Visualizing evolution of software issue-tracking practices

Jialiang Xie
Peking University
Beijing 100871, China
xiejl11@sei.pku.edu.cn

Audris Mockus
Avaya Labs Research
233 Mt Airy Rd, Basking Ridge, NJ
audris@avaya.com

Minghui Zhou
Peking University
Beijing 100871, China
zhmh@pku.edu.cn

Issue tracking systems are commonly used in research and practice to measure key aspects of software production. However, the interpretation of traces left by these systems strongly depends on ways (practices) the project team has used the issue tracking system. Such practices may also change over time, thus making it difficult to rely on an interview-derived interpretation of these practices. We set out to improve the quality of the issue-tracking-system-derived data and to improve the general understanding of issue-tracking practices by designing and implementing a visualization tool designed to reverse-engineer these practices and their evolution.

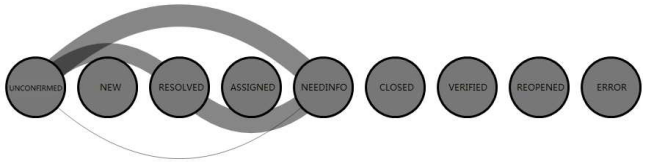

Figure 1: State transitions

1. DESCRIPTION

The tool has the two main views: the issue state transition and the timeline. The issue state transition diagram shows states as circles and transitions between states as arcs, with the thickness of the arc indicating the number of transitions in the selected sample(Figure 1). The two views are linked. In particular, the change of the selected time range results in the animation of the evolution of the transitions among states. Specific aspects of the evolution are shown in the timeline view (area chart,see Figure 2). It shows evolution of two statistics and their ratio, for example, the total number of issues and the number of issues that contain a specific pattern (e.g., had RESOLVED state followed by NEW state). In addition to graphical selection for various dimensions, regular expressions can be used to select desired transition patterns and the history of selections can be saved to store the analysis for future reference.

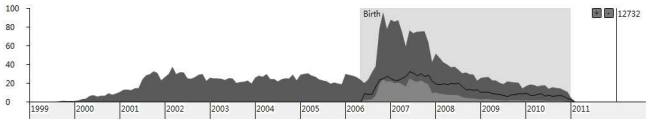

Figure 2: Timeline

2. USAGE SCENARIO

By applying our tool on GNOME issues we discovered a number of changes in issue-handling practices. In particular, the timeline view, shows a dramatic increase of new issues in the July of 2006 (Figure 2). By dragging the time range in the timeline view, we selected new issues between August, 2006 and December, 2010. State diagram showed that approximately 40% of invalid reports had resolution "INCOMPLETE". By selecting only these reports, the two main patterns emerged in the issue state transition diagram: UNCONFIRMED $\Longrightarrow$ NEEDINFO $\Longrightarrow$ RESOLVED and UNCONFIRMED $\Longrightarrow$ RESOLVED (Figure 1). The timeline view showed that the first pattern had 95% of the NEEDINFO issues transitioned to RESOLVED between August, 2006 and March, 2007. This shows that only rarely issue reporters would provide information needed to reproduce the issue. For the second pattern the issues were mostly after March, 2007. We conclude from these observations that when GNOME encountered an overflow of incomplete reports, the teams were exhausted from inspecting NEEDINFO reports. GNOME, therefore changed workflow to skip the NEEDINFO state and transferred those reports into RESOLVED. This optimization improved the efficiency dealing with incomplete reports. Our conclusions were verified by the administrator of GNOME.

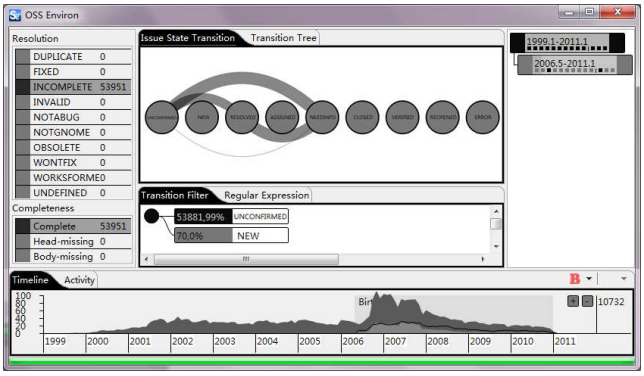

Figure 3: The screen-shot of the tool

3. CONCLUSION

We introduce a visualization tool designed to improve the quality of the issue-tracking-system-derived data and to reverse-engineer and improve issue-tracking practices. Using GNOME project data we demonstrate how our tool can be used to discover the evolution of issue tracking practices.