Reproducibility Report for ACM SIGMOD 2023 Paper: "INEv: In-Network Evaluation for Event Stream Processing"

SUPAWIT CHOCKCHOWWAT, University of Illinois Urbana-Champaign, USA
SEBASTIAN SCHMIDL, Hasso Plattner Institute, University of Potsdam, Germany
DENNIS TREDER-TSCHECHLOV, Parallel and Distributed Systems, University of Stuttgart, Germany

We successfully replicated the key performance outcomes presented in the experimental section of the original paper. The reproducibility package contains the primary artifact accompanied by experiment and visualization scripts that produce comparable figures.

1 INTRODUCTION

This reproducibility report concerns the paper "INEv: In-Network Evaluation for Event Stream Processing" [1] that proposes a framework to construct a distributed in-network evaluation of complex event processing (CEP) workloads to reduce transmission costs. The authors of the paper are Samira Akili, Steven Purtzel, and Matthias Weidlich, all from Humboldt-Universität zu Berlin, Germany. The authors have uploaded a reproduction package to GitHub, which qualifies the paper for the *Artifacts Available* badge. Additionally, they provide detailed instructions to execute their many experiments and to mostly automatically generate the experiments' result table and figures, meeting the criteria for the *Artifacts Evaluated* badge. Using the submitted reproduction package, we can confirm that the proposed INEv graphs outperform existing solutions and have generally very low transmission ratios, demonstrating the reproducibility of the results and earning the paper the *Results Reproduced* badge.

2 SUBMISSION

The submission contains the primary paper and its reproduction package hosted on a GitHub repository (https://github.com/samieze/INEv). The experiments are categorized into two groups: local experiments and Pi cluster experiments. Despite collected into tar-archives limiting the browsability directly from GitHub, the package is sufficient to reproduce all results and consists of:

- Detailed instructions (`INEv_reproducibility.pdf`) describing two ways (with and without using Docker) to reproduce the local experiments.
- The open-source implementation of a prototypical distributed CEP engine for INEv graphs at https://github.com/samieze/INEv/tree/main/INEv.
- An archive containing all datasets, scripts, and source code to execute the local experiments. The archive contains the script `all_figures.sh` that starts all experiments and builds all figures for the local experiments.
- Instructions on how to perform the local experiments using the provided Docker image (`README-localexperiments-docker.md`).
- A second archive containing all datasets, scripts, and source code to execute the Pi cluster experiments. The experiments are designed to be executed on a distributed cluster of ten Raspberry Pis from Mythic Beasts (a hosting provider, https://www.mythic-beasts.com).

The review process led to multiple refinements of the reproduction package to remove bugs in the experiments scripts and to improve the usability.

3 HARDWARE AND SOFTWARE ENVIRONMENT

In Table 1, we have summarized and compared the experimental environments used in our reproducibility study with those described in the original paper. Setting up INEv takes one to two hours. Running the main experiment script takes about two to five days.

Table 1. Hardware & Software environments

	Paper	Reviewer #1	Reviewer #2	Reviewer #3
CPU	Intel Xeon E7-4880	AMD EPYC 7763	Intel Xeon E5-2630 v4	2× AMD EPYC 7552 (NUMA)
Cores	60	256	20	192
GHz	2.5	3.3	2.2	3.3
RAM	1 TB	1 TB	64 GB	1 TB
System	Ubuntu 18.04 or later	Ubuntu 20.04.6	Ubuntu 20.04.6	Ubuntu 20.04.6
Java	OpenJDK 11	OpenJDK 11	OpenJDK 11	OpenJDK 8
Python	3.8	3.8	3.8	3.8

The Pi cluster experiments need to be executed on a distributed cluster of exactly 10 Raspberry Pis. One reviewer was able to reproduce the cluster experiments using the same hardware as the original authors: 10x Raspberry Pi 4 1.5 GHz 4 GB RAM running Raspbian Buster 10 32 bit, in a 1 GBit Ethernet LAN.

4 REPRODUCIBILITY EVALUATION

4.1 Process

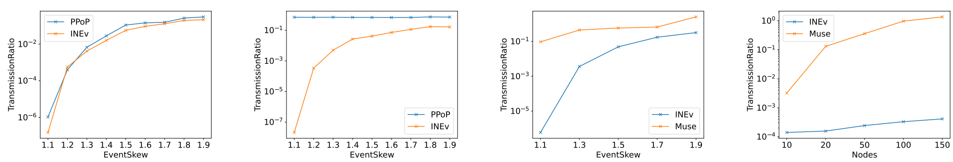
We followed the instructions using the Docker image to execute the "local experiment". There are only three steps to this process: (i) downloading the Docker image `ghcr.io/samieze/repro23` and running it, (ii) executing the fully automatic `all_figures.sh`-script within the Docker container, and (iii) copying the result figures from the Docker container to the host's file system.

Executing the Pi cluster experiments requires multiple manual steps. The authors provide automatic scripts to provision the required Raspberry Pis using Mythic Beasts. However, when using a different provider or local hardware, preparing the hardware requires manual work. Once the hardware is set up, there are seven steps to execute three different experiments (2x INEv and 1x Flink) and build the figures.

4.2 Results

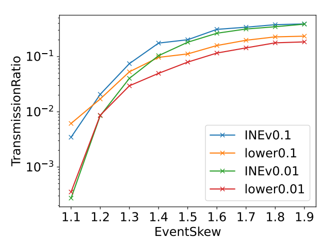
We have reproduced most of the results, including 8 out of 9 figures completely, having 24 plots in total. Figures 1, 2, 3, 4, 5, 6, 7, 8, 9 display our reproduced results, which support all the key findings. Although [1]'s Table 3 is missing, we have attempted to reproducing by pivoting the generated `table.csv`, resulting in Figure 10.

There are some differences in measurement due to differences in environment characteristics and because all experiments are subjected to some notion of randomness, but overall relationships remain consistent. Notably, for Figure 3, we were only able to recreate two out of four plots and the first plot misses the results for two queries. This figure summarizes the results of the Pi cluster experiments, which were difficult to configure and execute. The obtained latency results for queries Q1 and Q2, though, clearly demonstrate the improvement of INEv over Flink, already confirming the claim in the paper. QWL results such as those in Figure 4 are noisy, roughly showing a similar trend, but they may need a mitigation to reduce the noise level. We also observe lower transmission costs at 1.1 EventSkew in Figure 5 than those in [1], suggesting further investigation and discussion of such effect in the main paper. In addition, we observe a mixed result in invalidity between Projections and MNPlacements in Figure 8. Figure 9 contains a higher computation time outlier in higher query length and missing points in the lower-right corner of projections vs. computation time. Minor visualization errors include swapped line colors in Figure 1 and mismatched x-axis labels in Figure 9. The potentially reproduced Figure 10 displays the same relative patterns as that in [1], neglecting the tschedifferences in absolute numbers. Overall, we can faithfully confirm all the key insights in the original paper despite the minor measurement differences.



(a) (b) (c) (d)

Fig. 1. Reproduction of Figure 4 in [1].



(a)

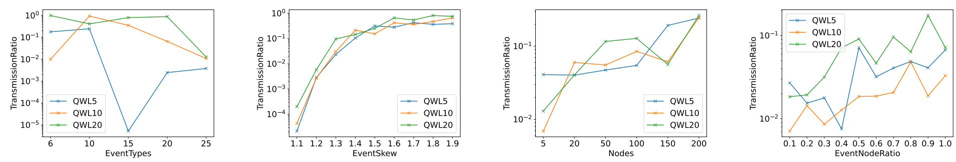
Fig. 2. Reproduction of Figure 5 in [1].

(a) (b) (c) (d)

Fig. 3. Reproduction of Figure 6 in [1].

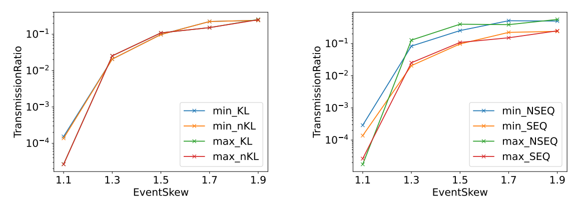
5 SUMMARY

All experimental results of the original paper are reproducible to support its key findings, and suggestions to improve the artifacts were considered. We thank the authors for their efforts toward delivering and maintaining the reproducibility artifact.



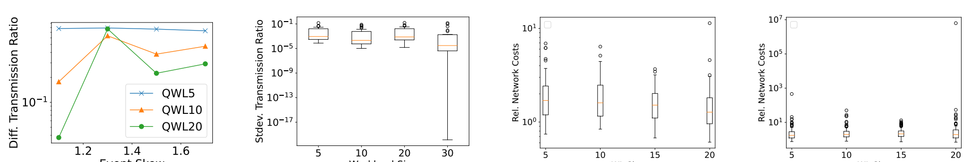
(a) (b) (c) (d)

Fig. 4. Reproduction of Figure 7 in [1].



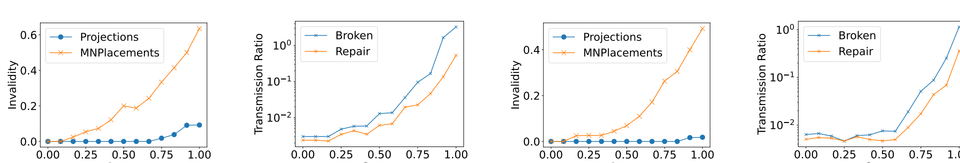
(a) (b)

Fig. 5. Reproduction of Figure 8 in [1].



(a) (b) (c) (d)

Fig. 6. Reproduction of Figure 9 in [1].

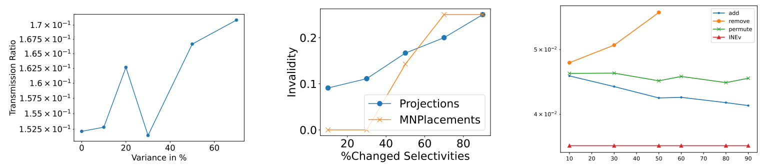


(a) (b) (c) (d)

Fig. 7. Reproduction of Figure 10 in [1].

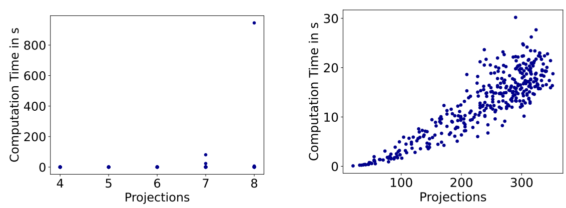
REFERENCES

[1] Samira Akili, Steven Purtzel, and Matthias Weidlich. 2023. INEv: In-Network Evaluation for Event Stream Processing. *Proc. ACM Manag. Data* 1, 1, Article 101 (may 2023), 26 pages. https://doi.org/10.1145/3588955



(a) (b) (c)

Fig. 8. Reproduction of Figure 11 in [1].



(a) (b)

Fig. 9. Reproduction of Figure 12 in [1].

		citibike				google			
		1	2	3	4	1	2	3	4
5	inev	0.0008	0.0086	0.1431	0.0000	0.0001	0.0000	0.0004	0.0001
	muse	0.4330	0.0140	1.0023	0.0000	0.0021	0.0000	0.0019	0.3509
	PPoP	0.8333	0.8333	0.8333	0.8333	1.0004	1.0000	1.0000	1.0008
10	inev	0.0010	0.0111	0.0051	0.0105	0.0001	0.0008	0.0000	0.0002
	muse	0.0031	1.4700	1.4901	0.0000	0.6456	0.4946	0.0000	0.0007
	PPoP	0.8095	0.8095	0.8095	0.8095	0.8097	0.8095	0.8095	0.8097
20	inev	0.0011	0.0000	0.0059	0.0000	0.0002	0.0016	0.0062	0.0000
	muse	0.0025	1.4615	1.4531	0.0000	0.4652	0.0000	0.0006	0.0000
	PPoP	0.5465	0.6944	0.5467	0.6944	0.8772	0.8772	0.8772	0.8772
30	inev	0.0377	0.0110	0.0053	0.0105	0.0001	0.0968	0.0059	0.0000
	muse	0.0038	1.5470	0.0000	1.2199	0.9821	0.8735	0.0008	0.0012
	PPoP	0.4062	0.7069	0.4064	0.7212	0.5850	0.6483	0.5280	0.5850
50	inev	0.0015	0.0153	0.2265	0.0142	0.0007	0.0021	0.0000	0.0003
	muse	0.0032	0.9711	1.1044	0.8428	0.4093	0.3718	0.0019	0.0007
	PPoP	0.2379	0.4326	0.2382	0.2954	0.3532	0.5355	0.2333	0.5355
10	Flink	1.4258	1.4196	1.4348	1.4263	1.4541	1.4136	1.8073	1.3242

Fig. 10. Potential reproduction of Table 3 in [1].