Evaluation & Data Management

沈斯杰

2021.12.15

Outline

- **Systems Benchmarking Crimes**
- **Data presentation (GNUPlot)**
- **Data management**
  - Artifact Evaluation

Outline

- **Systems Benchmarking Crimes**
- **Data presentation (GNUPlot)**
- **Data management**
  - Artifact Evaluation

USENIX ATC '22 Submission Instructions

Process Overview

A good submission will typically: motivate a significant problem; propose a practical solution or approach that makes sense; demonstrate the pros and cons of the latter using sound experimental and statistical evaluation methods; disclose what has and has not been implemented; articulate the new contributions beyond previous work; and refrain from over-claiming, focusing the abstract and introduction sections primarily on the *difference* between the new proposal and what is already available. Submissions will be judged on relevance, novelty, technical merit, correctness, and clarity. An idea or design that the PC committee deems flawed can be grounds for rejection.

Submissions are required to avoid committing benchmarking crimes.

Systems Benchmarking Crimes

- **Gernot Heiser**

Contents

https://www.cse.unsw.edu.au/~gernot/benchmarking-crimes.html
https://arxiv.org/pdf/1801.02381.pdf

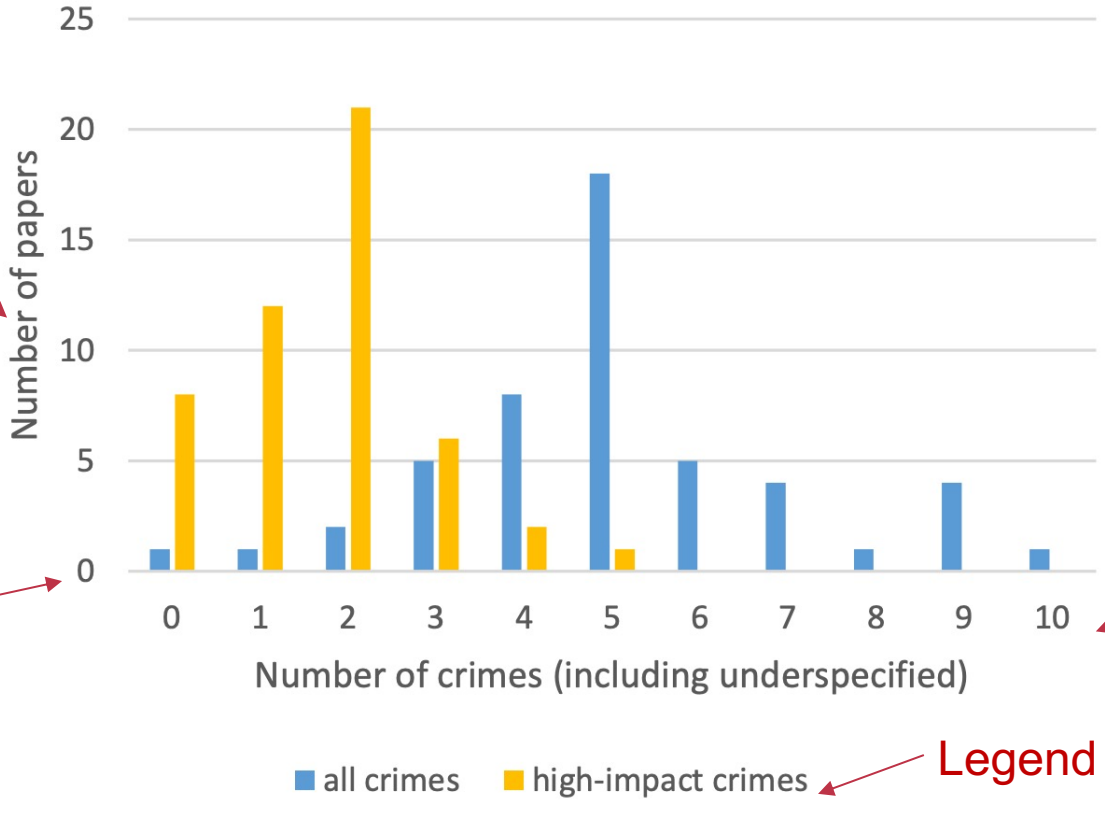
Outline

- **Systems Benchmarking Crimes**
- **Data presentation (GNUPlot)**
- **Data management**
  - Artifact Evaluation

Data presentation: diagram

- **Figures**
  - Vivid
  - Present data clearly
- **Tables**
  - Collection & statistics

How to read a figure?



Y-axis (label&unit)

Orgin

X-axis (label&unit)

Legend

Title Description, emphasized

Fig. 1. Histogram of number of crimes per paper

How to draw a figure?

- **No cheating**
  - Clear information about coordinate (e.g., logarithmic coordinates)
  - Y-axis: start from 0 as possible
  - Unit
- **Tools**
  - Excel, GNUPlot, Python+matplotlib, …

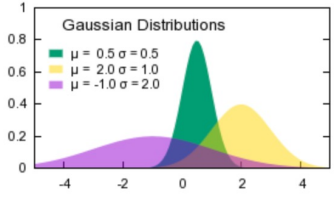
Report by figures.

Common figures

- Histogram
- Line chart
  - Scalability
  - Timeline
  - Throughput-latency
- Box plot
  - Max, min, median

GNUPLOT

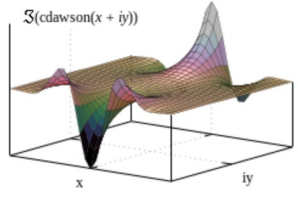
- Command-line driven graphing utility



gnuplot homepage

FAQ
Documentation
Demos
Download

Contributed scripts
External Links
Tutorials and guides



Gnuplot is a portable command–line driven graphing utility for Linux, OS/2, MS Windows, OSX, VMS, and many other platforms. The source code is copyrighted but freely distributed (i.e., you don't have to pay for it). It was originally created to allow scientists and students to visualize mathematical functions and data interactively, but has grown to support many non–interactive uses such as web scripting. It is also used as a plotting engine by third–party applications like Octave. Gnuplot has been supported and under active development since 1986.

Gnuplot supports many different types of 2D and 3D plots

Here is a Gallery of demos.

Gnuplot supports many different types of output

interactive screen display:	cross–platform (Qt, wxWidgets, x11) or system–specific (MS Windows, OS/2)
direct output to file:	postscript (including eps), pdf, png, gif, jpeg, LaTeX, metafont, emf, svg, ...
mouseable web display formats:	HTML5, svg

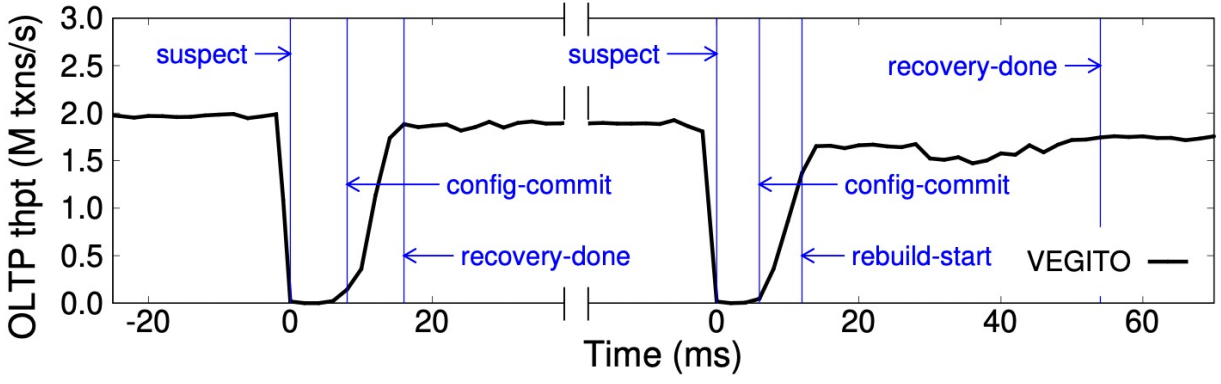
http://www.gnuplot.info/

Learn from samples

- **Our gitlab**
- **Gnuplot demo**
  - http://gnuplot.sourceforge.net/demo_5.4/

Problem of Gnuplot

- **Compatibility**
  - Version, architecture
- **Special labels**
  - Hand-written



Outline

- **Systems Benchmarking Crimes**
- **Data presentation (GNUPlot)**
- **Data management**
  - Artifact Evaluation

Artifact Evaluation

OSDI '21 Call for Artifacts

Overview

A scientific paper consists of a constellation of artifacts that extend beyond the document itself: software, hardware, evaluation data and documentation, raw survey results, mechanized proofs, models, test suites, benchmarks, and so on. In some cases, the quality of these artifacts is as important as that of the document itself. Last year, 70% of accepted OSDI papers participated in the artifact evaluation process. Based on last year's success, OSDI '21 will continue to run an optional artifact evaluation process.

The artifact evaluation process will consider the availability and functionality of artifacts associated with their corresponding papers, along with the reproducibility of the paper's key results and claims with these artifacts. Artifact evaluation is single-blind. Artifacts will be held in confidence by the evaluation committee.

All (conditionally) accepted OSDI papers are encouraged to participate in artifact evaluation. Because the time between paper acceptance and artifact submission is short, we strongly encourage authors to start preparing their artifacts for evaluation while their papers are still under consideration by the OSDI Program Committee. See the Submitting an Artifact section for details on the submission process.

Questions about the process can be directed to osdi21aec@usenix.org.

Badgets


ARTIFACT
EVALUATED
usenix
ASSOCIATION
AVAILABLE


ARTIFACT
EVALUATED
usenix
ASSOCIATION
FUNCTIONAL


ARTIFACT
EVALUATED
usenix
ASSOCIATION
REPRODUCED


Artifacts Evaluated
acm
Functional V1.1


Artifacts Available
acm
V1.1


Results Reproduced
acm
V1.1

Prepare for AE

- **Environment**
  - Hardware dependence (run on different architectures, special hardwares)
  - Software dependence (library)
  - Recommendation: docker

Prepare for AE

- **Original data**
  - Log the original data of main experiments
  - Especially for baseline systems

A good example

- AIFM @ OSDI’20

https://github.com/AIFM-sys/AIFM/