

Time-traveling through the GCC PR database and testsuite

Alexandre Oliva

<https://www.lx.oliva.nom.br/>

oliva@gnu.org, oliva@adacore.com

AdaCore

GNU Tools Cauldron, September, 2025

Copyright 2025 AdaCore (last changed in September 2025)

This work is licensed under the [Creative Commons BY-SA 4.0 International](https://creativecommons.org/licenses/by-sa/4.0/) License.

[git://git.lx.oliva.nom.br/blog](https://git.lx.oliva.nom.br/blog) pres/timetravel/



Context

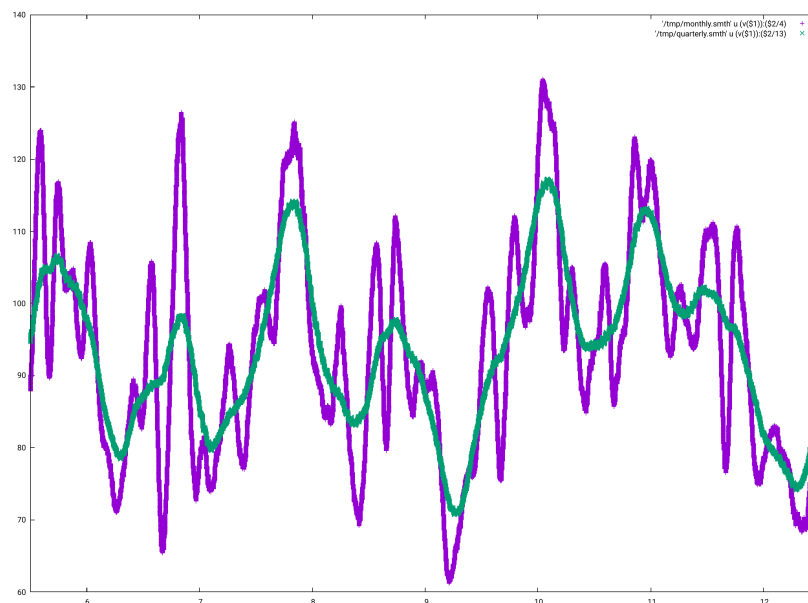
- Recommendations for stable compilation
- **Techniques** to select:
 - a compiler release
 - optimization flags
- Resources:
 - GCC PR database
 - GCC testsuite

Selecting a compiler release

- Candidates based on GCC 7. to 11.
- Latest is safest? Tension:
 - Latest accumulates more fixes
 - Earliest is more proven
- Data source: PR database
- Hypothesis: $\uparrow$ PRs $\Rightarrow$ $\downarrow$ stable

Bug reporting frequency

- Δ s smoothed over 4|13 weeks, 6. to 12.
- Visible releases
- $\uparrow$ PRs $\nRightarrow$ $\uparrow$ bugs
- $\uparrow$ PRs $\Rightarrow$ $\uparrow$ activity
- Inconclusive
- Recent (not shown) $\uparrow$ backported PRs; great?



Selecting optimization options

- -O1 vs -O2 tension:
 - -O1 runs fewer passes
 - -O2 is more widely used
- Identify and disable risky optimizations?
- Data source: GCC testsuite

Torture Testsuite Time Travel

- “Future” testsuite on older compiler
- 16k eligible added tests
- 13k failures, after compat adjustments
- Narrow down to torture tests

Avoiding risky optimizations?

- Add optimization(-disabling) torture options
 - -fno-inline
 - -fno-strict-aliasing
 - -fno-strict-overflow
 - other flags
- Analyze **differences**, not failures
 - 117 tests varied results across opts

Failure-causes by optimization level

	Only		Both	Neither
	-O1	-O2		
Compilation	1	4	17	5
Execution	3	12	12	7
Total	4	16	29	12

Effects of optimization flags

	Avoided		Exposed
	Comp	Exec	
-fno-inline	4	2	2
-fno-strict-aliasing	0	6	0
-fno-strict-overflow	1	0	3
-fno-common	0	0	1

Conclusions

- Interesting techniques
- Per-release PR frequency?
- Insights from testsuite
 - Not statistically significant
 - Confirm/dispute intuitions
- Useful to see the “future”?

Thank you!

AdaCore

<https://www.lx.oliva.nom.br/>

oliva@gnu.org, oliva@adacore.com

Questions?

