow***
- Views of the code both prior to and after the change should be available
- Difference lists should be organized logically – by ***tasks*** or groups of changes
- Reviewers and authors need the ability to include additional information – such as comments or notes
- Review data needs to be ***retained*** in a systematic way
- Past review data needs to be ***recoverable*** so that authors and reviewers can utilize this information later

Crosscheck Tool: Support & capture reviews

- *Author* identifies *tasks*, tags diff-blocks, and assigns reviewers
- *Reviewers* view diffs by task; comments on, accepts or rejects diff-blocks or whole tasks
- *Author* makes changes, answers comments; the review cycle continues
- Async or over-the-shoulder
- Knowledge is retained in a database (search, browse and link)


Code Review - Windows Internet Explorer
http://localhost:50095/Reviewer.aspx
Google
Code Review
Page
Tools
Task Accept
Task Reject
Comments
View Bugs
Submit
Reviewer.aspx
Local intranet | Protected Mode: On
100%

Method Rename

Manager.cs altered 73

BeforeCode → AfterCode

Enter a comment

○ Accept ○ Reject ○ No Action

New Method

Manager.cs altered 73

BeforeCode → AfterCode

Enter a comment

○ Accept ○ Reject ○ No Action

Employee.cs altered 26-27

BeforeCode → AfterCode

Enter a comment

○ Accept ○ Reject ○ No Action

Null ref protection

Remaining Changes:

Future Work

- Querying and visualization of review data
- Automation for initial task identification
- Structural diff
- Integration into modern development environments
- Better support for synchronous code reviews with video and audio recording