The Impact of Bug Management Patterns on Bug Fixing: A Case Study of Eclipse Projects

Masao Ohira  
Wakayama Univ.

Ahmed E. Hassan  
Queens Univ.

Naoya Osawa  
Nara Institute of Science and Technology

Ken-ichi Matsumoto
Background

• An efficient bug management process (reports, assignment and fixing) is critical for the success of software projects.

• As the user base grows, some large open source projects receive a large number of bug reports.
Complex challenges to the bug management process

- Bug management process

- Understand a large amount of new bug reports
- Figure out if they are real bugs and whether they were reported in the past (i.e., duplicate bugs)
- Assign them to appropriate persons to fix the bugs quickly
  - 44% of bugs in the Eclipse project are reassigned to more than one developer [Jeong et al. 2009]
Related work

- Quality of bug reports
  - A good report helps developers to quickly find, replicate, and understand the bugs.
  - [N. Bettenburg et al. 2008] [S. Breu et al. 2010] [T. Zimmermann et al. 2010]

- Detection of duplicate bug reports
  - Users often report the same problems which were reported and fixed in the past.
  - [X. Wang et al. 2008] [N. Bettenburg et al. 2008] [C. Sun et al. 2010]

- Re-opening and reassigning of bug reports
  - A bug sometimes is be reopen and reassigned when it was assigned to an inappropriate developer.
  - [Anvik et al. 2007] [G. Jeong et al. 2009] [E. Shihab et al. 2010] [P. J. Guo et al. 2011]
Our focus:
Relations between the individuals

• Relations between individuals involved in the bug management process
  – Who reports? → Who triages it? → Who fixes it?
Our goal

- To better understand the impact of the relations between the individuals on the efficiency of the bug management process
  - Impact of the time to assign bug fixing tasks
  - Impact on the time to fix bugs

Bug reported

UNCONFIRMED

NEW

ASSIGNED

FIXED

RESOLVED

Time to assign a bug ($T_{assignment}$)

Time to fix a bug ($T_{bug-fix}$)
Pilot study (1)
Eclipse Platform and JDT

• RQ1: Does the time to assign a bug fixing task depend on the fact that the same developer reports a bug and triages it?

When a triager reports a bug, the average time of assignments is about 17–47% faster (2.6 days in Platform and 9.4 days in JDT)

• Result

<table>
<thead>
<tr>
<th>project</th>
<th>Reporter = Triager?</th>
<th># of reports</th>
<th>ratio</th>
<th>average days</th>
<th>median days</th>
<th>SD</th>
<th>max days</th>
<th>min days</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform</td>
<td>yes</td>
<td>1,000</td>
<td>24.2%</td>
<td>12.6</td>
<td>0.0</td>
<td>66.8</td>
<td>812.1</td>
<td>0.0</td>
<td>&lt; 0.01 **</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>3,133</td>
<td>75.8%</td>
<td>15.2</td>
<td>0.5</td>
<td>68.9</td>
<td>842.9</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>JDT</td>
<td>yes</td>
<td>452</td>
<td>27.3%</td>
<td>10.6</td>
<td>0.0</td>
<td>57.8</td>
<td>713.7</td>
<td>0.0</td>
<td>&lt; 0.01 **</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>1,205</td>
<td>72.7%</td>
<td>20.0</td>
<td>0.5</td>
<td>79.6</td>
<td>927.0</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>
Pilot study (2)
Eclipse Platform and JDT

- **RQ2**: *Does the time to fix a bug depend on the fact that the same developer triages a bug and fixes it?*

The average time of bug fixing by triagers was about two times faster (23.8 days in Platform and 10.2 days in JDT)

- **Result**

<table>
<thead>
<tr>
<th>project</th>
<th>Triager = Fixer?</th>
<th># of reports</th>
<th>ratio</th>
<th>average days</th>
<th>median days</th>
<th>SD</th>
<th>max days</th>
<th>min days</th>
<th>P-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Platform</td>
<td>yes</td>
<td>2,294</td>
<td>55.5%</td>
<td>23.1</td>
<td>1.2</td>
<td>65.3</td>
<td>776.0</td>
<td>0.0</td>
<td>&lt; 0.01 **</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>1,839</td>
<td>44.5%</td>
<td>46.9</td>
<td>5.9</td>
<td>111.1</td>
<td>988.2</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>JDT</td>
<td>yes</td>
<td>817</td>
<td>49.3%</td>
<td>12.6</td>
<td>0.8</td>
<td>42.9</td>
<td>583.1</td>
<td>0.0</td>
<td>&lt; 0.01 **</td>
</tr>
<tr>
<td></td>
<td>no</td>
<td>840</td>
<td>50.7%</td>
<td>22.8</td>
<td>1.3</td>
<td>62.8</td>
<td>705.9</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>
Bug Management Patterns

**Reporter=Triager=Fixer (R=T=F)**
One contributor plays all of the roles.

- **Reporter=Triager=Fixer (R=T=F)**
  - Reporter: A
  - Triager: A
  - Fixer: A

- **RESOLVED**

**Reporter ≠ Triager=Fixer (R≠T=F)**
One contributor serves as triager and fixer.

- **Reporter ≠ Triager=Fixer (R≠T=F)**
  - Reporter: A
  - Triager: A
  - Fixer: B

- **RESOLVED**

**Reporter=Triager ≠ Fixer (R=T≠F)**
One contributor asks another contributor to fix a bug.

- **Reporter=Triager ≠ Fixer (R=T≠F)**
  - Reporter: A
  - Triager: A
  - Fixer: B

- **RESOLVED**

**Reporter ≠ Triager ≠ Fixer (R≠T≠F)**
Each contributor has a different role from others.

- **Reporter ≠ Triager ≠ Fixer (R≠T≠F)**
  - Reporter: A
  - Triager: B
  - Fixer: C

- **RESOLVED**

2012/09/26 The Impact of Bug Management Patterns @ ICSM2012
Pattern #1

- Pattern #1 is assumed to make bug fixing faster.
  - He likely knows the bug source.
  - He likely has good confidence in his ability.
Pattern #2

- From our pilot study (RQ1), Pattern #2 can be assumed to make the bug assignment faster, but bug fix may be slower.
Pattern #3

- Reporter=Triager=Fixer (R=T=F)
  - One contributor plays all of the roles.

- Reporter\neq Triager=Fixer (R\neq T=F)
  - One contributor serves as triager and fixer.

- Reporter=Triager\neq Fixer (R=T\neq F)
  - Each contributor has a different role from others.

- Reporter\neq Triager\neq Fixer (R\neq T\neq F)

- Pattern #3 would make bug fixing itself faster if (B) has a good understanding of the bug reported by (A).
- otherwise it would make bug fixing difficult, because (B) has to spend the time to investigate the bug.
Pattern #4

• This pattern is assumed to make both the bug assignment and bug fixing most difficult
• The mismatches of knowledge and skills between them would be larger than the other patterns.
Case study on bug management patterns
Eclipse Platform and JDT

- **Data sets** (also used in the pilot study)
  - fixed bug reports from 2007 to 2009 (*)
    - Eclipse Platform: 4,133 reports
      - 811 reporters, 54 triagers and 85 fixers
    - Eclipse JDT: 1,657 reports
      - 369 reporters, 23 triagers and 33 fixers
  - The ratio of #3 and #4 are almost same and large (i.e., important patterns for the bug management process)

<table>
<thead>
<tr>
<th>project</th>
<th>pattern</th>
<th>ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R=T=F</td>
<td>17% (719/4,133)</td>
</tr>
<tr>
<td></td>
<td>R=T≠F</td>
<td>7% (281/4,133)</td>
</tr>
<tr>
<td>Platform</td>
<td>R≠T=F</td>
<td>38% (1,575/4,133)</td>
</tr>
<tr>
<td></td>
<td>R≠T≠F</td>
<td>38% (1,558/4,133)</td>
</tr>
<tr>
<td></td>
<td>R=T=F</td>
<td>14% (241/1,657)</td>
</tr>
<tr>
<td></td>
<td>R=T≠F</td>
<td>13% (211/1,657)</td>
</tr>
<tr>
<td>JDT</td>
<td>R≠T=F</td>
<td>35% (576/1,657)</td>
</tr>
<tr>
<td></td>
<td>R≠T≠F</td>
<td>38% (629/1,657)</td>
</tr>
</tbody>
</table>
Case study (1)
Bug management patterns in Eclipse Platform and JDT

- RQ3: How do the bug management patterns impact the time to complete bug assignments?

**One contributor plays all of the roles.**

**One contributor serves as triager and fixer.**

**One contributor asks another contributor to fix a bug.**

**Each contributor has a different role from others.**
• Result

![Box plots showing time assignment for different roles and platforms.](image)

- **Reporter=Triager=Fixer (R=T=F)**
  - One contributor plays all of the roles.

- **Reporter ≠ Triager=Fixer (R≠T=F)**
  - One contributor serves as triager and fixer.

- **Reporter=Triager ≠ Fixer (R=T≠F)**
  - One contributor asks another contributor to fix a bug.

- **Reporter ≠ Triager ≠ Fixer (R≠T≠F)**
  - Each contributor has a different role from others.

**Platform**

- **JDT**
• Result

```
Platform

JDT
```

```
Ratio of assigned bug

Ratio of assigned bug

days (log scale)

days (log scale)

R=T=F
R=T≠F
R≠T=F
R≠T≠F

R=T=F
R=T≠F
R≠T=F
R≠T≠F
```
Case study (2)
Bug management patterns in Eclipse Platform and JDT

- RQ4: *How do the bug management patterns impact the time to fix bugs?*

```plaintext
Reporter=Triager=Fixer (R=T=F)
One contributor plays all of the roles.

Reporter ≠ Triager=Fixer (R≠T=F)
One contributor serves as triager and fixer.

Reporter=Triager ≠ Fixer (R=T≠F)
One contributor asks another contributor to fix a bug.

Reporter ≠ Triager ≠ Fixer (R≠T≠F)
Each contributor has a different role from others.
```
• Result
• Result

Platform

<table>
<thead>
<tr>
<th>Days (log scale)</th>
<th>Ratio of fixed bugs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>100</td>
<td>0.0</td>
</tr>
<tr>
<td>1000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

JDT

<table>
<thead>
<tr>
<th>Days (log scale)</th>
<th>Ratio of fixed bugs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>0.0</td>
</tr>
<tr>
<td>100</td>
<td>0.0</td>
</tr>
<tr>
<td>1000</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Diagram descriptions:
- Reporter=Triager=Fixer (R=T=F): One contributor plays all of the roles.
- Reporter≠Triager=Fixer (R≠T=F): One contributor serves as triager and fixer.
- Reporter=Triager≠Fixer (R=T≠F): One contributor asks another contributor to fix.
- Reporter≠Triager≠Fixer (R≠T≠F): Each contributor has a different role from others.
Discussions (1)
Summary of our findings

RQ1
When a triager makes a bug report as a reporter, the time to assign a bug fixing task is 17–47% faster than a regular reporter.

Task Assignment

RQ2
When a triager assigns a bug fixing task to himself, he can fix the bug around two times faster than other developers.

Bug Fix

RQ3
Surprisingly when the triager assigns a task to himself, he needs 48%–58% longer time for the assignment than when he assigns it to other developers.

RQ4
The pattern [R≠T≠F] exhibits the worst performance in bug fixing.
• The boxplot of \([R\neq T\neq F]\) had the widest distribution.
  – This implies that in some cases the pattern works better than other patterns.
Discussions (2)
The impact of discussions among developers

- Discussions about bugs before bug report assignment made a difference in the bug-fixing performance.
Discussions (3)
Other factors that would impact the time to fix

- There are many other factors that would impact the time to fix bugs.

<table>
<thead>
<tr>
<th>factor</th>
<th>metrics (variable name)</th>
<th>scale</th>
<th>descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>bug</td>
<td>Component</td>
<td>nominal</td>
<td>component name specified in the bug report</td>
</tr>
<tr>
<td></td>
<td>Priority</td>
<td>nominal</td>
<td>priority for fixing the bug</td>
</tr>
<tr>
<td></td>
<td>Severity</td>
<td>nominal</td>
<td>severity of the reported bug</td>
</tr>
<tr>
<td></td>
<td>Milestone</td>
<td>nominal</td>
<td>whether or not a milestone is specified in the bug report</td>
</tr>
<tr>
<td></td>
<td>DescriptionWords</td>
<td>interval</td>
<td>number of words in “Description” in the bug report</td>
</tr>
<tr>
<td></td>
<td>CommentsCount</td>
<td>interval</td>
<td>number of comments in the bug report</td>
</tr>
<tr>
<td></td>
<td>CommentsWords</td>
<td>interval</td>
<td>number of words in comments</td>
</tr>
<tr>
<td></td>
<td>AttachmentsCount</td>
<td>interval</td>
<td>number of attachments (e.g., patches and screen shots)</td>
</tr>
<tr>
<td></td>
<td>DependsOnCount</td>
<td>interval</td>
<td>number of bugs which must be resolved before the reported bug</td>
</tr>
<tr>
<td></td>
<td>BlocksCount</td>
<td>interval</td>
<td>number of other bugs which are blocked by the reported bug</td>
</tr>
<tr>
<td></td>
<td>CCCount</td>
<td>interval</td>
<td>number of users who might be interested in the bug report</td>
</tr>
<tr>
<td>day and time</td>
<td>AssignTime</td>
<td>interval</td>
<td>time to assign the bug fixing task to a developer (i.e., $T_{assignment}$)</td>
</tr>
<tr>
<td></td>
<td>AssignedMonth</td>
<td>interval</td>
<td>month in which the bug fixing task was assigned to a developer</td>
</tr>
<tr>
<td></td>
<td>AssignedDay</td>
<td>interval</td>
<td>day in which the bug fixing task was assigned to a developer</td>
</tr>
<tr>
<td></td>
<td>AssignedWeekEnd</td>
<td>nominal</td>
<td>whether or not the bug fixing task was assigned in the weekend</td>
</tr>
<tr>
<td>stakeholder</td>
<td>Reporter</td>
<td>nominal</td>
<td>email address of the reporter (who reports the bug)</td>
</tr>
<tr>
<td></td>
<td>Triager</td>
<td>nominal</td>
<td>email address of the triager (who triages the bug)</td>
</tr>
<tr>
<td></td>
<td>Fixer</td>
<td>nominal</td>
<td>email address of the fixer (who resolves the bug)</td>
</tr>
<tr>
<td></td>
<td>Pattern</td>
<td>nominal</td>
<td>bug management pattern used in fixing the bug (main scope of this paper)</td>
</tr>
</tbody>
</table>
Discussions (3)
Other factors that would impact the time to fix

- We analyzed which metrics contributed to our prediction (logistic regression) model.

<table>
<thead>
<tr>
<th>factor</th>
<th>metrics</th>
<th>deviance residuals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(variable name)</td>
<td>60 day</td>
</tr>
<tr>
<td>bug</td>
<td>Component</td>
<td>263.06</td>
</tr>
<tr>
<td></td>
<td>Priority</td>
<td>9.22</td>
</tr>
<tr>
<td></td>
<td>Severity</td>
<td>1.38</td>
</tr>
<tr>
<td></td>
<td>Milestone</td>
<td>6.31</td>
</tr>
<tr>
<td></td>
<td>DescriptionWords</td>
<td>0.42</td>
</tr>
<tr>
<td></td>
<td>CommentsCount</td>
<td>3.39</td>
</tr>
<tr>
<td></td>
<td>CommentsWords</td>
<td>4.56</td>
</tr>
<tr>
<td></td>
<td>AttachmentsCount</td>
<td>3.84</td>
</tr>
<tr>
<td></td>
<td>DependsOnCount</td>
<td>6.26</td>
</tr>
<tr>
<td></td>
<td>BlocksCount</td>
<td>0.70</td>
</tr>
<tr>
<td></td>
<td>CCCount</td>
<td>12.08</td>
</tr>
<tr>
<td>day and time</td>
<td>AssignTime</td>
<td>4.71</td>
</tr>
<tr>
<td></td>
<td>AssignedMonth</td>
<td>9.87</td>
</tr>
<tr>
<td></td>
<td>AssignedDay</td>
<td>0.61</td>
</tr>
<tr>
<td></td>
<td>AssignedWeekEnd</td>
<td>0.10</td>
</tr>
<tr>
<td>stakeholder</td>
<td>Reporter</td>
<td>7.23</td>
</tr>
<tr>
<td></td>
<td>Target</td>
<td>8.69</td>
</tr>
<tr>
<td></td>
<td>Fixer</td>
<td>76.26</td>
</tr>
<tr>
<td></td>
<td>Pattern</td>
<td>154.49</td>
</tr>
</tbody>
</table>
Discussions
Threats to Validity

• Only three years (from 2007 to 2009) bug report data without reassignments
  – Such data selection might bring bias against the complete picture of open source development

• Only the two open source projects
  – The Eclipse projects is large enough, but they have developers who are fully employed by IBM
  – The user base of the Eclipse products is different from that of other products such as Mozilla
Conclusion and future work

• A need for better ways to communicate and share knowledge between the different individuals.
  – In cases where all roles were played by different individuals, the efficiency of the bug fixing was negatively impacted.
  – Communication appears to have a positive impact on speeding up bug fixing time even when every role is played by different individuals.

• Our future work includes investigating other projects and other factors (e.g., complexity of bugs).
Questions?

- Email: masao@sys.wakayama-u.ac.jp
- HP: http://oss.sys.wakayama-u.ac.jp
Discussions
Other factors that would impact the time to fix

• We created a prediction model based on logistic regression to quantify the relationships between the factors.
  – All the F1-values of our results also perform the result which is predicted by using randomly selected independent variable.

<table>
<thead>
<tr>
<th>prediction accuracy of our logistic regression model</th>
<th>prediction period</th>
<th>precision</th>
<th>recall</th>
<th>F1-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>in a day</td>
<td>68.14</td>
<td>38.22</td>
<td>48.97</td>
<td></td>
</tr>
<tr>
<td>in a weak</td>
<td>67.90</td>
<td>76.66</td>
<td>72.02</td>
<td></td>
</tr>
<tr>
<td>in a month</td>
<td>76.67</td>
<td>98.77</td>
<td>86.33</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>improvement rate against random prediction</th>
<th>prediction period</th>
<th>precision</th>
<th>recall</th>
<th>F1-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>in a day</td>
<td>66.68%</td>
<td>-6.50%</td>
<td>19.80%</td>
<td></td>
</tr>
<tr>
<td>in a week</td>
<td>14.22%</td>
<td>28.95%</td>
<td>21.14%</td>
<td></td>
</tr>
<tr>
<td>in a month</td>
<td>2.35%</td>
<td>31.86%</td>
<td>15.24%</td>
<td></td>
</tr>
</tbody>
</table>
Our focus:
Relations between the individuals

• The triager plays a very important role in the bug management process

• The triager needs to
  – have a good understanding of the bug report
  – assign the bug fixing task to the most appropriate developer who can fix the bug as quickly as possible