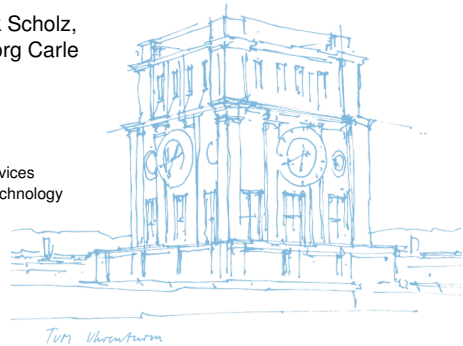
Reproducible by Design: Network Experiments with pos

Sebastian Gallenmüller, Dominik Scholz,
Henning Stubbe, Eric Hauser, Georg Carle

Chair of Network Architectures and Services
School of Computation, Information, and Technology
Technical University of Munich



Reproducibility

Reproducible experiments

- Everyone agrees that reproducible research is important
- The best solution our community has come up so far:

Reproducibility

Reproducible experiments

- Everyone agrees that reproducible research is important
- The best solution our community has come up so far:

Reproducibility

Problems with reproducibility

- Two workshops at SIGCOMM conference dedicated to reproducible research:
  - SIGCOMM'03: MoMeTools workshop
  - SIGCOMM'17: Reproducibility workshop
  - Problems remained the same over 14 years

Best solution so far . . .

- Artifact Evaluation Committees & Reproducibility Badges
- Problems:
  - High effort
  - Potentially low robustness (CCR Apr. '20[1])

ACM's badges awarded by the Artifact Evaluation Committee

[1] [1] N. Zilberman, "An Artifact Evaluation of NDP", Comput. Commun. Rev., vol. 50, no. 2, pp. 32–36, 2020

Reproducibility-as-a-Service

What is reproducibility?

- 3-stage process according to ACM[2]:
  1. Repeatability: Same team executes experiment using same setup
  2. Reproducibility: Different team executes experiment using same setup
  3. Replicability: Different team executes experiment using different setup
- Our testbed-driven approach mainly targets the experimental setup

→ Focus our effort on repeatability and reproducibility

→ Replicability requires additional effort by others

[2] [2] ACM, Artifact Review and Badging Ver. 1.1, 2020. [Online]. Available: https://www.acm.org/publications/policies/artifact-review-and-badging-current

Reproducibility-as-a-Service

How can we limit effort spent on reproducibility?

- Reduce amount of work for artifact evaluators or other researchers
- Make reproducibility part of experiment design
- → Automate entire experiment (setup, execution, evaluation)

How can we create robust, reproducible experiments?

- Document all relevant parameters for experiments
- Automate the documentation of experiments
- → Well-structured experiment workflow serving as documentation

The Plain Orchestrating Service (pos)

Our solution to create reproducible research

1. Create a testbed management system
2. Create a well-defined experiment workflow

The Plain Orchestrating Service (pos)

Our solution to create reproducible research

1. Create a testbed management system
2. Create a well-defined experiment workflow

Achieving Repeatability

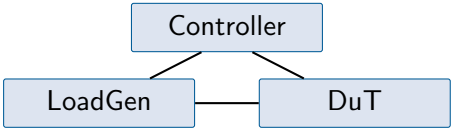
- Automation
- Live images
  - Researchers **must** automate configuration
  - No residual state between reboots

➔ Experiments become repeatable

Achieving Reproducibility

- Providing access to experiment infrastructure
- Other researchers can easily (re-)run experiment

➔ Experiments become reproducible

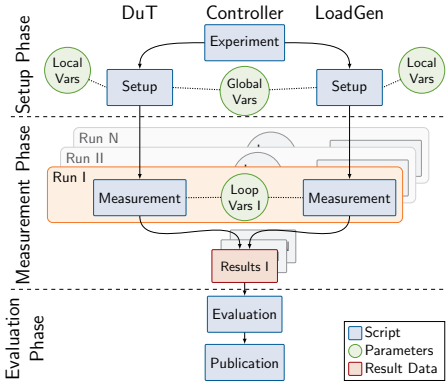


Minimal pos experiment topology

pos' Methodology

Setup phase

- Controller manages experiment workflow
- Initialization of experiment nodes
  - Reboot experiment nodes
  - Live Linux images via network boot
  - Recover from possible error states
  - Supported interfaces:
    - IPMI
    - Intel management engine
    - Network-controlled power plugs
- Configuration of experiment nodes:
  - Prepare system for experiments (e.g., install software, configure addresses)
  - Configuration management tools are supported, e.g., Ansible, Chef, etc.
  - Install testbed utility scripts (e.g., synchronization tool)
  - Global / local variables (vars) help parametrize configuration
- Configuration and initialization are fully automated



pos workflow

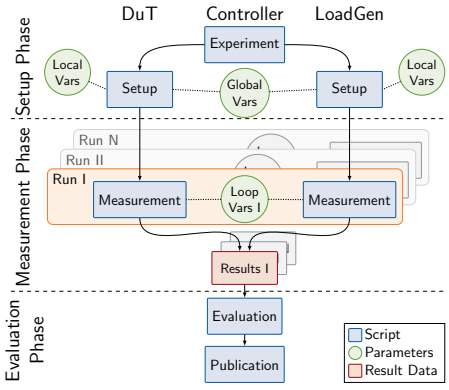
pos' Methodology

Measurement phase

- Performing the actual experiment
- Repeated execution of measurement script
- Loop variables parameterize each measurement run
  - For instance, different packet rates and different packet sizes
  - Experiment results of each run is associated to a specific set of loop vars

Loop vars example

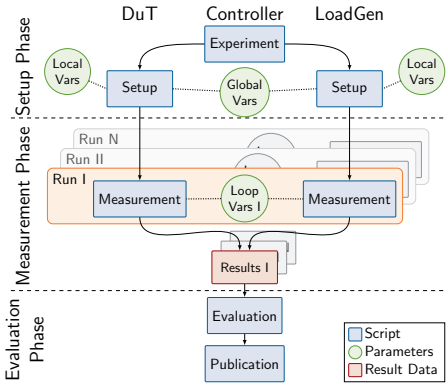
- pos calculates the cross product for the given loop vars:
  - `pkt_rate: [1000, 5000]`
    `pkt_sizes: [64, 1500]`
- Measurement script is executed for each tuple in the cross product:
  - `Run1: {pkt_rate: 1000, pkt_size: 64}`
  - `Run2: {pkt_rate: 1000, pkt_size: 1500}`
  - `Run3: {pkt_rate: 5000, pkt_size: 64}`
  - `...`



pos workflow

Evaluation phase

- Result file upload from experiment nodes to the controller:
  - pos tags all result files with the specific measurement run
  - ➔ `result_run1.csv`
  - Loop vars can be considered as metadata for the result
  - ➔ `Run1: {pkt_rate: 1000, pkt_size: 64}`
- Collected results / loop vars for experiment evaluation
  - Plotting tool evaluates loop variables and measurement files
  - Loop vars are used for automated plotting, e.g., aggregating over `pkt_rate`
- Well-defined format for pos scripts, loop vars, and results:
  - Well-defined format allows automated evaluation
  - Automated preparation of experiment artifacts (git repository, website)
  - e.g., https://gallenmu.github.io/pos-artifacts/



pos workflow

Testbed-as-a-Service (TaaS)

Using pos

- Virtualized version of our testbed[3] available as a service for other researchers
- Affordable single-server testbed with low complexity
- Realistic performance using hardware acceleration (SR-IOV)
- Future use cases:
  - Stand-in replacement for a real (future) testbed
  - Development, training, or teaching facility

Try out toast for yourself . . .

- https://testtestbed.net.in.tum.de

Server for virtualized testbed

[3] [3] S. Gallenmüller, E. Hauser, and G. Carle, “Prototyping Prototyping Facilities: Developing and Bootstrapping Testbeds”, in IFIP Networking 2022 WKSHPS SLICES, Catania, Italy, Jun. 2022

Conclusion

- pos[4] is ...
  - a testbed orchestration service, and
  - an experiment methodology.
- Methodology makes experiments ...
  - repeatable as everything is automated,
  - reproducible as others can re-run the automated pos experiments, and
  - easier to replicate as the experiment scripts document experiments.

→ pos reduces the effort to create reproducible experiments.

→ pos complements the ACM awards—it does not replace them.

[4] [4] S. Gallenmüller, D. Scholz, H. Stubbe, et al., "The pos framework: A methodology and toolchain for reproducible network experiments", in CoNEXT '21, Virtual Event, Munich, Germany, December 7 - 10, 2021, ACM, 2021, pp. 259–266. DOI: 10.1145/3485983.3494841

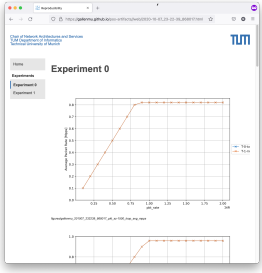
Conclusion

- pos[4] is ...
  - a testbed orchestration service, and
  - an experiment methodology.
- Methodology makes experiments ...
  - repeatable as everything is automated,
  - reproducible as others can re-run the automated pos experiments, and
  - easier to replicate as the experiment scripts document experiments.

➔ pos reduces the effort to create reproducible experiments.

➔ pos complements the ACM awards—it does not replace them.

- Resources are publicly available:
  - VM: https://testtestbed.net.in.tum.de
  - Repository: https://github.com/gallenmu/pos-artifacts
  - Website: https://gallenmu.github.io/pos-artifacts



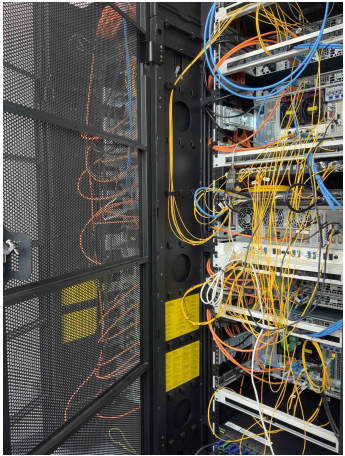
Website generated by pos experiment workflow

[4] S. Gallenmüller, D. Scholz, H. Stubbe, et al., "The pos framework: A methodology and toolchain for reproducible network experiments", in CoNEXT '21, Virtual Event, Munich, Germany, December 7 - 10, 2021, ACM, 2021, pp. 259–266. DOI: 10.1145/3485983.3494841

SLICES-RI

SLICES RI (Research Infrastructures)

- Goal: provide advanced computing, storage, and high-speed network infrastructure
- The pos framework and workflow is part of our contribution to slices
- How to get access:
  - We provide access to a virtual instance of pos: https://testtestbed.net.in.tum.de
  - Experiments can be developed and executed in the virtual instance of pos
  - Reproducible pos experiments can be automatically re-run on any pos testbeds
  - Experiments can be handed in to be run on real hardware
- The pos testbeds are part of the SLICES Open Call



Bibliography

[1] N. Zilberman, “An Artifact Evaluation of NDP,” Comput. Commun. Rev., Jg. 50, Nr. 2, S. 32–36, 2020.

[2] ACM, Artifact Review and Badging Ver. 1.1, 2020. Adresse: https://www.acm.org/publications/policies/artifact-review-and-badging-current.

[3] S. Gallenmüller, E. Hauser und G. Carle, “Prototyping Prototyping Facilities: Developing and Bootstrapping Testbeds,” in IFIP Networking 2022 WKSHPS SLICES, Catania, Italy, Juni 2022.

[4] S. Gallenmüller, D. Scholz, H. Stubbe und G. Carle, “The pos framework: a methodology and toolchain for reproducible network experiments,” in CoNEXT '21, Virtual Event, Munich, Germany, December 7 - 10, 2021, ACM, 2021, S. 259–266. DOI: 10.1145/3485983.3494841.