ELTE Eötvös Loránd University, Faculty of Informatics

Confirming the Impact of Warning Message Quality in the Clang Static Analyzer

Kristóf Umann[1], Zoltán Porkoláb[2]

[1]Department of Programming Languages and Compilers [2]Ericsson

Background

Since the 2010s, numerous studies have emerged on issues that hinder the broader adoption of static analysis tools. Among other things, the large volume of false positives, poor configurability, and poor quality warning messages are leading causes. This last issue remained in an academic blind spot for years until recently.

After a successful Google Summer of Code project in 2019 in the Clang Static Analyzer, we managed to add control dependency analysis to the Clang Static Analyzer, improving entire classes of bug reports emitted by the tool. Still, while the community of contributors and the sentiment from the user base were positive about this, and other QOL changes, we never had conclusive proof in hand to show that our changes did, in fact, improve the user experience.

Measurement method

We assembled a suite of 5 programs, synthesized from bugs found in real-world software. For each program, we created 2-3 bug reports, a total of 13, either by using different configurations of the Clang Static Analyzer, or by hand. Then, we created a survey form, which we distributed internally in Ericsson, and in Hungary based C++ meetups.

Respondents were randomly shown one of the 2-3 reports for each program, but they did not how many reports, if any others existed. They were asked to rate each step (bubble) in the report on a scale of 1-4. They could also, optionally, leave additional comments.

We collected a total 64 responses across 10 countries.

Bug report comparison example

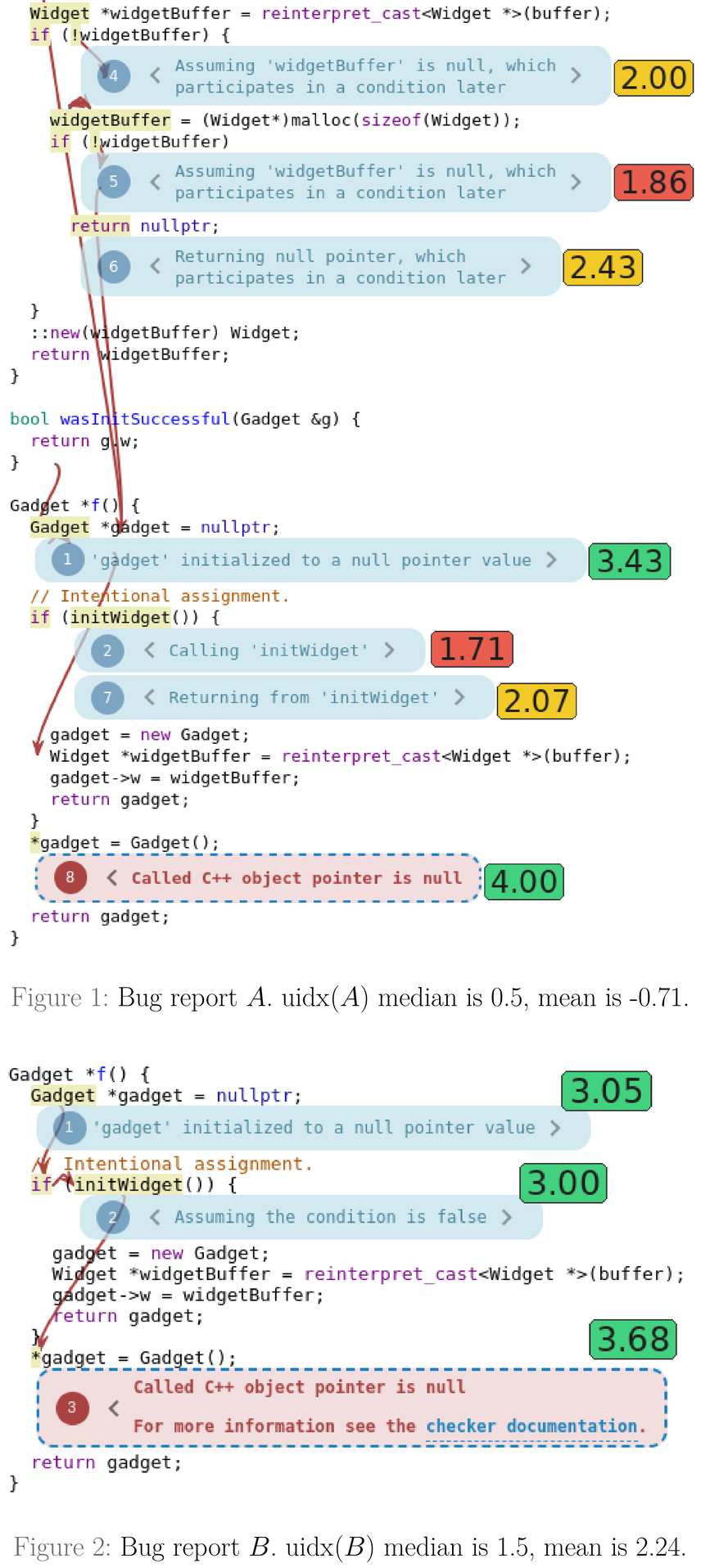

Figure 1: Bug report A. uidx(A) median is 0.5, mean is -0.71.

Figure 2: Bug report B. uidx(B) median is 1.5, mean is 2.24.

The Mann-Whitney U-test rejects the null hypothesis $Hn_{A,B}$, indicating that report B is significantly better than A with a large effect size ($p = 0.01$, Cliff's $\delta = -0.52$).

Analysis procedure

We designed a *informational utility index* (uidx). For a bug report with R responses and a length of N, this is a matrix of $S \in \{1, 2, 3, 4\}^{R \times N}$. For each $j \in [1, R]$ response given to bug report A, we calculated

$$\text{uidx}_j(A) = \sum_{i=1}^{N} (S_{i,j} - c)$$

where $c = 2.5$ is the center point of a 1-to-4 scale. As such, bug reports with higher uidx scores are better. We conducted Mann-Whitney U tests against the null hypothesis $Hn_{A,B}$ that there is no stochastic difference between uidx(A) and uidx(B) for bug reports A and B.

Results

Our results back up decades old contributor suspicions with regards to bug report quality. We showed that (1) incorrectly identifying a function as interesting can be shown to negatively impact bug report quality. To the opposite, we also showed that (2) functions misidentified as non-interesting is also detrimental.

To keep bug report length in check, we showed that (3) leading notes – those that explain the flow of control to the function that is solely at fault of a bug – can, on occasion, be safely eliminated.

We also supported the idea behind (4) function summaries being preferred in place of a multi-step explanation in certain cases, especially for low-level function calls.

You can find more examples, data, and much deeper analysis from this document starting from Section 3.4: https://szelethus.web.elte.hu/dissertation_final.pdf

Acknowledgement

Project no. C2314106 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the KDP-2023 funding scheme.

Information

Author: Kristóf Umann
Contact: szelethus@inf.elte.hu
Date: 2026. April 14.