Evaluator guide of DSN 2025 Artifact Evaluation

This document is a guide through the artifact evaluation process for evaluators.

If you have general questions, please contact the artifact evaluation chairs. If you have a question about a specific artifact, see below for instructions on asking the authors.

1. Your Goal as a Reviewer

The goal of artifact evaluation is to help science by ensuring that published papers are accompanied by high-quality artifacts that can be reused and extended by others. Authors are incentivized to participate through the awarding of badges.

Keep in mind that artifact evaluation is a cooperative process. Artifacts that initially do not meet the requirements for badges can still get badges if the authors make necessary improvements in time, and evaluators should provide actionable feedback to enable this. Artifacts should only "miss" badges if there was not enough time to reasonably address the evaluators' concerns, or if the authors were unresponsive or unreasonable.

The papers under evaluation have already been accepted by the technical program committee, so **you do not need to evaluate their scientific soundness**. However, *if you believe you have found a technical flaw in a paper anyway, contact the artifact evaluation chairs*.

2. Timeline

The bidding deadline (April 2, AoE, hard deadline) is an important deadline, as it will allow the chairs to distribute artifacts in a way that maximizes evaluator expertise and interest. Bidding maximizes your chances to evaluate artifacts in domains you know about and are interested in.

The "kick the tires" period **(April 4 – 9, soft deadline)** is when evaluators go through the artifacts to ensure they will be able to properly evaluate them later. It is important to do this as soon as possible, so that authors have enough time to fix big issues if needed.

There is some time to agree on badges before the **final deadline (April 24, AoE, hard deadline)** to ensure that there is time for discussion of the artifacts that need it, and to leave time for any extra or late evaluations. Keep in mind that the final deadline for agreeing on badges is strict, as the rest of the conference depends on it.

The review is **single-blind**, i.e., you are able to see the author names but **the authors must not have the information of who reviewed their artifacts** so that you can be frank in your assessment. **Please do not communicate with authors through emails or any other ways that could reveal your identity.** We are still configuring the EasyChair system for author-reviewer communication.

3. Evaluation setups

You can evaluate artifacts on various setups, in order of preference:

- Your own machine, if the artifact supports it
    - Even if the artifact requires a particular OS or multiple machines, you may be able to run it locally via Docker or using virtual machines such as with VirtualBox
- Any servers with beefy/special hardware you may have access to, for artifacts that could benefit from this
- Research clouds, for artifacts that require more hardware than you have available; see the end of this document
- The artifact authors' machines, accessed via SSH or such, for artifacts that cannot run anywhere else due to hardware dependencies
- Commercial clouds such as AWS or Azure, but only if absolutely necessary and the authors are willing to pay for it; in that case, please agree with the authors on a protocol in which you agree to spawn and tear down machines to minimize unnecessary costs.

Note on anonymization: If you have to SSH to the authors' machines, make sure your public SSH key does not identify you (remove the user@host part at the end). If you believe you could be identified through other means, such as because of your IP address, contact the chairs.

4. Initial "kick the tires" phase

Once you have been assigned artifacts comes the initial "kick the tires" period. The goal of this period is **to quickly determine whether you have everything you need for a full review: the artifact itself, any necessary hardware or other dependencies, and a plan on how you will evaluate the artifact.** If that is not the case, you must discuss with your fellow evaluators and let the authors know of any problems as soon as possible, so that they have enough time to fix issues.

Double-check which badges the authors requested in their artifact submission; you do not need to evaluate the artifact for badges that were not requested (if you believe an artifact already meets the requirements for a badge the authors did not request, ask the authors; they may have forgotten to request that badge).

Carefully read the artifact documentation. In particular, **check the software and hardware dependencies to make sure you have all you need.** You are allowed to use your own judgment when making decisions, for instance to evaluate reasons why some artifacts may not be able to reproduce everything their paper contains.

Discuss with your fellow evaluators, in reviewer discussion comments, so that you all agree on:

- Whether you have everything you need to do the evaluation, and if not, what is missing including:
    - Access to the necessary hardware owned by you, by the authors or academic clouds
    - For artifacts requesting the “reviewed” badge, documentation and full source code as mentioned in the checklist (See Section 7), and whether the code compiles
    - For artifacts requesting the “reproducible” badge, scripts to run the experiments and generate figures as mentioned in the checklist (See Section 7)
- A plan on how you will evaluate the artifact during the review period:
    - List which results will be reproduced to verify a claim
    - Time frames of when experiments will be run in case hardware is shared

5. Reviewing artifacts

For each artifact you are assigned to, you will produce one review explaining which badges you believe should be awarded and why or why not. You will work with the authors to produce your review, as this is a cooperative process. Authors are a resource you can use, exclusively through EasyChair, if you have trouble with an artifact or if you need more details about specific portions of an artifact. There is an example review at Section 6.

First, Section 7 gives a checklist: artifacts that meet these requirements should get the corresponding badges, while artifacts that do not should either justify why or not get the badges. If an artifact does not satisfy a checklist but the authors provide a good reason as to why they should get the badge anyway, use your judgment based on the definitions of the badges. Remember that the Artifacts Reviewed and Reproducible badges require not only running the code but also auditing it to ensure that **(for Artifacts Reviewed) the code is documented and understandable**, and **(for Reproducible) the code actually does what the paper states it does**. Merely reproducing similar output as the paper, such as performance metrics, is not enough, the artifact must actually do what it claims to do. **You are not expected to understand every single line of code, but you should be confident that the artifact overall matches the paper’s description.**

Most of your time should be spent auditing artifacts, not debugging them. If you run into issues such as missing dependencies, try to quickly work around them, such as by finding the right package containing the dependency for your operating system and letting the authors know they have to fix their instructions. However, it is the authors’ responsibility to make their artifacts work, not yours. You do not need to spend hours trying to debug and fix complex issues; if you encounter a non-trivial error, first ask your fellow evaluators if they encountered it too or if they know how to fix it, then ask the authors to fix it.

It is acceptable to deny badges if artifacts require unreasonable effort, especially if such effort could be avoided through automation. For instance, if reproducing a claim requires 50 points of data, and the artifact requires you to manually edit 5 config files then run 4 commands on 3 machines for each data point, you do not need to actually perform hundreds of manual steps; instead, ask the authors to automate this, or even write a script yourself if you have the time that you can then share with the authors.

Once you are finished evaluating an artifact, fill in the review form and submit it. Your review must explain in detail why the artifact should or should not get each of the badges that the authors requested. You can also include additional suggestions for the authors to improve their artifacts if you have any.

Remember that the artifact evaluation process is cooperative, not adversarial. Give authors a chance to fix issues by discussing with authors before deciding that their artifact should not get a badge. However, you are allowed to edit your review until the deadline, so if authors are being unresponsive or unreasonable, feel free to submit an early review denying a badge and listing the actionable steps the authors should take to get the badge.

6. Example review

We provide here an example review for a fictitious artifact/paper. This review started by copy-pasting each point from the badge checklists, then modifying the text to suit the artifact and starting each point with one of ✔ (= yes), ❌ (= no), or ⚠️ (= yes, but has issues). For the "Reproducible" badges, if results differ from the original in any way it's good to explain how, even if the badge should be awarded.

===Available===

- ✔ The artifact is available on a public Figshare repository (or Zenodo)
- ✔ The artifact has a "read me" file with a reference to the paper. The artifact is consistent and complete with respect to the paper
- ✔ The artifact provides sufficient user documentation with command-line syntax
- ✔ The artifact can potentially be used to support reproducibility of the paper
- ⚠️ The artifact does not have a license that allows use for comparison purposes; this is not necessary but it would be good to have

I suggest awarding the badge.

=== Reviewed===

- ❌ The artifact 's "read me" file is missing some information:
    - ✔ It has a description

    - ✔ It has compilation and running instructions
    - ✔ It has usage instructions to run experiments
    - ❌ It does not have a list of supported environments
    - ❌ It does not have configuration instructions to select the client and server IPs
- ❌ It does not have instructions for a “minimal working example”, only for full experiments on 12 machines
- ✔ The code is well documented at both the module and class level, good job!
- ⚠ The artifact contains all major parts described in the paper; it would be good if it included the extra experiments mentioned in the paper’s Limitations section as well, but this is not mandatory

I hope authors can fix the issues mentioned above so that the Artifact Reviewed badge can be awarded.

===Reproducible===

- ✔ The artifact has a “read me” file that documents:
    - ✔ The claims of the paper it can reproduce and the detailed instructions to reproduce them
    - ✔ The exact environment the authors used
    - ✔ The exact commands to run to reproduce each claim from the paper
    - ⚠ The time used per claim, but not the disk space which would be nice to indicate since it is multiple GBs
    - ✔ The scripts to reproduce claims are very well documented and correspond to what the paper states

I considered the 3 claims, except for claim #3 in which I used a different configuration file after our discussion with the authors. All experiments were carried out on CloudLab using the authors’ profile.

- ✔ I am able to run the analysis automatically, with a reasonable effort and time/resource requirements

- ✔ I am able to fully reproduce the results of claim 1 and claim 3. For claim 2, I obtained 4400 ops/s for the artifact and 3500 ops/s for the baseline, which is a bit lower for the artifact than what the paper claims. However, the key part of the claim is that the artifact is at least as fast as the baseline, not an absolute performance number, so I believe this is fine.

I suggest awarding the badge.

7. Checklist for Evaluation

For the "Available" badges:

- Is the artifact publicly available through an open-access repository (Zenodo or Figshare)?
- Is the artifact consistent and complete with respect to the paper?
- Does it provide sufficient user documentation (e.g., command-line syntax)?
- Can it potentially be used to support reproducibility of the paper (even if you could not run the artifact)?
- Does it include a license that allows researchers to reuse and extend the artifact (e.g., for comparison purposes in a future paper)? Creative Commons licenses are a typical choice for open data.

For the "Reviewed" badges:

- Does the artifact include enough documentation about configuration and installation (e.g., on external dependencies, supported environments)?
- Does it include instructions for a "minimum working example", and could you run it?
- Does the artifact include documentation about its internals (e.g., organization of modules and folders, code comments for explaining non-obvious code) that is understandable for other researchers?

For the "Reproducible" badges:

- Does the documentation of the artifact explain which claims of the paper it can reproduce, and how to reproduce them?
- Can you run the analysis automatically, with a reasonable effort and time/resource requirements?
- Do the results of the execution support the claims of the paper?

Additional suggestions:

- You are allowed to provide your artifact as a virtual machine. Even in that case, you should still provide source code and scripts that were used to build the virtual machine.
- Please minimize the number of dependencies and the amount of hardware resources needed to run the artifact.
- Please provide clear step-by-step instructions to install and run the artifacts. Remember to test them on a clear environment!
- When providing instructions to users and reviewers, please provide the expected outputs (or any other side effect) of these instructions, and the estimated amount of human and compute time.

8. Information about academic clouds

All reviewers must understand the infrastructure requirements of their artifacts, to determine whether their own infrastructure is enough (if they have access to comparable hardware), or, if it is not, whether they can use an academic cloud offering or, as a last resort, whether they need the authors to provide access to hardware.

Chameleon Cloud

Chameleon is a large-scale, deeply-reconfigurable (bare metal) testbed for systems research based on enhanced OpenStack. Users get full control of the software stack including root privileges, kernel customization, console access, as well as the ability to experiment with software-defined networking and diverse set of hardware consisting of different types of powerful GPUs, FPGAs, storage hierarchy nodes with a mix of HDDs, SDDs, and NVMe, high-bandwidth I/O storage, SDN-enabled networking hardware, and Infiniband.

CloudLab

CloudLab is a facility that provides bare-metal access to about 2,000 servers at 5 locations. Users have full 'root' access, and most infrastructure is not shared, making it a good choice for experiments that require low-level access to hardware, and/or a non-shared environment for repeatable performance. Processors available on CloudLab include many generations of Intel, AMD, ARM, and IBM POWER architectures, as well as several varieties of GPUs. Networks include both Ethernet (up to 100Gbit) and Inifiband. The full list of CloudLab hardware is available in its manual.

*This document is edited based on the EuroSys'22 Evaluator guide (https://sysartifacts.github.io/eurosys2022/guide).