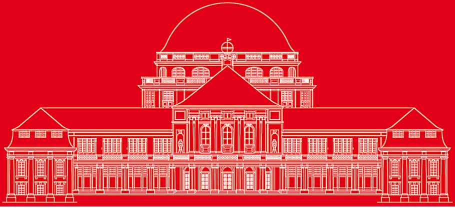
Tim Krämer, Pascal Rost
[7kraemer,7rost] @informatik.uni-hamburg.de

Paper presentation: "Programmers' Build Errors: A Case Study (at Google)"

13. November 2014



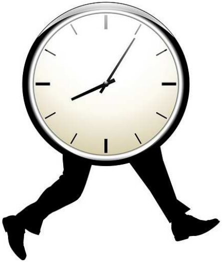
Agenda

- Motivation
- Study Design
- Results
  - □ Findings
  - □ Implications
- Conclusion

Programmers' Build Errors: A Case Study (at Google) by Hyunmin Seo, Caitlin Sadowski, Sebastian Elbaum, Edward Aftandilian, Robert Bowdidge. To appear on International Conference on Software Engineering (ICSE) (2014)

Motivation

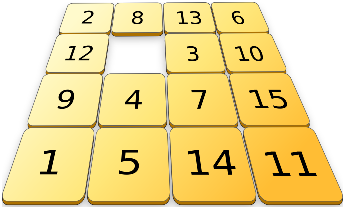
- increase programmer productivity
  - do slow compilers reduce efficiency?
  - failed compiles may let programmers loose focus...
- edit-compile-debug cycle
  - key part: keeping build process fast!
- How to improve tools to support developers?



Study Design

- 26.6 million builds
- 18 thousand developers
- 9 months period

- filtered out extremes (bots, janitors, non-programmers)
- only standard developers



Study Design

- C++ (LLVM Clang)
- Java (javac)

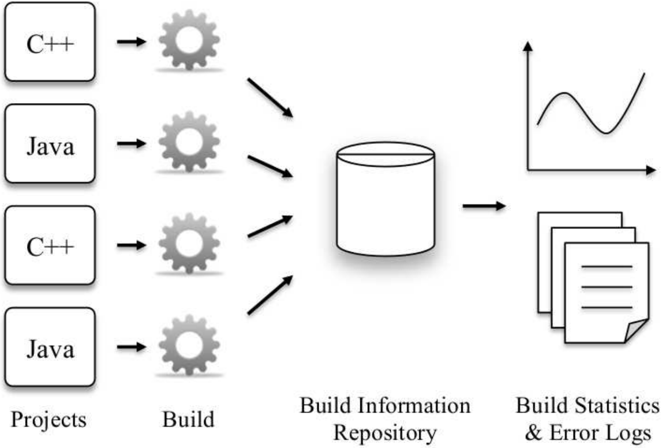
- cloud-based build process
- Analysis of persistent logs
- Categorization of build errors

Study Design - Metrics

collected log files of cloud-like build system:

1. total of builds
2. failure ratio of build
3. most often occurring error types
4. time needed to fix errors

Study Design



Study Design - Research Questions

1. How often do builds fail?
2. Why do builds fail?
3. How long does it take to fix broken builds?

Results - Findings

1. How often do builds fail?
   - □ Java: 28,5%, C++: 38,4%
2. Why do builds fail?
   - □ Dependency-related errors: Java: 64.71%, C++: 52.68%
3. How long does it take to fix broken builds?
   - □ 75% of top 25 errors were fixed after 2 builds.

- 10% of error types account for 90% percent of build failures

Results - Implications

- Practitioners
  - Value of assisting tools
- Tool builders
  - Enhance/create tools to resolve dependency errors
- Researchers
  - Highlights the importance of building
  - Quantification of resolution times

Conclusion

- largest study about building errors
- greatest potential payoff: tools to resolve dependency errors

Thank you for listening!

Questions? Discussion!