The Five Stages of Benchmark Loss

Matthew Tippett

Michael Larabel

Agenda

- Some Background
- Some Classic Examples
- The Five Stages of Benchmark Loss
- Lessons Learnt
- Conclusions
- Questions

Some Background

About Us

- Matthew Tippett
  - In and about Linux since 1992 (that's Linux 0.99pl15). Presented at conferences, ran user-groups and other fun stuff.
  - Got involved in Phoronix and Benchmarking while running the Linux graphics driver team at AMD.
- Michael Larabel
  - Founded Phoronix.com in 2004 and the parent company Phoronix Media. Focuses on Linux hardware reviews, drivers, and compatibility.

Phoronix Test Suite & Benchmarking

- Phoronix Test Suite evolved out of a set of in-house tools that provided reliable and reproducible benchmarks for the Phoronix.com website.
- 1.0 was released in 2008
- Has evolved into a generic test suite capable of supporting a wide range of testing requirements
  - Everything from system benchmarking to functional software testing.

Phoronix Test Suite & Benchmarking (2)

- The test suite has been picked up by many large technology vendors
  - pick a name in the Technology Sector's who's-who, and we're probably in use there.
- Phoronix Test Suite has driven the creation of extra systems and services
  - Phoromatic, Phoronix Global
- Commercial Services are available.
  - http://commercial.phoronix-test-suite.com/

Some comments on Benchmark Reporting

- Fundamentally any journalistic reporting is based on ***comparing*** items and ***contrasting*** their differences.
- With benchmarks, the whole intent is to ***compare*** the systems.
- Most benchmark reporting is all about the ***contrasting*** of the results; ie: Declaring a winner for the test or a full benchmark.

Some Comments on Benchmark Reporting (2)

- Interesting results either demonstrate an **impactful** difference or a **divisive** gap.
- One person's *impactful* is another person's *divisive*.
  - *Hence, when you report on a benchmark you will never win with the loser (but you always win with the winner)*

Winning and Losing

- When you **win**
  - People move on very quickly
  - *'See - validation that we rock'*
- When you **lose**
  - It gets a lot more complex, people can't move on.
    - Excuses, complaints, blame follow.
  - Phoronix sees lots of this.

Some Classic Examples

Names removed to protect the innocent

The Build with the Debug Symbols

- A number of distributions build with debug symbols ***early*** in the development cycle
- We start testing ***late*** in the development cycle
- Of *course* it's unfair to test the development version because they are **always** built with debug symbols…

The Slow Path by Default

- A Commercial Unix with amazing benchmarks results (vendor provided)
- The default compiler is 32 bit gcc
- The fast path is the vendors compiler in 64 bit
  - No documentation
  - Non-trivial reconfiguration
  - Non-discoverable

The Virtualized Accelerator

- The guest ran a database test about 100 times faster than the host
- Four projects involved, lots of assumptions about the correct or incorrect
- In the end, integrity had been traded for performance
  - Not everyone knew
  - (It's now fixed)

The Five Stages of Benchmark Loss

The Five Stages

- The following was inspired by the **Kübler-Ross model**; or the **Five Stages of Grief**.
- We have seen huge multi-nationals to small individual developers go through a fairly consistent set of reactions to losing a benchmark comparison.
- Reducing the painful part (the 2nd and 3rd stage) is what we are here for today.
  - We'd like to see the 1st stage stay since it's fun to watch :).

Stage 1: **Shock**

- The first reaction when you see an article, scanning down to see your competition (or ideological opposite; or fork from last year) wipe the floor with you in a particular test.
- Typical postings
  - 'WTF…', 'No Way…", "You lie…"
- Usually the fastest stage to pass.

Stage 2: **Denial**

- Shock moves to denial very quickly
  - Usually the second sentence in a posting on a benchmark loss
- Comments are usual baseless attacks without any analysis or technical basis
  - Slashdot commentators pretty much stop here
  - Usually starting with absolute words like “obviously”, “clearly”, etc.

Stage 3: **Discreditation**

- We're geeks, so we look for a technical reason for the loss
  - Most hit on the most probable reason that the results are not valid.
- Typical postings
  - 'We lost because of debug symbols'
  - 'They left it in the default config'
  - 'Obviously the problem is in this other component'
  - 'They don't know how to test'
- Most (non-slashdotters) don't ever leave this stage.

Stage 4: **Analysis**

- This is the first *emergent* stage.
- Facts are checked, issues are understood.
- Unfortunately, discussion and analysis usually have to be facilitated
- Eventually the underlying causes become understood.

Stage 5: **Acceptance**

- If you stay the course and make it through the analysis, you ultimately accept the result.
- The reason for the loss is internalized and implemented upstream
- Then you wait for the next benchmark to see if you win (or at least close the gap)

Lessons Learnt

and what we have done to help

Reproducibility

- Any benchmark should be reproducible by anyone.
- Phoronix Test Suite is highly reproducible
  - Results sets can be transferred and re-executed.
- Ironically, most people who don't get past stage 2 or 3 and never attempt to reproduce.

Out of the box experience

- Most systems can be tweaked and tuned for maximum performance.
- Most mere mortals have neither the skills, time or the awareness to tune an installation.
- Consequently, most of our testing is through a default config/install/etc.
  - Although we have offered many times to have a tuned head-to-head comparison, we have had virtually no takers :(

InfiniFUD™

- FUD that becomes a meme that is passed through the community.
- When facts are checked, the assumption quashed, but the FUD remains in the communities consciousness and returns again and again and again - infinitum
- To kill the InfiniFUD you have to make it to stage 4 (Analysis)
  - most don't make it that far

Posed Questions and Answers

- When there is a large gap, people want to understand why.
  - Most spectators of course want a complete analysis for any delta, but won't invest the time themselves.
- In general, the company or team behind a product or project can usually answer those questions
  - But of course, a lot of time they don't get beyond stage 1, 2 or 3.

Posing Questions (2)

- In most cases a performance delta is confluence of multiple causes
  - To get to the bottom of things, involves lots of emails, lots of cross-referencing, lots of pain, lots of time.
  - But in almost all cases where we have invested the effort to dig through and correlate the causes, there have been fixes applied!

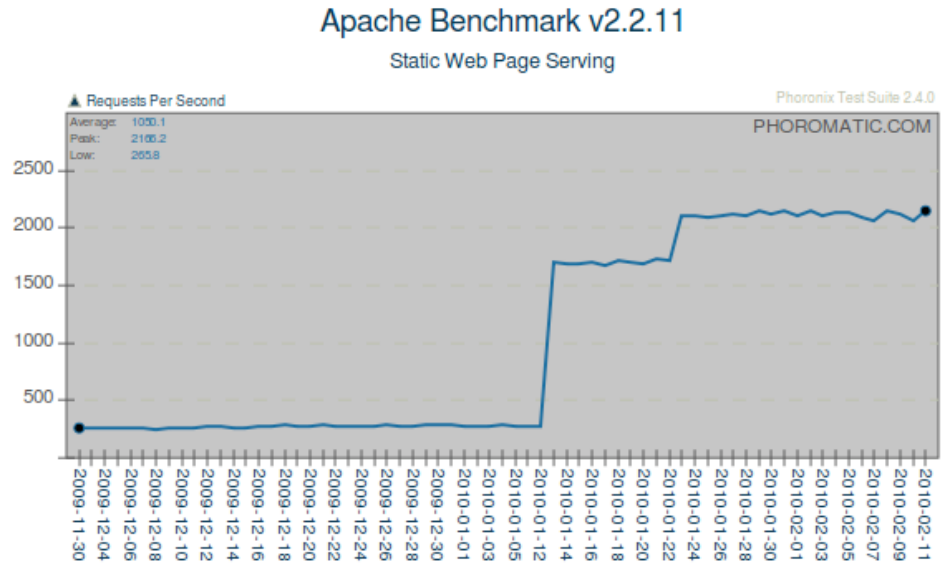
Surprises

- Results shouldn't be surprising to the either the project or the aggregator (distribution)
- Unfortunately competitive benchmarking is a relatively foreign concept for a lot of people and so results are almost always surprising.
  - Fortunately, it makes simple comparisons very **impactful** or **divisive** as people hit stage 1 very quickly.
    - As a result - it makes good news copy

Performance Management

- If you have regular builds, regular automated testing, there shouldn't be surprises.
- Phoronix currently has a tracker running against daily builds of kernels at
  - http://www.phoromatic.com/kernel-tracker.php
- There is virtually no excuse for any performance sensitive project to **NOT** have a performance management system in place.

Performance Management (2)



- This shouldn't be a surprise to the kernel community. Unfortunately it probably is.
  - And hopefully, it isn't a sign of a really bad regression elsewhere.

Regression Management

- Now that we have performance management, managing performance regressions is easy.
  - Same for functional regressions
- Keep developers out of the code, use tools to run repeated tests and bisect the code changes over the ordered set of builds.
- Sounds simple, but tools are rare

Conclusions

The Take-aways…

- If a comparison shows an unexpected issue
  - Move beyond the emotive to the cognitive
  - **Understand** the delta
- Empathize with the naïve user
  - 80% of your customers can't or won't tune for max performance; out-of-the-box is **critical**.
- Performance and Regression management is ***easy***
  - There is no excuse (we can help)

Questions?

Feel free to contact us:

Matthew Tippett matthew@phoronix.com
Michael Larabel michael@phoronix.com

http://commercial.phoronix-test-suite.com/