Usability of Error Messages for Novice Programmers

Paul A. Schliep
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
schli202@morris.umn.edu

ABSTRACT

This paper discusses how well error messages can help novice programmers find mistakes in their programs and what aspects make an error message more user-friendly for introductory computer science students. I will also be discussing several methodologies and programs developed to help improve the experience a novice programmer has when attempting to understand causes for errors.

1. KEY POINTS

The question being addressed in this research is to figure out how well error messages in programming languages help introductory students find mistakes in their programs. This question is being addressed through various methods of evaluating the usability of the error messages and the methodologies in which they can be or potentially have been improved. The methods of evaluating the error messages include usability studies, trials, and developed rubric to assess them, and some of the methodologies that suggested improvements to the messages include a set of suggested principles for compiler errors, enhancement of syntax errors, and proposed recommendations for exception messages.

The sources for this paper each propose or analyze something new to help address this problem while still being related. On Compiler Error Messages: What They Say and What They Mean discuss the current state of compiler error messages, how that affects novice programmers, and what they suggest to resolve those issues. Enhancing Syntax Error Messages Appears Ineffectual talks about the implications of syntax error messages are on how introductory CS students can fix errors in their programs and they present an enhanced library of compiler messages specifically for syntax errors and analyze its effectiveness. Mind Your Language: On NovicesâĂŹ Interactions with Error Messages and Measuring the Effectiveness of Error Messages Designed for Novice Programmers both evaluate error messages in the Racket programming language and discuss how well they help introductory students fix mistakes in their programs, and how to apply their developed rubric to improve these error messages. Each paper discusses an analysis of different types of error messages and in what ways they may be resolved, and by studying these analyses and their respective methodologies, we can be able to compare and contrast their methods and results.

Error message usability has been tested for many years, but it has gained more attraction in recent years with the rising studies in HCI. Through these studies, developers and professors have been able to implement error message libraries and test them, but not always with conclusive results as seen in my source, Enhancing Syntax Error Messages Appears Ineffectual where the number of studentsâĂŹ syntax errors did not fall.

In order to explain some of the analyses of the error messages, the reader will need to be informed of concepts such as usability studies and compiler errors.

2. REFERENCES

[1] P. Denny, A. Luxton-Reilly, and D. Carpenter. Enhancing syntax error messages appears ineffectual. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education*, ITiCSE '14, pages 273–278, New York, NY, USA, 2014. ACM. *Discusses syntax errors and helps coincide with compiler source.*

[2] B. Hartmann, D. MacDougall, J. Brandt, and S. R. Klemmer. What would other programmers do: Suggesting solutions to error messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1019–1028, New York, NY, USA, 2010. ACM. *Presents a framework for adding hints for error messages, can probably mention hints in methodologies.*

[3] B. S. Isa, J. M. Boyle, A. S. Neal, and R. M. Simons. A methodology for objectively evaluating error messages. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '83, pages 68–71, New York, NY, USA, 1983. ACM. *can use this in methodologies and probably as a background in several other areas.*

[4] G. Marceau, K. Fisler, and S. Krishnamurthi. Measuring the effectiveness of error messages designed for novice programmers. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education*, SIGCSE '11, pages 499–504, New York, NY, USA, 2011. ACM. *Helps with analyses and results of usability of error messages and presents a nice rubric*

UMM CSci Senior Seminar Conference, December 2013 Morris, MN.

for improving error messages.

[5] G. Marceau, K. Fisler, and S. Krishnamurthi. Mind your language: On novices' interactions with error messages. In *Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software*, Onward! 2011, pages 3–18, New York, NY, USA, 2011. ACM. *This paper along with the next help with a lot of the analyses and results of error message research.*

[6] C. Murphy, E. Kim, G. Kaiser, and A. Cannon. Backstop: A tool for debugging runtime errors. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '08, pages 173–177, New York, NY, USA, 2008. ACM. *Can possibly mention this in the conclusions as a background source.*

[7] V. J. Traver. On compiler error messages: What they say and what they mean. 2010.

[7] [5] [4] [1] [2] [6] [3]