LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Maximilian Höb

eBPF-based Performance Fingerprint of containerized HPC applications

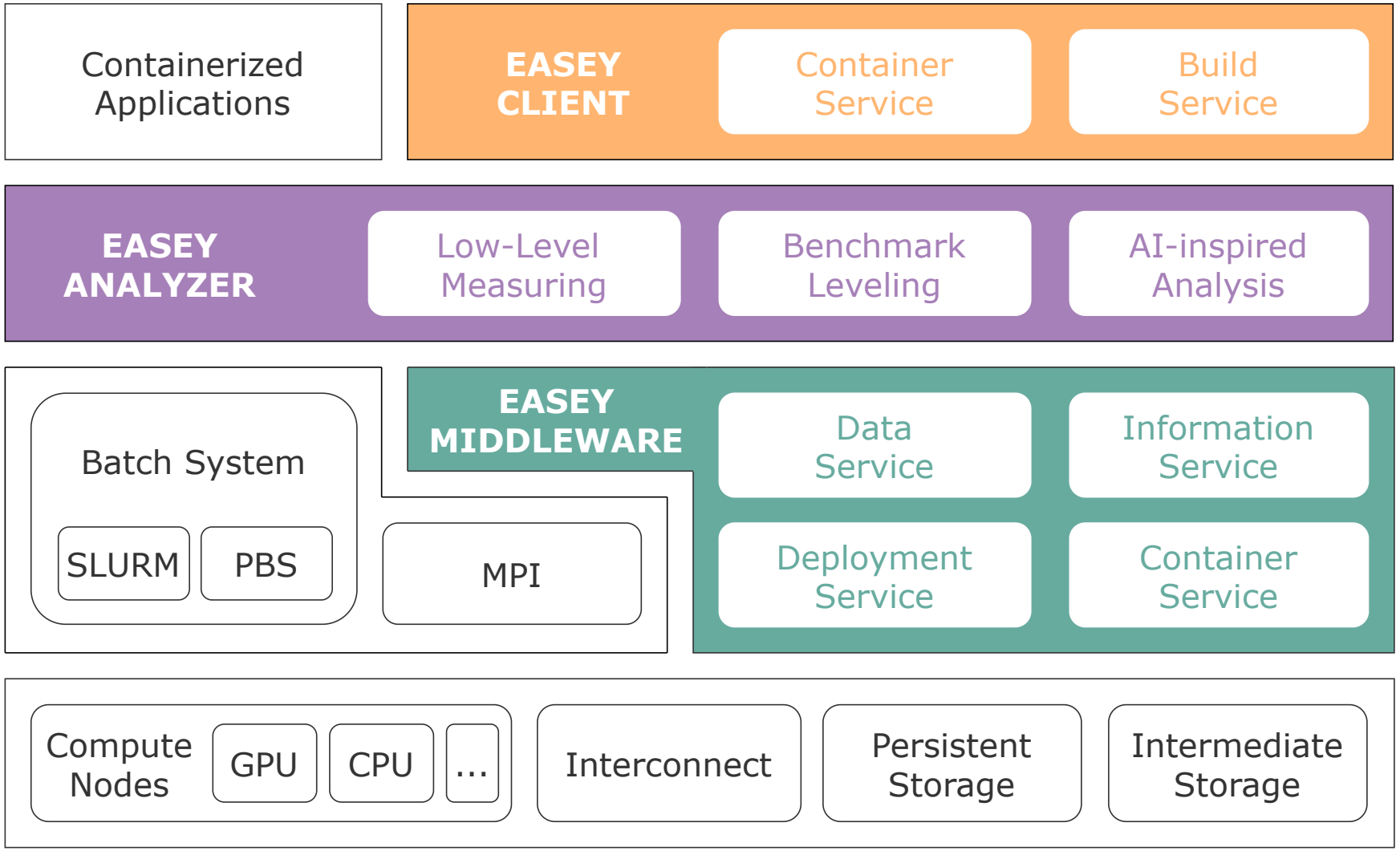
hoeb@mnm-team.org

CANOPIE-HPC @ SC2023

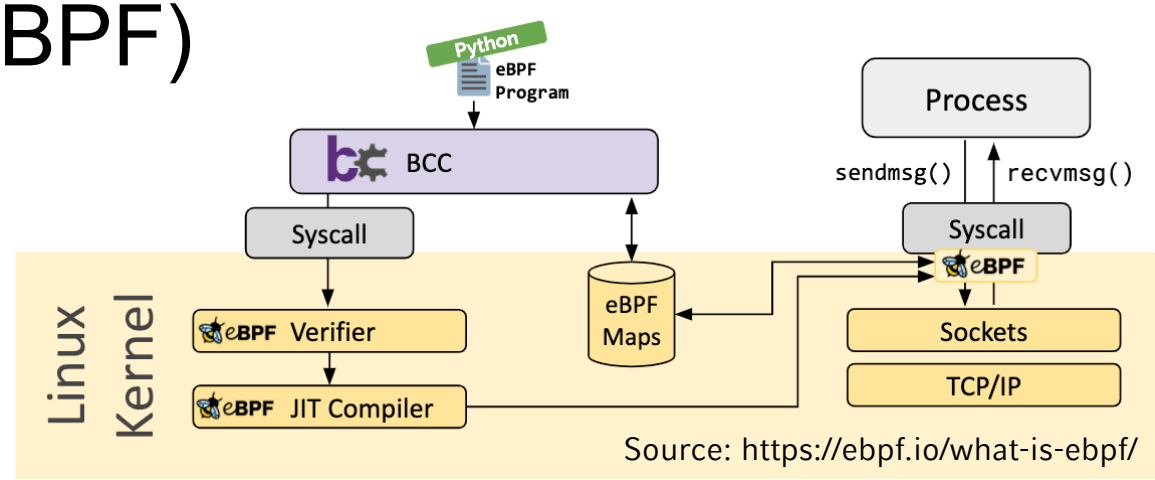
Objectives

- ➔ A performance characterization of containerized HPC applications with unknown behavior.
- ➔ “Map a container to a target system.”
- ➔ With a quantified and comparable runtime characteristic for HPC containers: we call this a Fingerprint.

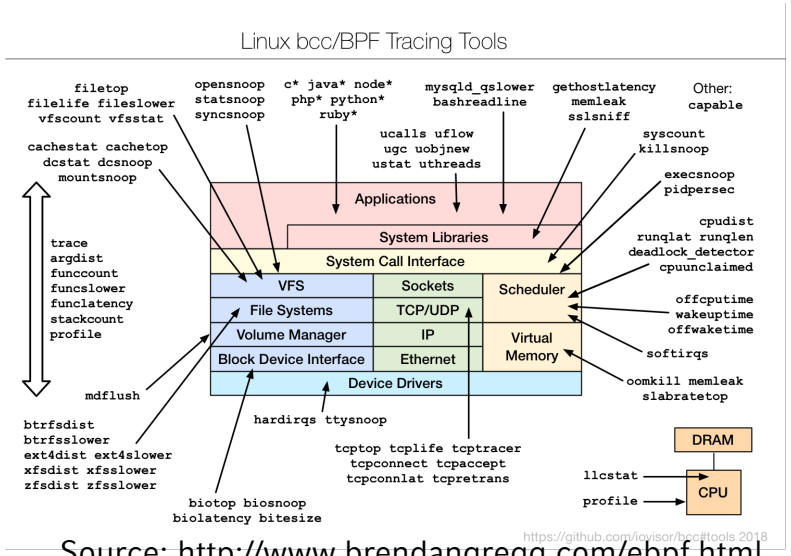
EASEY Architecture



- Kernel-level measurements (eBPF)
- Perform custom analyses
  - Compute usage
  - Memory usage
  - Network/Communication pattern
    - I/O usage
  - [Power Usage]



Source: https://ebpf.io/what-is-ebpf/

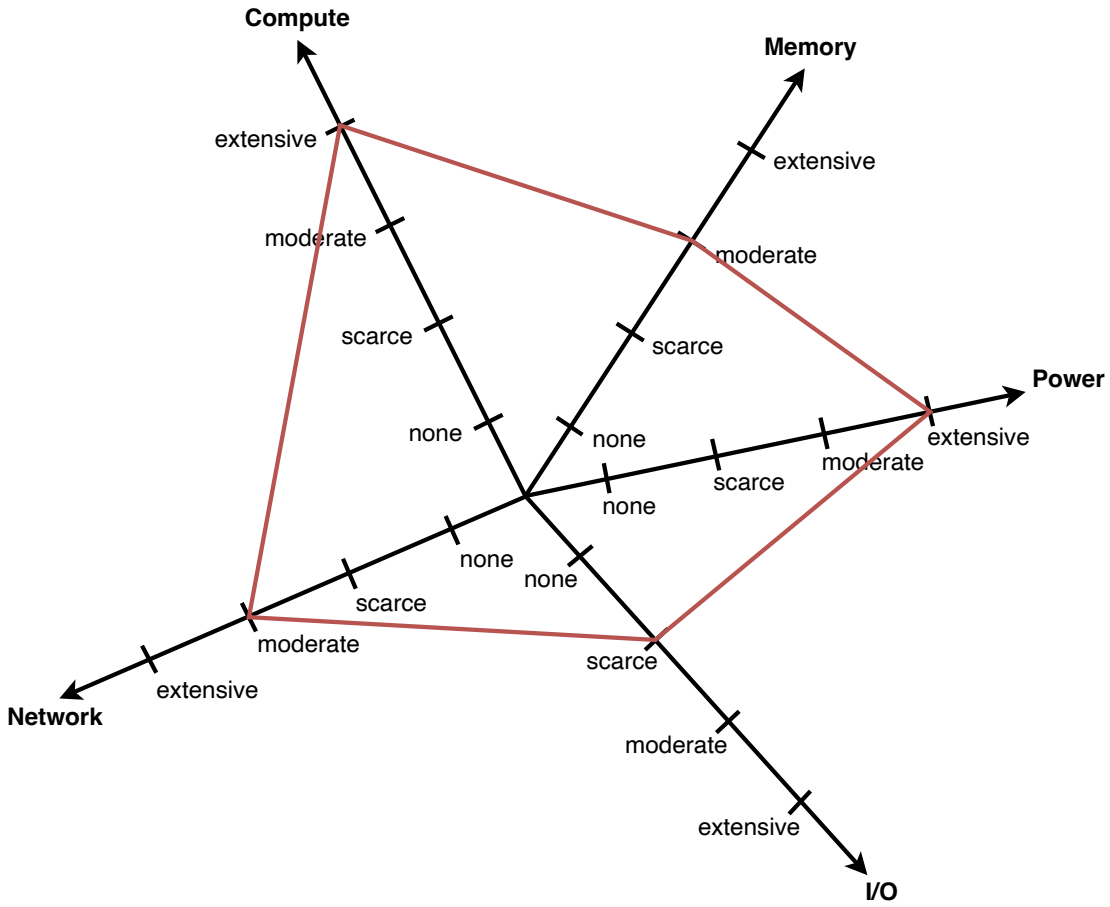


Source: http://www.brendangregg.com/ebpf.html

Categorization example with Dimensional Star

Based on kernel-level measurements:

- container categorization in all relevant dimensions

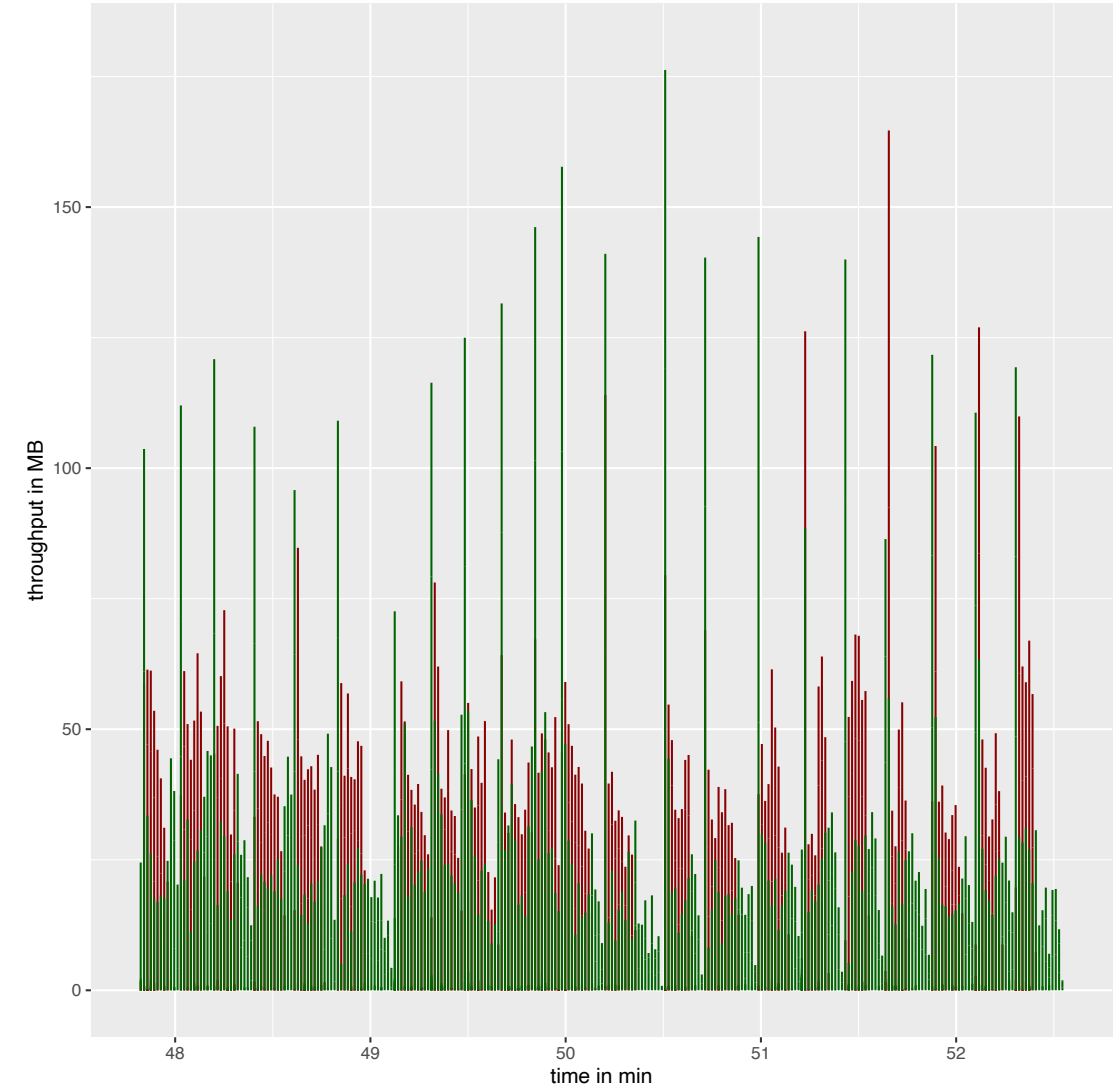


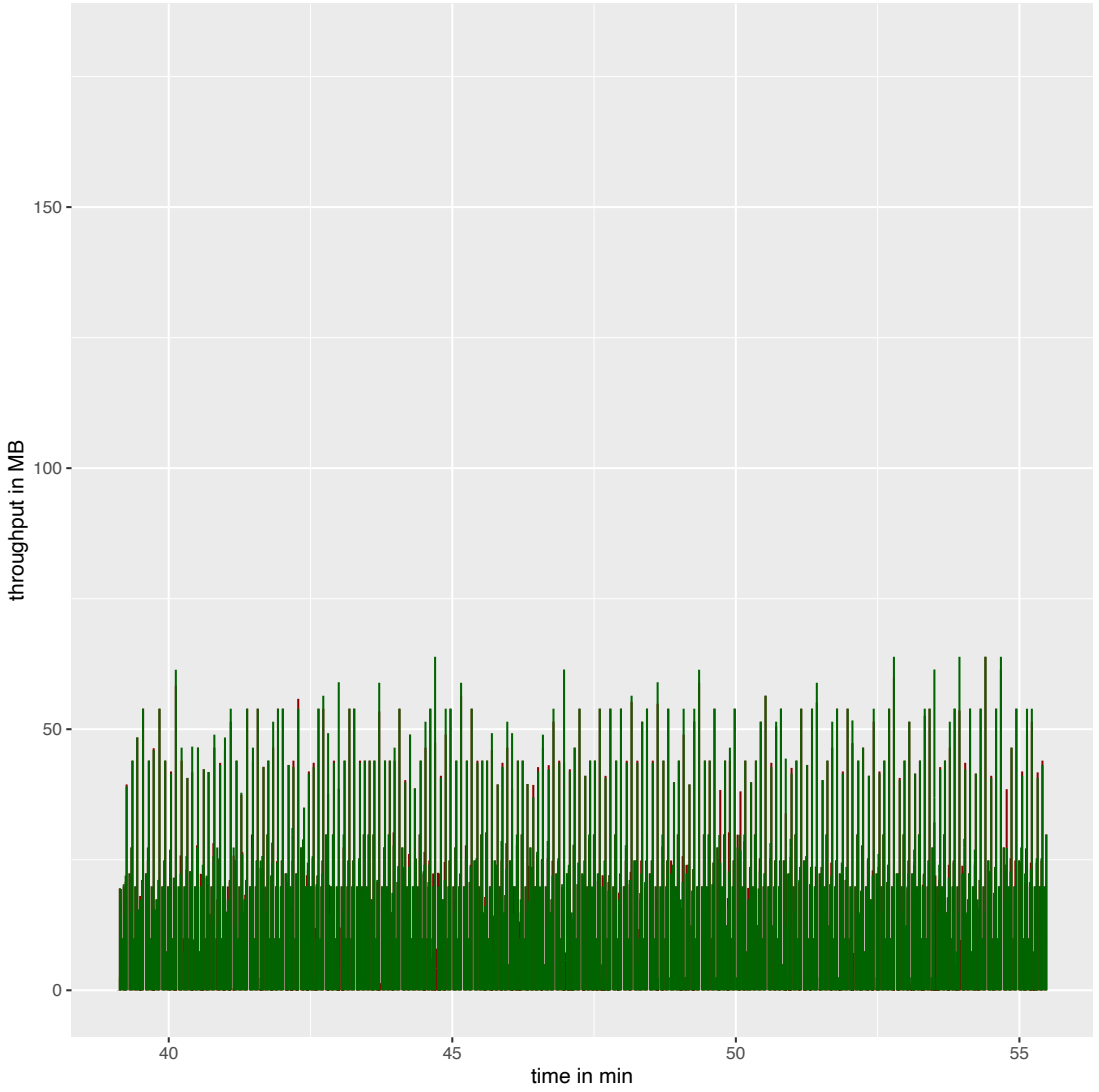
Algorithm 1: Fingerprinting Algorithm

Result: A quantified and comparable runtime characteristic

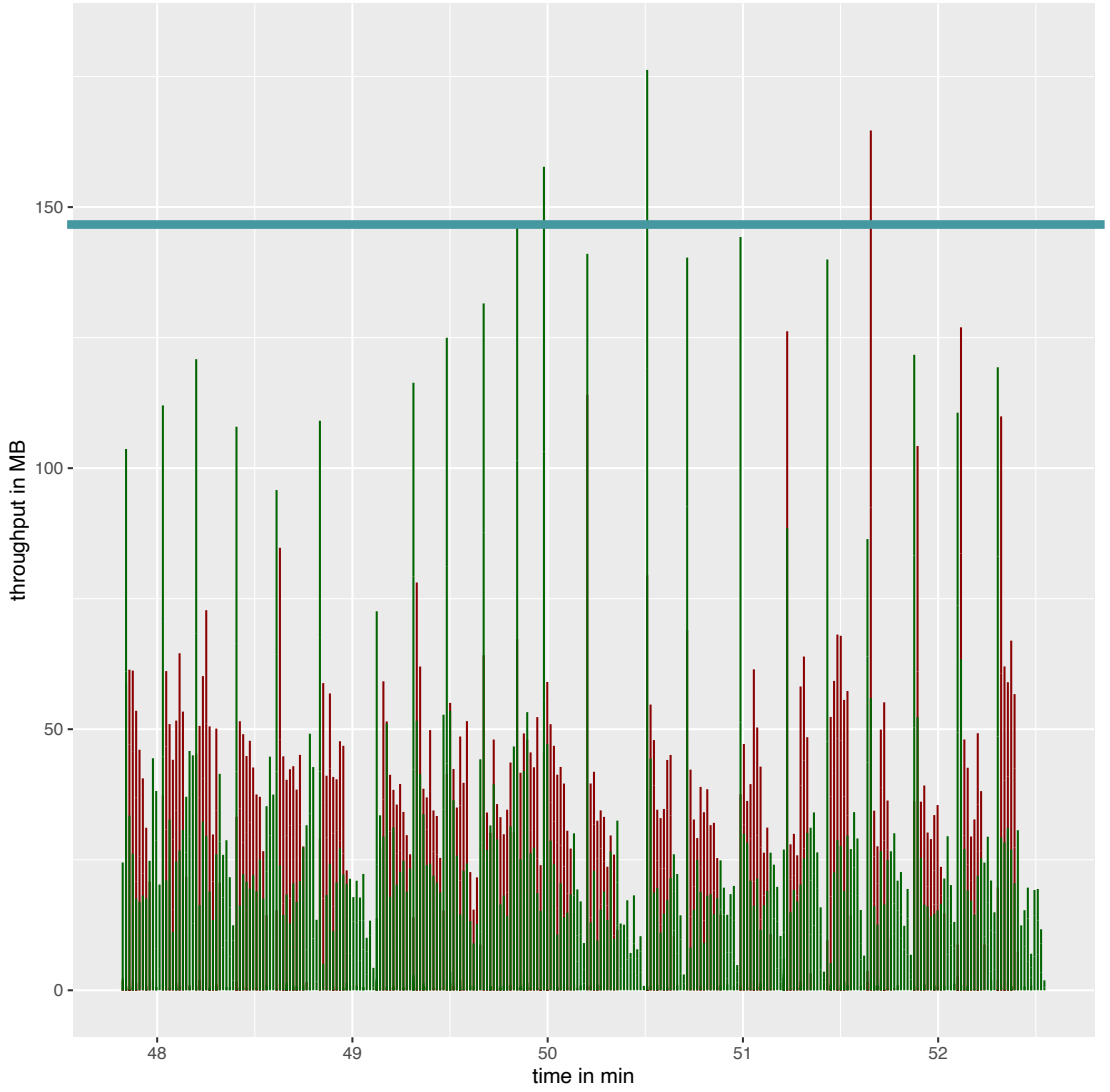
```
Low-LevelMeasurements(compute,memory,i/o,network);
PowerMeasurements();
ProcessData(measurements);
ApplyStatisticalModel(data);
ExportFingerprintValues();
```

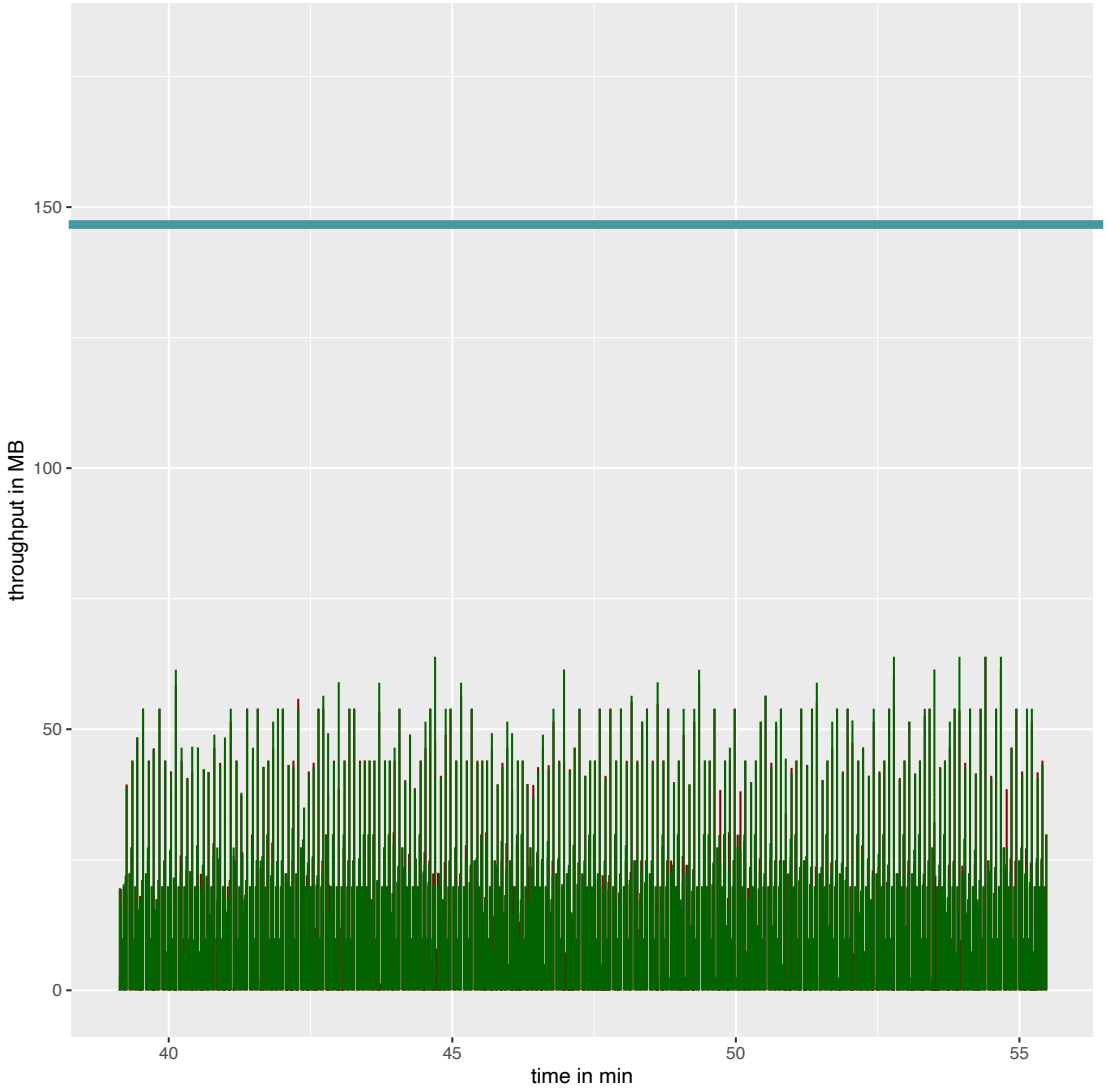
Communication Example



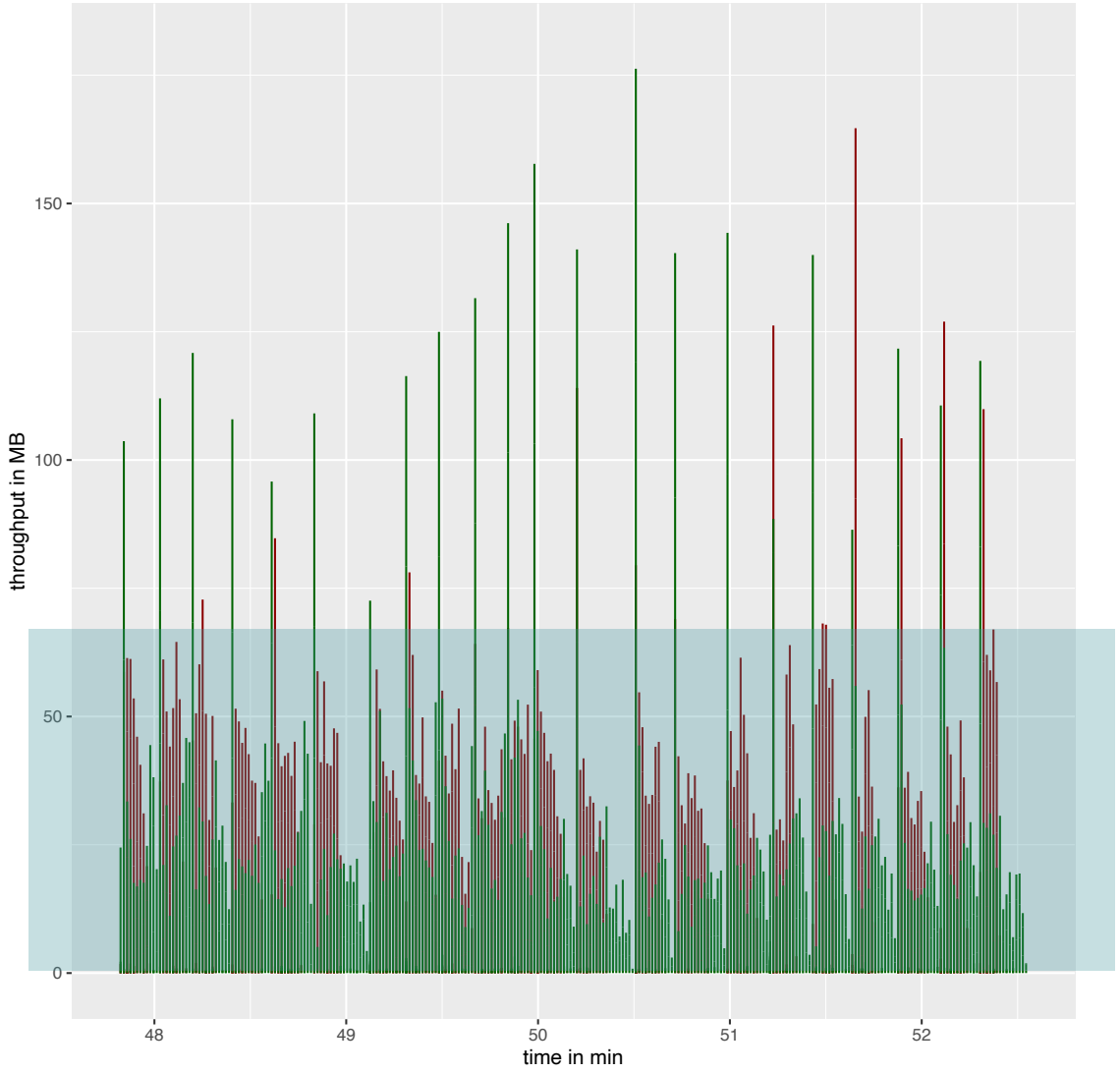


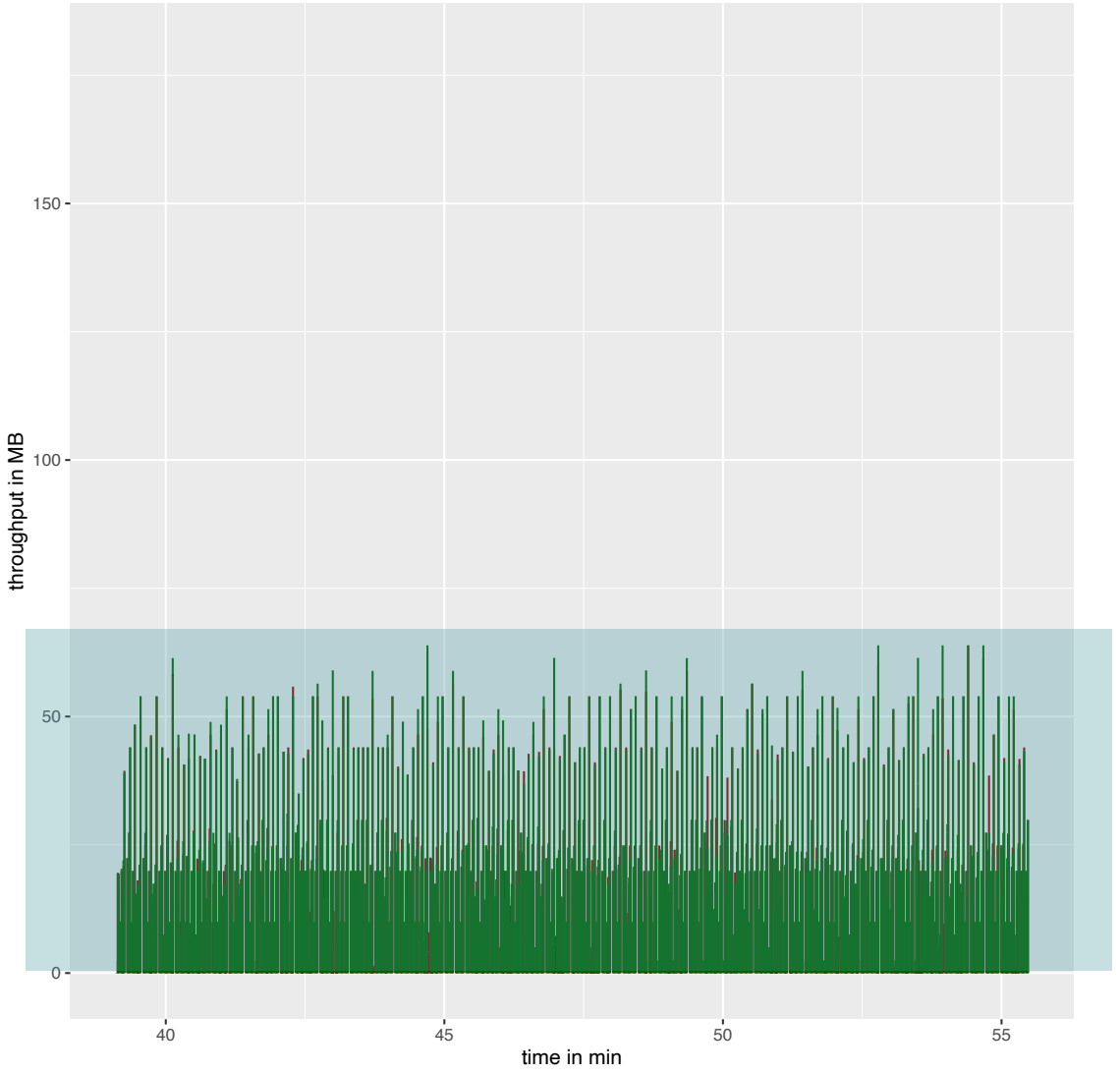
Communication Example – Detecting Peaks





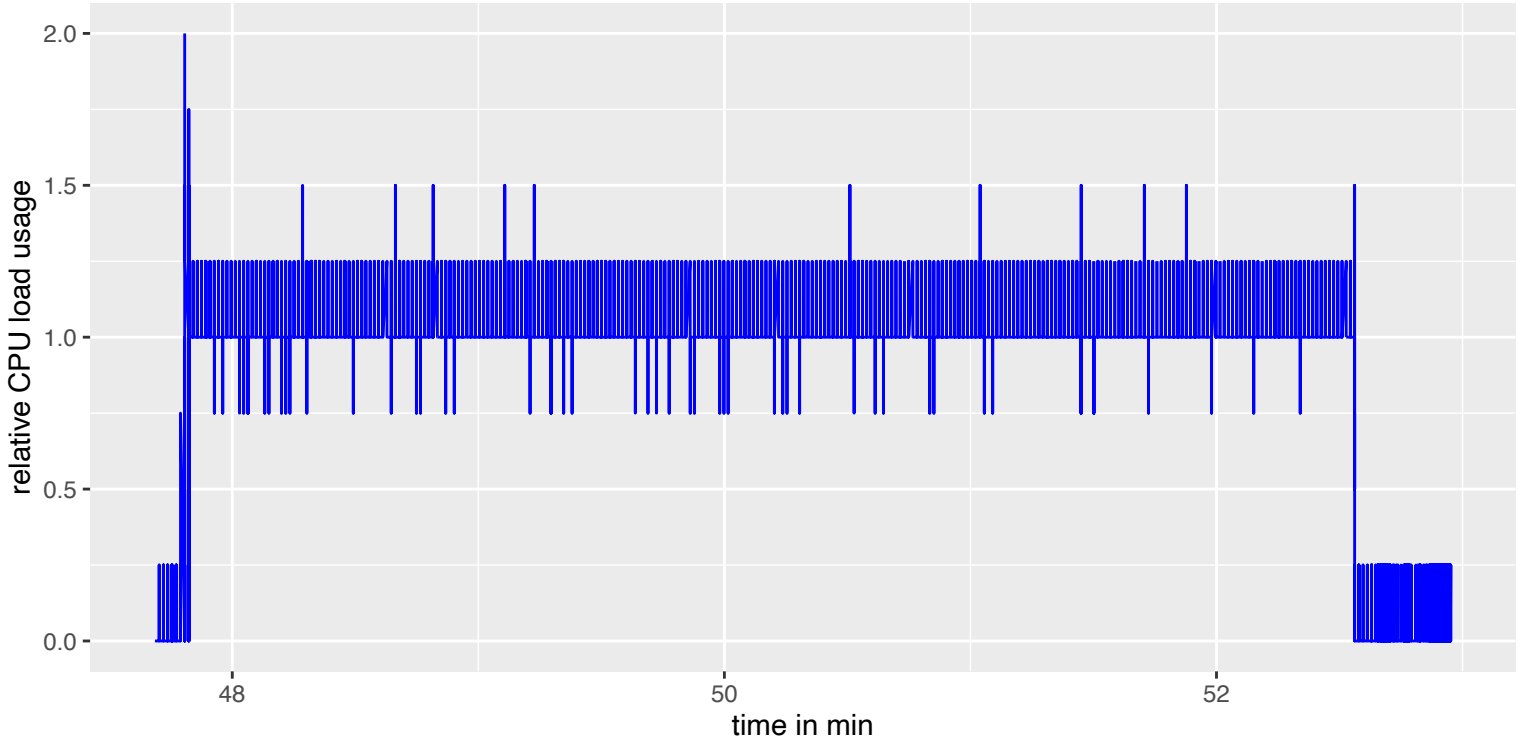
Communication Example – Detecting Throughput





Fingerprint Values

→ N-tupel of relevant Fingerprint-Values



FINGERPRINT

$u(comp)$	(1.45)
$\mathrm{P}(u(comp[3]))$	(0.02)
$u(mem)$	(2.00)
$\mathrm{P}(u(mem[3]))$	(0.02)
$B_{u(comp[3])}(mem)$	(0.05)
$B_{u(mem[3])}(comp)$	(0.05)
...	

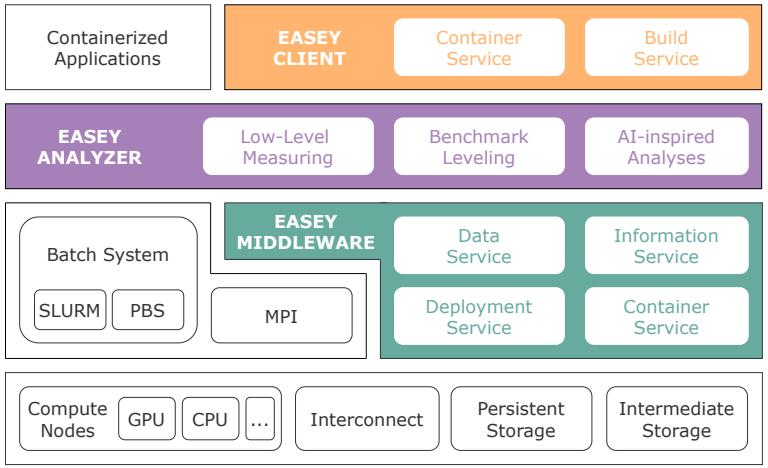
Fingerprint Mapping

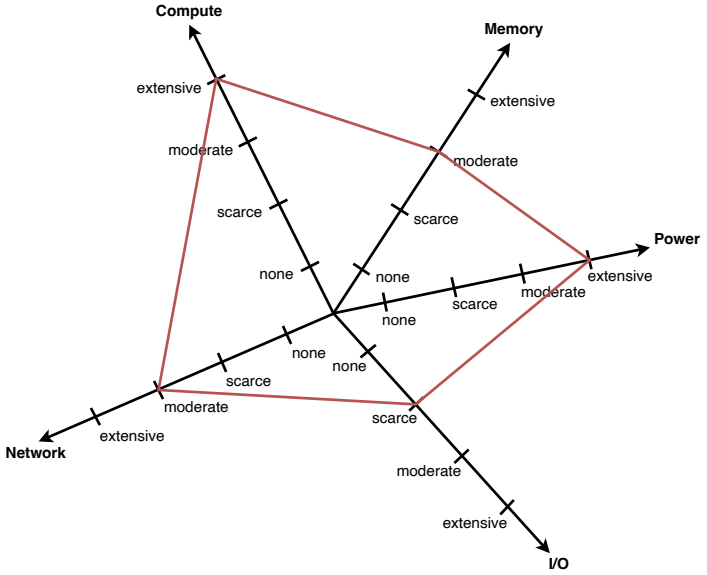
$$\boldsymbol{F_c = \left(u(comp), u(comp), u(io), \ldots, \mathbf{c}\left(u(comp[3])\right), \ldots, B_{u(mem[3])}(comp), \ldots\right)^T}$$

Mapping to a System Fingerprint within specification:

SYSTEM		<->	CONTAINER	
$u(comp)$	$[1.2; 2.6]$		$u(comp)$	(1.45)
$\mathrm{P}(u(comp[3]))$	$[0.00; 0.20]$		$\mathrm{P}(u(comp[3]))$	(0.02)
$u(mem)$	$[0.0; 3.0]$		$u(mem)$	(2.80)
$\mathrm{P}(u(mem[3]))$	$[0.00; 1.00]$		$\mathrm{P}(u(mem[3]))$	(0.02)
$B_{u(comp[3])}(mem)$	$[0.00; 1.00]$		$B_{u(comp[3])}(mem)$	(0.05)
$B_{u(mem[3])}(comp)$	$[0.00; 0.40]$		$B_{u(mem[3])}(comp)$	(0.05)

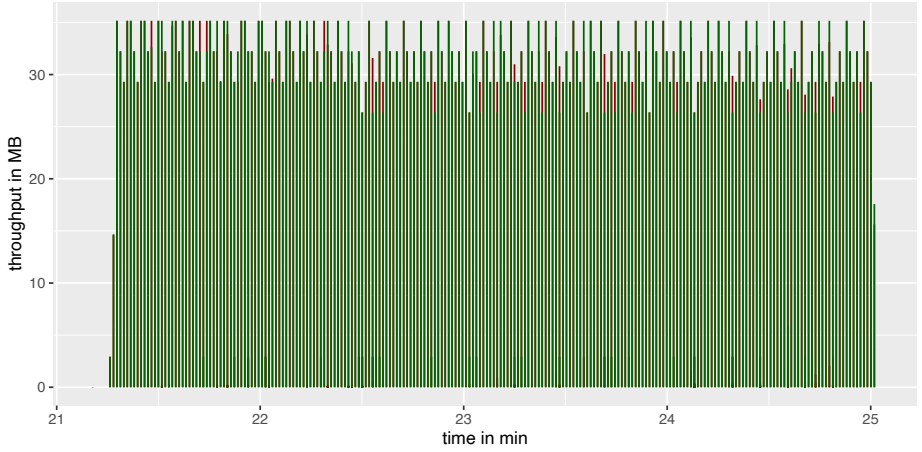
Summary





$u(comp)$	(1.45)
$\mathrm{P}(u(comp[3]))$	(0.02)
$u(mem)$	(2.00)
$\mathrm{P}(u(mem[3]))$	(0.02)
$B_{u(comp[3])}(mem)$	(0.05)
$B_{u(mem[3])}(comp)$	(0.05)
...	

FINGERPRINT



Maximilian Höb
hoeb@mnm-team.org

MNM

TEAM

Munich Network Management Team