SIGPLAN Empirical Evaluation Checklist

*This checklist is meant to **support** informed judgement, not **supplant** it.*

Clearly Stated Claims
Example Violations

""

Claims not Explicit
Claims must be explicit in order for the reader to assess whether the empirical evaluation supports them. Missing claims cannot possibly be assessed. Claims should also aim to state not just what is achieved but how. A common example of an implicit claim is the generality that is implied by unstated limitations.

"precise analysis"

Claims not Appropriately Scoped
The truth of a claim should clearly follow from the evidence provided. Claims that are not fully supported mislead readers. 'Works for all Java' is over-broad when based on a subset of Java. Other examples are 'works on real hardware' when evaluating only with (unrealistic) simulation, and 'automatic process' when requiring human intervention.

"devs liked it"

Fails to Acknowledge Limitations
A paper should acknowledge its limitations to place the scope of its results in context. Stating no limitations at all, or only tangential ones, while omitting the more relevant ones may mislead the reader into drawing overly-strong conclusions. This could hold back efforts to publish future improvements, and may lead researchers down wrong paths.

Suitable Comparison
Example Violations

"hiked 4k mountain"

Fails to Compare Against Appropriate Baseline
Empirical evidence for a claim that a technique/system improves upon the state-of-the-art should include a comparison against an appropriate baseline. The lack of a baseline means empirical evidence lacks context. A 'straw man' baseline that is misrepresented as state-of-the-art is also problematic, as it would inflate apparent benefit.

"finished before you"

Comparison is Unfair
Comparisons to a competing system should not unfairly disadvantage that system. Doing so would inflate the apparent advantage of the proposed system. For example, it would be unfair to compile the state-of-the-art baseline at -O0 optimization level, while using -O3 for the proposed system.

Principled Benchmark Choice
Example Violations

"low sync overhead"

Inappropriate Suite
Evaluations should be conducted using appropriate established benchmarks where they exist so that claimed results are more likely to generalize. Not doing so may yield results that are not sufficiently general. Established suites should be used in context; it would be wrong to use a single-threaded suite for studying parallel performance.


"we used TheBench"

Unjustified Use of Non-Standard Suite(s)
The use of standard benchmark suites improves the comparability of results. However, sometimes a non-standard suite, such as one that is subsetted or homegrown, is the better choice. In that case, a rationale, and possible limitations, must be provided to demonstrate why using a standard suite would have been worse.

"large speedup"

Kernels instead of Full Applications
Kernels can be useful and appropriate in a broader evaluation. However, a claim that a system benefits applications should be tested on such applications directly, and not only on micro-kernels, which may lack important characteristics of full applications.

Adequate Data Analysis
Example Violations

10 s
"10 seconds"

Insufficient Number of Trials
Modern systems with non-deterministic performance properties may require many trials (e.g., of a single time measurement) to characterize their behavior adequately. Failure to do so risks treating noise as signal. Similarly, more trials may be needed to get the system into an intended state (e.g., into a steady state that avoids warm-up effects).

147 s
2 s
1 s
"mean of 50 seconds"

Inappropriate Summary Statistics
Summary statistics can usefully characterize many results. But they should be selected carefully, because each statistic presents an accurate view only under appropriate circumstances. An inappropriate summary may amplify noise or hide an important trend.

10 s
90 s
95 s
5 s
"50 seconds"

No Data Distribution Reported
A measure of variability (e.g., variance, std. deviation, quantiles) and/or confidence intervals is needed to understand the distribution of the data. Reporting just a measure of central tendency (e.g., a mean or median) can mislead the reader, especially when the distribution is bimodal or has significant variance.

Relevant Metrics
Example Violations

$$$
"energy consumed"

Indirect or Inappropriate Proxy Metric
Proxy metrics can substitute for direct ones only when the substitution is clearly, explicitly justified. For example, it would be misleading and incorrect to report a reduction in cache misses as a substitute for actual end-to-end performance or energy consumption.

devs
testers
"devs were satisfied"

Fails to Measure All Important Effects
All important effects should be measured to show the true cost of a system. For example, compiler optimizations may speed up programs at the cost of drastically increasing compile times of large systems, so the compile time should be measured as well as the program speedup. Failure to do so distorts the cost/benefit of the system.

Appropriate and Clear Experimental Design
Example Violations

Version ?
OS ?
Hardware ?
"sped up apache"

Insufficient Information to Repeat
Experiments evaluating an idea need to be described in sufficient detail to be repeatable. All parameters (including default values) should be included, as well as all version numbers of software, and full details of hardware platforms. Insufficient information impedes repeatability and comparison of future ideas and can hinder scientific progress.

SuperCPU
150 Watt
"for sensor net"

Unreasonable Platform
The evaluation should be on a platform that can reasonably be said to match the claims; otherwise, the results of the evaluation will not fully support the claims. For example, a claim that relates to performance on mobile platforms should not have an evaluation performed exclusively on servers.

$
$$$$
"10 times faster"

Ignores Key Design Parameters
Key parameters should be explored over a range to evaluate sensitivity to their settings. Examples include the size of the heap when evaluating garbage collection and the size of caches when evaluating a locality optimization. All expected system configurations (e.g., from warmup to steady state) should be considered.

wait
MD
"prompt treatment"

Gated Workload Generator
Load generators for typical transaction-oriented systems should be 'open loop', to generate work independent of the performance of the system under test. Otherwise, results are likely to mislead because real-world transaction servers are usually open-loop.

train
19+3   22
7+1   8
test
19+3   22✓
7+1   8✓
"perfect"

Tested on Training Set
When a system aims to be general but was developed by training on or with close consideration of specific examples, it is essential that the evaluation explicitly perform cross-validation, so that the system is evaluated on data distinct from the training set.

Appropriate Presentation of Results
Example Violations

"have up to 4 leaves"

Misleading Summary of Results
The summary of the results must reflect the full range of their character to avoid misleading the reader. For example, it is not appropriate to summarize speedups of 4%, 6%, 7%, and 49% as 'up to 49%'. Instead, the full distribution of results must be reported.

"B is much faster"

Inappropriately Truncated Axes
Graphs provide a visual intuition about a result. A truncated graph (with an axis not including zero) will exaggerate the importance of a difference. 'Zooming' in to the interesting range of an axis can sometimes aid exposition, but should be pointed out explicitly to avoid being misleading.

"Slowed A a little, sped up B lots"

Ratios Plotted Incorrectly
Incorrectly plotted ratios badly mislead visual intuition. For example, 2.0 and 0.5 are reciprocals, but their linear distance from 1.0 does not reflect that, so plotting those numbers on a linear scale significantly distorts the result. This misleading effect can be avoided either by using a log scale or by normalizing to the lowest (highest) value.

"9.36 s startup time"

Inappropriate Level of Precision
Measurements reported at a proper level of precision reveal relevant information. Under-precise reports may hide such information, and over-precise ones may overstate the accuracy of a measurement and obscure what is relevant. For example, reporting '49.9%' when the experimental error is +/- 1% overstates the level of precision of the result.

Notes

Claims not Explicit   This includes *implied* generality — implied: *'works for all Java'*, but actually only on a static subset; implied: *'works on real hardware'*, but actually only works in simulation; implied: *'automatic process'*, but in fact required non-trivial human supervision; implied: *'only improves the systems' performance'*, but actually the approach requires breaking some of the system's expected behavior.

Fails to Acknowledge Limitations   One concern we have heard multiple times is that this example, previously titled *Threats to validity*, is not useful. The given reason is that *threats to validity* sections in software engineering papers often mention threats of little significance while ignoring real threats. This is unfortunate, but does not eliminate the need to clearly scope claims, highlighting important limitations. For science to progress, we need to be honest about what we have achieved. Papers often make, or imply, overly strong claims. One way this is done is to ignore important limitations. But doing so discourages or undervalues subsequent work that overcomes those limitations because that progress is not appreciated. Progress comes in steps, rarely in leaps, and we need those steps to be solid and clearly defined.

Fails to Compare Against Appropriate Baseline   An evaluation of an idea that improves upon the state-of-the-art should evaluate that idea against a baseline. This baseline could be a best-of-breed competitor, but should not be a straw man, e.g., something that once was, but is no longer, the state-of-the-art. The baseline could also be an unsophisticated approach to the same problem, e.g., a fancy testing tool is usefully compared against one that is purely random, in order to see whether it does better.

Comparison is Unfair   For example, the authors were unable to build the state-of-the-art baseline at the -O3 optimization level and used -O0 instead, while using -O3 for their system.

Inappropriate Suite   This includes misuse of incorrect established suite e.g. use of SPEC CINT2006 when considering parallel workloads.

Unjustified Use of Non-Standard Suite(s)   A concern we heard was that use of standard suites may lead to work that overfits to that benchmark. While this is a problem in theory, and is well known from the machine learning community, our experience is that PL work more often has the opposite problem. Papers we looked at often subset a benchmark, or cherry-picked particular programs. Doing so calls results into question generally, and makes it hard to compare related systems across papers. We make progress more clearly when we can measure it. Good benchmark suites are important, since only with them can we make generalizable progress. Developing them is something that our community should encourage.

   Note that *'benchmark'* in this category includes what is measured and the parameters of that measurement. One example of an oft-unappreciated benchmark parameter is timeout choice.

Inappropriate Summary Statistics   As particular best practices: The geometric mean should only be used when comparing values with different ranges, and the harmonic mean when comparing rates. When distributions have outliers, a median should be presented. There are many excellent resources available, including: *Common errors in statistics (and how to avoid them).* (Phillip I Good and James W Hardin, 2012), *What is a P-value anyway?: 34 stories to help you actually understand statistics.* (Andrew Vickers, 2010), and *Statistical misconceptions.* (Schuyler W Huck, 2009).

Ratios Plotted Incorrectly   For example, if times for a and b are 4 sec and 8 sec respectively for benchmark x and 6 sec and 3 sec for benchmark y, this could be shown as a/b (0.5, 2.0) or b/a (2.0, 0.5), where 1.0 represents parity. Although the results (0.5 & 2.0) are reciprocals, their distance from 1.0 on a linear scale is different by a factor of two (0.5 & 1.0), overstating the speedup. This is why showing ratios (or percentages) greater than 1.0 (100%) and less than 1.0 (100%) on the same linear scale is visually misleading.

FAQ

Why a checklist?   Our goal is to help ensure that current, accepted best practices are followed. Per the Checklist Manifesto, checklists help to do exactly this. Our interest is the good practices for carrying out empirical evaluations as part of PL research. While some practices are clearly wrong, many require careful consideration: Not every example under every category in the checklist applies to every evaluation – expert judgment is required. The checklist is meant to assist expert judgment, not substitute for it. 'Failure isn't due to ignorance. According to best-selling author Atul Gawande, it's because we haven't properly applied what we already know.' We've kept the list to a single page to make it easier to use and refer back to.

Why now?   When best practices are not followed, there is a greater-than-necessary risk that the benefits reported by an empirical evaluation are illusory, which harms further progress and stunts industry adoption. The members of the committee have observed many recent cases in which practices in the present checklist are not followed. Our hope is that this effort will help focus the community on presenting the most appropriate evidence for a stated claim, where the form of this evidence is based on accepted norms.

Is use of the checklist going to be formally integrated into SIGPLAN conference review processes?   There are no plans to do so, but in time, doing so may make sense.

How do you see authors using this checklist?   We believe the most important use of the checklist is to assist authors in carrying out a meaningful empirical evaluation.

How do you see reviewers using this checklist?   We also view the checklist as a way to remind reviewers of important elements of a good empirical evaluation, which they can take into account when carrying out their assessment. However, we emphasize that proper use of the checklist requires nuance. Just because a paper has every box checked doesn't mean it should be accepted. Conversely, a paper with one or two boxes unchecked may still merit acceptance. Even whether a box is checked or not may be subject to debate. The point is to organize a reviewer's thinking about an empirical evaluation to reduce the chances that an important aspect is overlooked. When a paper fails to check a box, it deserves some scrutiny in that category.

How did you determine which items to include?   The committee examined a sampling of papers from the last several years of ASPLOS, ICFP, OOPSLA, PLDI, and POPL, and considered those that contained some form of empirical evaluation. We also considered past efforts examining empirical work (Gernot Heiser's "Systems Benchmarking Crimes", the "Pragmatic Guide to Assessing Empirical Evaluations", and the "Evaluate Collaboratory"). Through regular discussions over several months, we identified common patterns and anti-patterns, which we grouped into the present checklist. Note that we explicitly did not intend for the checklist to be exhaustive; rather, it reflects what appears to us to be common in PL empirical evaluations.

Why did you organize the checklist as a series of categories, each with several examples?   The larger categories represent the general breadth of evaluations we saw, and the examples are intended to be helpful in being concrete, and common. For less common empirical evaluations, other examples may be relevant, even if not presented in the checklist explicitly. For example, for work studying human factors, the Adequate Data Analysis category might involve examples focusing on the use of statistical tests to relate outcomes in a control group to those in an experimental group. More on this kind of work below.

Why did you use checkboxes instead of something more nuanced, like a score?   The boxes next to each item are not intended to require a binary "yes/no" decision. In our own use of the list, we have often marked entries as partially filling a box (e.g., with a dash to indicate a "middle" value) or by coloring it in (e.g., red for egregious violation, green for pass, yellow for something in the middle).

What about human factors or other areas that require empirical evaluation?   PL research sometimes involves user studies, and these are different in character than, say, work that evaluates a new compiler optimization or test generation strategy. Because user studies are currently relatively infrequent in the papers we examined, we have not included them among the category examples. It may be that new, different examples are required for such studies, or that the present checklist will evolve to contain examples drawn from user studies. Nonetheless, the seven category items are broadly applicable and should be useful to authors of any empirical evaluation for a SIGPLAN conference.

How does the checklist relate to the artifact evaluation process?   Artifact evaluation typically occurs after reviewing a paper, to check that the claims and evidence given in the paper match reality, in the artifact. The checklist is meant to be used by reviewers while judging the paper, and by authors when carrying out their research and writing their paper.

How will this checklist evolve over time?   Our manifesto is: Usage should determine content. We welcome feedback from users of the checklist to indicate how frequently they use certain checklist items or how often papers reviewed adhere to them. We also welcome feedback pointing to papers that motivate the inclusion of new items. As the community increasingly adheres to the guidelines present in the checklist, the need for their inclusion may diminish. We also welcome feedback on presentation: please share points of confusion about individual items, so we can improve descriptions or organization.

   Feedback via: `http://www.sigplan.org/Resources/EmpiricalEvaluation/`