Empirical Evaluation Checklist (alpha version)

☐ Clearly Stated **Claims**

""

Explicit Claims

Claims must be explicit in order for the reader to assess whether the empirical evaluation supports the claim.

"precise analysis"

Appropriately-Scoped Claims

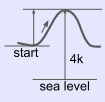
The truth of claims should follow from the evidence provided. Overclaiming is often the consequence of inadequate evidence, e.g., claiming 'works for all Java', but evaluating only a static subset or claiming 'works in real world', but evaluating only in (unrealistic) simulation.

"devs liked it"

Threats to Validity of Claims

A paper should state the most important threats to the validity of its claims, to place the scope of results in context. Stating no threats at all, or only tangential ones while omitting the more relevant ones, may mislead the reader to drawing too-strong conclusions.

☐ Suitable **Comparison**



"hiked 4k mountain"

Appropriate Baseline for Comparison

An empirical evaluation of a contribution that improves upon the state-of-the-art should evaluate that contribution against an appropriate baseline, such as the current best-of-breed competitor or a randomized baseline.

"finished before you"

Fair Comparison

Comparisons to a competing system should not unfairly disadvantage that system. For example, ideally, the compared systems would be compiled with the same compiler and optimization flags.

☐ Principled **Benchmark** Choice

"low sync overhead"

Appropriate Suite

Evaluations should be conducted using the appropriate established benchmarks where they exist. Established suites should be used in the designed-for context; for example, it would be wrong to use a single-threaded suite for studying parallel performance..

"we used TheBench"

Non-Standard Suite(s) Justified

Sometimes an established benchmark suite does not exist. A rationale should be provided for the selection of homegrown benchmarks or subsetting established benchmark suites.

"large speedup"

Applications, Not (Just) Kernels

A claim that a system benefits overall applications should be tested on such applications directly, and not only on micro-kernels (which can be useful and appropriate, in a broader evaluation)

☐ Adequate **Data Analysis**

"10 seconds"

Sufficient Number of Trials

In modern systems, which have non-deterministic performance, a small number of trials (e.g., a single time measurement) risks treating noise as signal.

"mean of 50 seconds"

Appropriate Summary Statistics

There are many summary statistics, and each presents an accurate view of a dataset only under appropriate circumstances. For example, 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.

"50 seconds"

Confidence Intervals

Reporting just a measure of central tendency (e.g., a mean or median) fails to capture the extent of any non-determinism. A measure of variability (e.g., variance, std deviation, quantiles) and/or confidence intervals help to understand the distribution of the data.

☐ Relevant **Metrics**

"energy consumed"

Direct or Appropriate Proxy Metric

If the most relevant evaluation metric is not (or cannot be) measured directly, the proxy metric used instead must be well justified. For example, a reduction in cache misses is not an appropriate proxy for actual end-to-end performance or energy consumption.

"devs were satisfied"

Measures All Important Effects

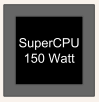
The costs and benefits of a technique may be multi-faceted. All facets should be considered, both costs and benefits. For example, compiler optimizations may speed up programs but at the cost of drastically increasing compile times.

☐ Appropriate and Clear **Experimental Design**

"sped up apache"

Sufficient Information to Repeat

Experiments should 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.



"for sensor net"

Reasonable Platform

The evaluation should be on a platform that can reasonably be said to match the claims. For example, a claim that relates to performance on mobile platforms should not have an evaluation performed exclusively on server.

"10 times faster"

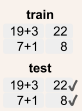
Explores Key Design Parameters

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.

"prompt treatment"

Open Loop in Workload Generator

Load generators for transaction-oriented systems should not be gated by the rate at which the system responds. Rather, the load generator should be 'open loop', generating work independent of the performance of the system under test.



"perfect"

Cross-Validation Where Needed

When a system aims to be general but was developed by training on or 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

"have up to 4 leaves"

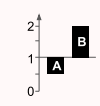
Comprehensive Summary Results

Appropriate statistics should be used to characterize the full range of results, not just the most favorable values, which may be outliers. For example, it is not appropriate to summarize speedups of 4%, 6%, 7%, and 49% as 'up to 49%'.

"B is much faster"

Axes Include Zero

A truncated graph (with an axis not including zero) can exaggerate the importance of a difference. While 'zooming' in to the interesting range of an axis can potentially aid exposition, there is a significant risk that this is misleading (especially if it is not immediately clear that the axis is truncated).



"Slowed A a little, sped up B lots"

Ratios Plotted Correctly

When ratios such as speedups and slowdowns are plotted, the size of the bars must be linearly/logarithmically proportional to the change. When shown on the same linear scale, results are visually distorted by 1/r, where r is the ratio. This misleading effect can be avoided either by using a log scale or by normalizing to the lowest (highest)

"9.36 s startup time"

Appropriate Level of Precision

The number of significant digits should reflect the precision of the experiment. Reporting improvements of '49.9%' when the experimental error is +/- 1% is an example of mis-stated precision, misleading the reviewer's understanding of the significance of the rest.

January 2018. E. D. Berger, S. M. Blackburn, M. Hauswirth, and M. Hicks for the ACM SIGPLAN EC