A User Evaluation Report of Energy Efficiency Benchmark

Leon Tabaro, Ahmed M. A. Sayed, Mona Jaber

School of EECS, Queen Mary University of London, UK

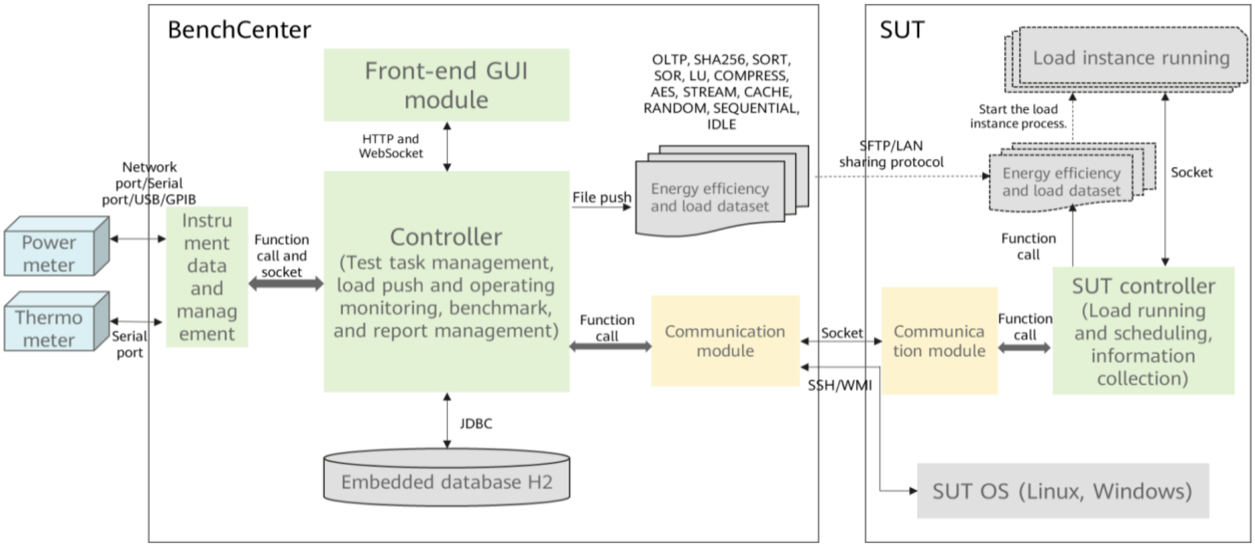
Introduction

At the current rate of global data-centre expansion driven by surging demand from advanced AI workloads, large-scale analytics, and cloud services, server energy efficiency (EE) has become a first-order constraint. Operators face power-and-cooling limits, grid-capacity constraints, and rising scrutiny to reduce operating expenditure while demonstrating measurable progress against sustainability targets. As utilisation and rack-level power density increase, organisations require comparable, auditable metrics to inform procurement, configuration, and operational policy.

Benchmarking addresses this need by providing standardised procedures that relate work performed to energy consumed under controlled conditions. A good EE benchmark is characterised by (i) representativeness of relevant workload behaviours, (ii) repeatability across laboratories, and (iii) transparency in test environment, instrumentation, and analysis. An industry trend toward common practices, e.g., multi-load evaluations, performance-per-watt metrics, and detailed environmental manifests, has improved comparability of results across platforms and sites. Within this context, the document presents the results of evaluating the BenchSEE, a server EE benchmarking tool, focusing on its usability and stability in a lab setting.

1.1 BenchSEE

The Benchmark of Server Energy Efficiency (BenchSEE) is a tool developed by the Branch of Resource and Environment of the China National Institute of Standardisation (CNIS) for measuring and evaluating the energy efficiency of server systems. The test benchmark evaluates the energy efficiency of key server components, including the CPU, memory, and storage, via pre-configured workloads. The tool includes automatic report generation and cross-run comparison to facilitate the identification of energy-efficient configurations and systems. BenchSEE is designed based on the opinions of multiple server and chip vendors, energy efficiency certification organizations, and scientific research institutes in the IT energy-saving field, to meet the requirements for energy-efficiency evaluation in the server market. BenchSEE supports server hardware platforms such as ARM, x86_64, MIPS, and PowerPC, and is compatible with mainstream operating systems (OSs) such as Linux and Windows Server. Below is a figure depicting the BenchSEE benchmark architecture.


BenchCenter
Front-end GUI module
HTTP and WebSocket
OLTP, SHA256, SORT, SOR, LU, COMPRESS, AES, STREAM, CACHE, RANDOM, SEQUENTIAL, IDLE
Network port/Serial port/USB/GPIB
Power meter
Thermo meter
Serial port
Instrument data and management
Function call and socket
Controller (Test task management, load push and operating monitoring, benchmark, and report management)
File push
Energy efficiency and load dataset
Function call
Communication module
JDBC
Embedded database H2
SFTP/LAN sharing protocol
Socket
SSH/WMI
SUT
Load instance running
Start the load instance process.
Energy efficiency and load dataset
Socket
Function call
Communication module
Function call
SUT controller (Load running and scheduling, information collection)
SUT OS (Linux, Windows)

1.2 Objectives of this Report

- **Assess usability**: This includes the user experience during equipment setup (software and hardware), the clarity of the user interface and documentation, and troubleshooting and reporting.
- **Assess stability**: This includes the reliability and consistency of results across identical runs, the resilience of the measurements under different workloads, and the data collection integrity.
- Provide **evidence-backed recommendations** to improve the usability and stability of BenchSEE.

1.3 Methodology

- **Environment definition**: Document System Under Test (SUT), controller, OS, firmware, and instrumentation (power meter, temperature).
- **Workloads & configuration:** Use BenchSEE's predefined workloads targeting CPU, memory, and storage. Record all parameters and save task configurations.
- **Run protocol**: Calibrate instruments → execute test suite → capture logs, reports, and screenshots.
- **Validity criteria**: Ambient temperature threshold, instrument calibration, workload completeness.
- **Data capture**: Store HTML/PDF/CSV/TXT reports, controller/SUT logs, and BenchSEE task configs with versioning.
- **Repeatability**: At least N≥3 runs per workload profile on identical settings to quantify variance.
- **Analysis approach**: Descriptive statistics, run-to-run variance (%), incident categorisation, severity triage.

- **Acceptance criteria:** A run set is considered stable if all load levels meet the coefficient of variation (CoV- See Section 3) threshold and no data-collection errors or disconnects occur.

1. Usability of BenchSEE

This section provides a comprehensive analysis of the BenchSEE user experience. Results are given in a question-answer format to highlight key performance metrics we seek to evaluate at this stage. **A full summary with evaluation scores is given in Appendix A.**

2.1 Setup & Installation

Question	Answer
How straightforward was installation on the controller?	Installation of the BenchSEE on the host was generally smooth. The packaged launcher detected Java and network interfaces correctly, and the SUT connected on first attempt. Section 4.3 Procedure of the Test User Guide gave clear installation instructions that were easy to follow. The two options for installing the Java Running Environment were clearly laid out with clear reasoning for each path and helpful suggestions.
Complete Installation Duration	<5 minutes.
Were prerequisites clearly identified and verified?	Prerequisites were clearly documented and matched what the software checked at startup. Java ≥8 was recognised; 1 GbE connectivity and WinRM access validated during SUT registration.

2.2 Configuration & Task Creation

Question	Answer
Startup duration	<1min utilizing the bat script.
How easy was it to create and configure a new task (define SUT, instruments, workloads)?	Task creation was intuitive and linear: add SUT → add instruments → select workloads → save. We created and configured baseline profile task in under five minutes.
Could prior tasks be cloned/edited efficiently?	Cloning preserved all configuration parameters and workload sets, accelerating iteration. Edits are scoped and reversible before execution.

2.3 Instrument Integration (Power/Temperature)

Question	Answer
How easy was it to connect power sources?	Tests were run using the RS-232 interface type and the GPM-8213 power meter device model. Only minor issues with detecting the device at first, however this was resolved when switching to a new serial cable.
How easy was it to configure Simulation power and temperature?	Simulation sources were trivial to enable and useful for testing. Curves rendered live, allowing us to validate the full pipeline before attaching real meters.
How easy was it to configure IPMI power?	As our testbed was limited to Dell servers using iDRAC, which is not standard IPMI power control, IPMI power configuration was not completed.
Are error messages informative when instruments fail?	In general, yes.

2.4 UI/UX & Navigation

Question	Answer
Task information display	Task details are well consolidated on the Run page. The header shows task/run IDs, SUT and workload set, while the timeline clearly labels phase and load-level progression. A live panel presents power, performance, and temperature with hover tooltips for precise values, and links to logs/reports are surfaced at completion.
Are live telemetry and progress indicators clear and responsive?	Live charts updated smoothly with simulated sources; run phases and ETA were visible. However, the ETA is generally inaccurate and underestimates total task completion time.

2.5 Documentation & Support

Question	Answer
Could documentation resolve issues without external help?	Documentation was sufficient for setup, first runs, and interpreting reports. It included environment notes (Java, networking) and validity criteria, which matched our observations.
Are examples and screenshots aligned with the current UI?	Examples were indeed aligned to the current build.

2.6 Reporting & Export

Question	Answer
Are generated reports easy to locate, read, and interpret?	Reports were produced automatically at run completion and linked from the task page. Layout, legends, and metrics were readable without cross-referencing.
Is it easy to extract structured data (CSV/TXT) for analysis?	CSV/TXT exports were triggered with one click and opened cleanly in analysis tools. Column naming was consistent with UI terms.
Do reports contain sufficient metadata (SUT, instruments, config)?	Reports captured SUT hardware, OS, instrument details, and workload parameters, enabling reproducibility. This metadata also helped diagnose instrument issues.

2.7 Overall Usability Analysis

The interface supports a clear end-to-end workflow with intuitive dialogs for system registration, instrument selection, workload configuration, and run monitoring. During execution, live readings and status cues make progress easy to track; on completion, the reporting views and one-click exports facilitate rapid analysis and cross-run comparison. The accompanying documentation provides concise installation steps, accurate screen references, and practical troubleshooting paths that were sufficient to resolve routine setup issues.

Two minor limitations were observed: (i) incomplete IPMI power integration due to the fact that only iDRAC is available in this evaluation, which is not standard IPMI; and (ii) the absence of an export function for the comparative analysis report, which limits portability and external review of aggregated results. Neither issue blocked testing, but both are candidates for near-term refinement. On balance, BenchSEE enables efficient, low-friction benchmarking for server EE studies and is well-suited to laboratory adoption.

3. Stability of BenchSEE

This section analyses the stability of BenchSEE across 10 runs executed over four tasks on different days and times under identical workload configurations. Stability is quantified using the coefficient of variation (CoV), defined here as %CoV = 100 × (standard deviation/mean) for each workload/load-level. We adopt %CoV ≤ 3% as the primary repeatability threshold (indicative), with 3–5% flagged as amber and >5% as red. Under these criteria, the comparative analysis shows minimal variability in the majority of workload scores across load levels, with only a small number of low-load cases exhibiting higher relative dispersion (detailed in §3.5).

3.1 Run Consistency (Repeatability)

Question	Answer
How consistent are results across N≥3 identical runs per workload? (report % variance)	Across 10 runs spanning 4 tasks over multiple days/times, workload scores were visually tightly clustered in the comparative charts (see Appendix D). For the most part, bar heights across repeats are identical at each load level, indicating low run-to-run spread.

3.2 Runtime Robustness

Question	Answer
Did any runs crash or hang? Provide frequency and conditions.	No crashes or hangs observed across the 10 completed runs. All tasks reached report generation without manual intervention.
Any controller ↔ SUT connection drops?	No controller–SUT disconnects during execution. Live readings and progress indicators continued uninterrupted.
Does long duration testing (≥2 hours) remain stable?	Yes. The longest tasks (~2 hours) completed reliably with steady UI responsiveness and report creation.

3.3 Telemetry Stability

Question	Answer
Is power/temperature telemetry continuous and time-aligned?	With simulated sources, curves updated smoothly and aligned with workload phases.
Are invalid/edge conditions flagged (e.g., ambient too low)?	The tool surfaces data quality/validity cues in the UI and reports. No invalid ambient flags were triggered in these runs.

3.4 Reporting Reliability

Question	Answer
Are reports generated without errors after each run?	Yes, 10/10 runs produced comparative and per run reports with no missing sections. Links were available immediately from the task page.
Any missing charts/tables or corrupted files?	None observed. Bar charts for all workloads and load levels rendered consistently across reports.

3.5 Overall analysis of stability evaluation metrics

Across the 40 (workload, load-level) combinations, results were generally consistent. Using **%CoV ≤ 3%** as a practical repeatability threshold, the vast majority of cases met the criterion (**see Appendix B for detailed results and representative charts**). Variability was concentrated in a small subset at lower load levels:

- Cases with higher %CoVs include: LU-50%, OLTP-25%, SORT-25%, AES-25%, OLTP-37.5%, SORT-50%, LU-25%, with an average %CoV around 5%.

These patterns are expected: at lower absolute throughput, small fluctuations appear as larger relative variation. The tightly clustered mid–high load results, together with 10/10 successful repetitions per point, support a positive stability assessment.

More results from the comparative analysis report, specifically “04 Raw Performance Score Comparison of Each Workload” and “05 Performance–Power Ratio Comparison,” are not reproduced here for space; however, the figures presented are representative of the broader trends observed.

4. Application scenarios of BenchSEE in research and industry

4.1 Academic research

BenchSEE provides a controlled, reproducible environment for empirically studying server-level energy efficiency. Typical use cases include:

- **Architecture and systems studies.** Quantify the energy/performance impact of CPU microarchitecture, memory-channel population, storage configurations, and NUMA/affinity policies across multi-load regimes.
- **Algorithmic and software optimisations.** Evaluate compiler flags, runtime libraries, thread schedulers, and I/O stacks; assess trade-offs between throughput, latency, and energy per task.
- **Energy-aware resource management.** Prototype scheduling and power-capping policies (e.g., DVFS, SMT, turbo, C-states) and measure their effects on perf/W and stability.
- **Benchmark methodology research.** Compare variance reduction techniques, window lengths, reporting precision, and guard-band definitions (e.g., %CoV thresholds) to improve repeatability.

Recommendations for academic use.
Adopt a preregistered protocol (workloads, load levels, acceptance thresholds), publish the environment manifest and task config alongside datasets, and report mean, SD, %CoV for each (workload, load-level). Where possible, archive raw exports and analysis scripts to an institutional or open repository.

4.2 Industry and enterprise

BenchSEE can be integrated into engineering and operations workflows to improve procurement quality, platform tuning, and sustainability reporting:

- **Procurement and vendor selection.** Compare candidate platforms on **performance-per-watt** under identical load profiles; require environment manifests and repeatability thresholds for acceptance.
- **Capacity and thermal planning.** Use multi-load curves (100→25% or finer) to derive rack-level power envelopes, informing facility provisioning and thermal risk assessment.
- **Change management and CI.** Run nightly or pre-release stability regressions to detect energy/performance drift after firmware, driver, or OS updates.
- **Sustainability and compliance reporting.** Generate auditable records that support internal carbon accounting, customer disclosures, and alignment with emerging efficiency programmes.
- **Operational policy validation.** Quantify the impact of power caps and scheduler policies on service-level objectives; set guard-bands where %CoV or tail latency degrades.

Limitations and extensions.
Where workloads rely on accelerators or advanced NIC/DPU offload, consider adding complementary worklets and, if needed, higher-rate side captures for transient analysis. Adding an export for comparative analyses (aggregated HTML/PDF + CSV bundle) will further streamline adoption in both research and industry contexts.

5. Summary

This report evaluated **BenchSEE** as a server energy-efficiency benchmarking tool with emphasis on usability and stability in a realistic laboratory setting. Overall, BenchSEE proved highly usable and stable in our environment. BenchSEE provided a low-friction workflow with clear configuration dialogs, live metrics during runs, and autogenerated reports with CSV/TXT exports. Across 10 runs, stability was high: using %CoV = 100 × (SD/mean) and a ≤ 3% repeatability threshold, major workload–load-level combinations met the criterion; variance was concentrated at lower load levels (e.g., LU-50%, OLTP-25%). Two main friction points were (i) incomplete IPMI power integration due to using iDRAC and (ii) the absence of an export function for the comparative analysis report, which constrains portability of aggregated findings. We used a calibrated external power meter for measurements and simulated temperature; for completeness we recommend adding a physical temperature probe. We also recommend integrating mean/SD/**%**CoV computation and threshold flagging directly into comparative reports; adding a guided IPMI setup and validation test; enabling export of the comparative analysis as a bundled HTML/PDF + CSV. Overall, BenchSEE already supports reproducible, evidence-rich energy efficiency studies with a small set of targeted enhancements.

Appendix A

Summary of Usability and Stability

Scale: 1 = poor, 2 = fair, 3 = adequate, 4 = good, 5 = excellent.

Area	Feature / Criterion	Score (1–5)	Rationale (brief justification)
Usability	**Setup & Installation**	**5**	Packaged launcher worked on first try; Java/network auto-detected; SUT connected first attempt; clear install steps.
	Configuration & Task Creation	**5**	Intuitive linear flow (add SUT → instruments → workloads → save); cloning/editing tasks accelerates iteration.
	Instrument Integration (Power/Temp)	**4**	RS-232 meter worked (minor cable issue resolved); simulation sources easy; IPMI power not available in this evaluation (iDRAC not standard IPMI).
	UI/UX & Navigation	**4**	Run page consolidates key info; live readings and progress clear; charts readable; minor polish items remain.
	Documentation & Support	**4**	Sufficient for install, first runs, and report interpretation; examples/screens align with current UI.
	Reporting & Export	**4**	Auto-generated reports + one-click CSV/TXT; no export for comparative analysis bundle yet (limits portability/review).
Stability	**Run Reliability (crashes/hangs)**	**5**	10/10 runs completed; no crashes or hangs; reports generated without intervention.
	Connection Robustness (controller↔SUT)	**5**	No disconnects; readings and progress indicators uninterrupted.
	Long-Duration Stability (≈2 h)	**5**	Longest tasks (~1:58) completed reliably; UI remained responsive.
	Repeatability (%CoV)	**4**	Majority of (workload, level) points ≤ **3% CoV**; a small set at low loads exceeded threshold (e.g., LU-50%, OLTP-25%).
	Telemetry Continuity & Alignment	**5**	With simulated sources, readings updated smoothly and aligned with workload phases; no invalid-ambient flags.
	Reporting Reliability	**5**	All runs produced complete comparative and per-run reports; no missing charts/tables or corrupted files.

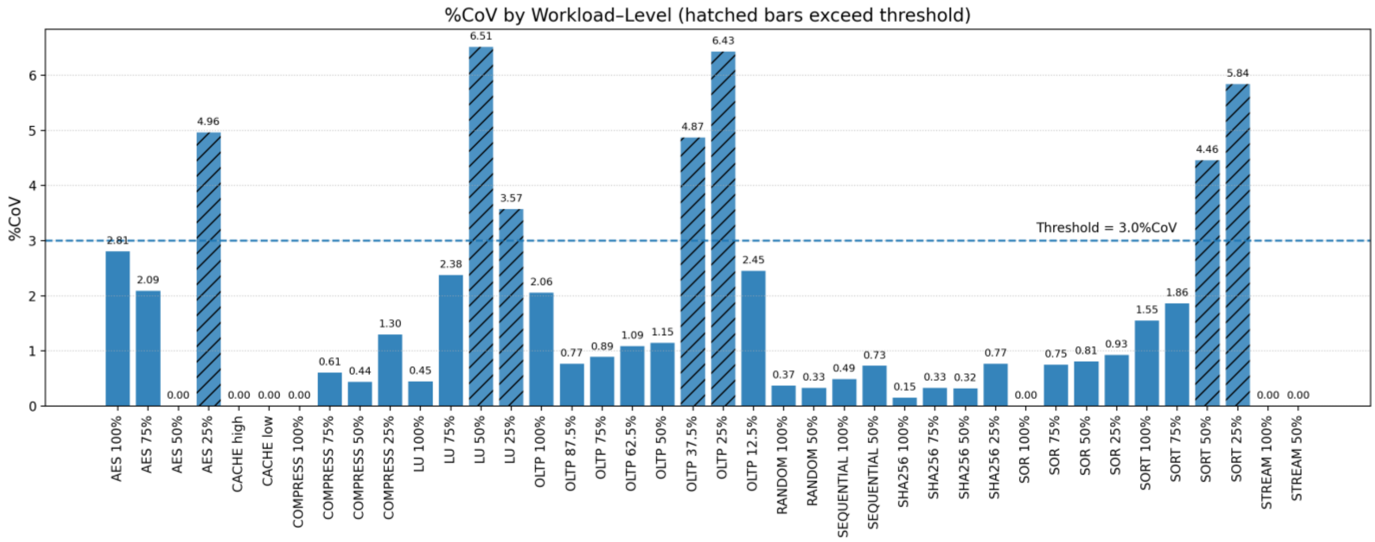
Appendix B

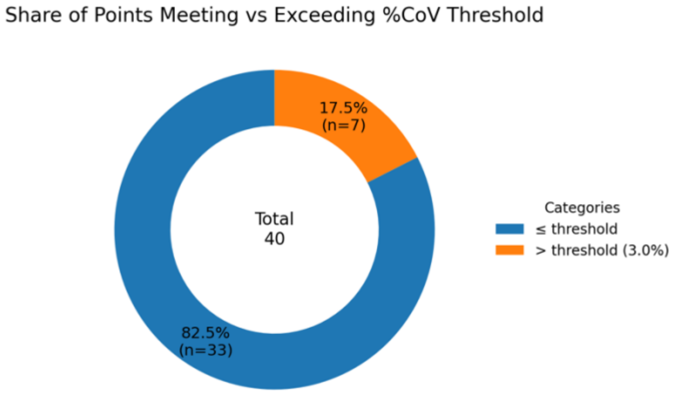
Stability of the results from the benchmark for different workloads at various load levels

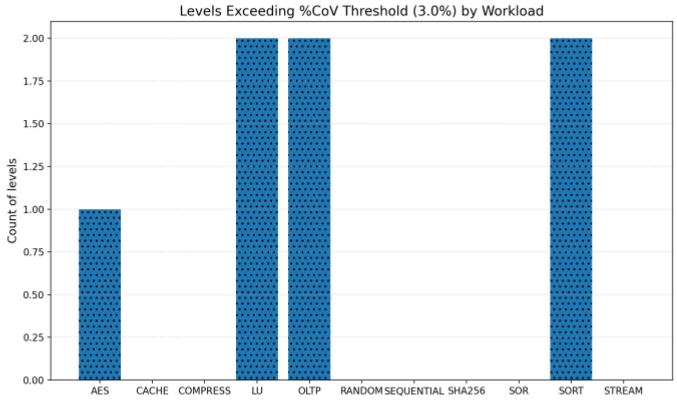
Workload	Level	Mean	StdDev	%CoV	N
AES	100%	1.840	0.052	2.81	10
AES	75%	1.510	0.032	2.09	10
AES	50%	1.200	0.000	0.00	10
AES	25%	1.040	0.052	4.96	10
COMPRESS	100%	9.700	0.000	0.00	10
COMPRESS	75%	8.460	0.052	0.61	10
COMPRESS	50%	7.110	0.032	0.44	10
COMPRESS	25%	5.360	0.070	1.30	10
LU	100%	7.120	0.032	0.45	10
LU	75%	5.990	0.143	2.38	10
LU	50%	5.080	0.331	6.51	10
LU	25%	4.420	0.158	3.57	10
OLTP	100%	6.140	0.127	2.06	10
OLTP	87.5%	5.520	0.042	0.77	10
OLTP	75%	4.890	0.044	0.89	10
OLTP	62.5%	4.300	0.047	1.09	10
OLTP	50%	3.680	0.042	1.15	10
OLTP	37.5%	3.250	0.158	4.87	10
OLTP	25%	2.350	0.151	6.43	10
OLTP	12.5%	1.720	0.042	2.45	10
SHA256	100%	21.100	0.032	0.15	10
SHA256	75%	18.950	0.063	0.33	10
SHA256	50%	15.460	0.050	0.32	10
SHA256	25%	11.650	0.090	0.77	10
SOR	100%	8.800	0.000	0.00	10
SOR	75%	8.560	0.064	0.75	10
SOR	50%	7.310	0.059	0.81	10
SOR	25%	5.340	0.050	0.93	10
SORT	100%	5.820	0.090	1.55	10
SORT	75%	4.850	0.090	1.86	10

Workload	Level	Mean	StdDev	%CoV	N
SORT	50%	3.840	0.171	4.46	10
SORT	25%	2.660	0.155	5.84	10
CACHE	high	0.200	0.000	0.00	10
CACHE	low	0.200	0.000	0.00	10
STREAM	100%	2.300	0.000	0.00	10
STREAM	50%	1.700	0.000	0.00	10
RANDOM	100%	25.530	0.095	0.37	10
RANDOM	50%	12.820	0.042	0.33	10
SEQUENTIAL	100%	18.010	0.088	0.49	10
SEQUENTIAL	50%	9.230	0.068	0.73	10

Summary of stability metrics

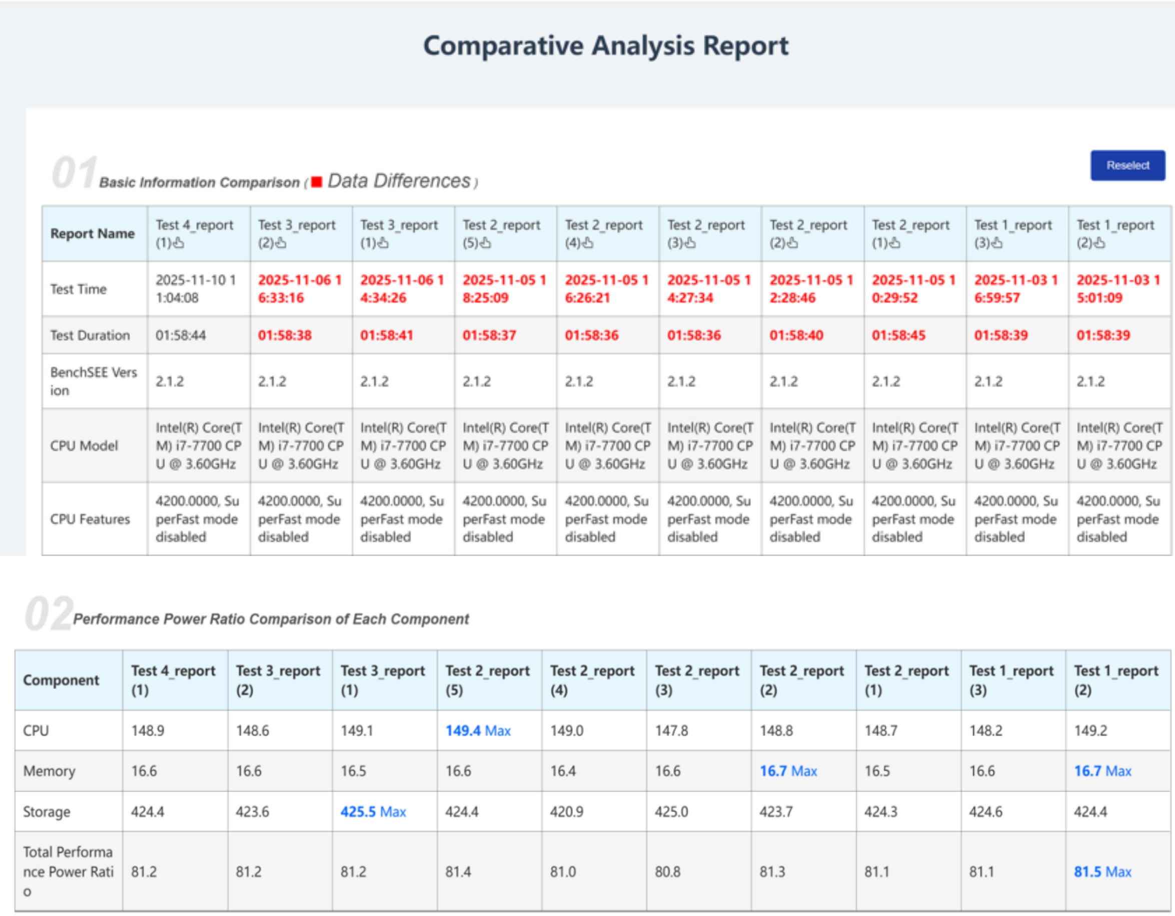






Appendix D

Comparative analysis report from screenshots taken from the benchmark.



Comparative Analysis Report

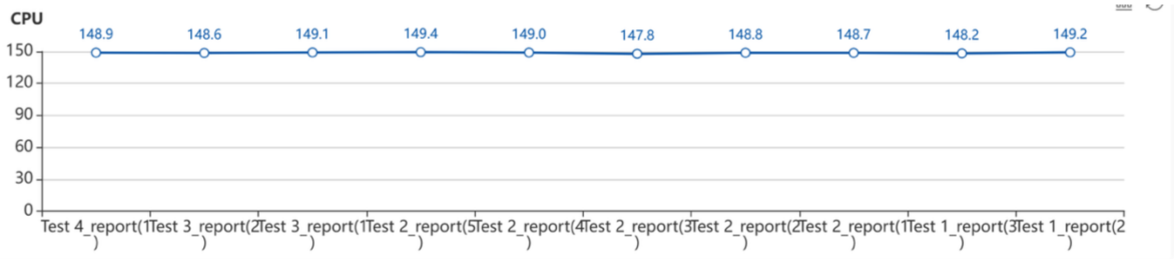
01 Basic Information Comparison (■ Data Differences)

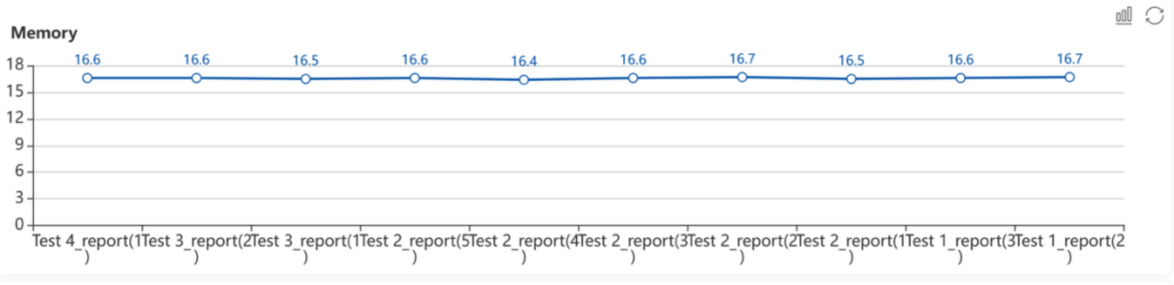
Reselect

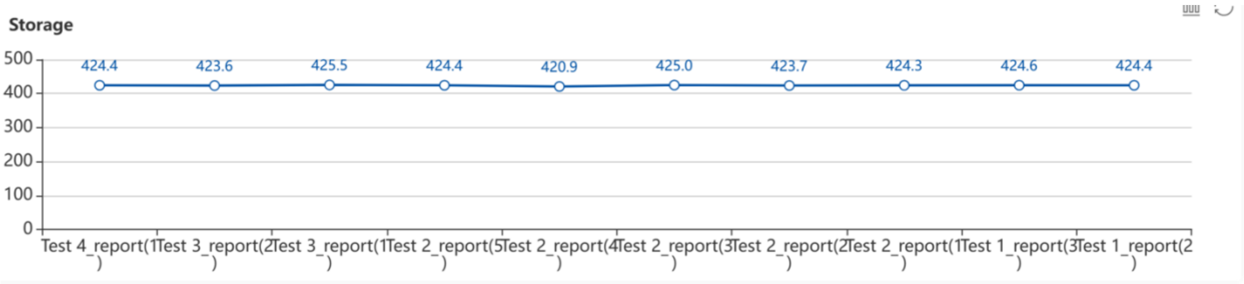
Report Name	Test 4_report (1)	Test 3_report (2)	Test 3_report (1)	Test 2_report (5)	Test 2_report (4)	Test 2_report (3)	Test 2_report (2)	Test 2_report (1)	Test 1_report (3)	Test 1_report (2)
Test Time	2025-11-10 11:04:08	**2025-11-06 16:33:16**	**2025-11-06 14:34:26**	**2025-11-05 18:25:09**	**2025-11-05 16:26:21**	**2025-11-05 14:27:34**	**2025-11-05 12:28:46**	**2025-11-05 10:29:52**	**2025-11-03 16:59:57**	**2025-11-03 15:01:09**
Test Duration	01:58:44	**01:58:38**	**01:58:41**	**01:58:37**	**01:58:36**	**01:58:36**	**01:58:40**	**01:58:45**	**01:58:39**	**01:58:39**
BenchSEE Version	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2	2.1.2
CPU Model	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz	Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz
CPU Features	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled	4200.0000, SuperFast mode disabled

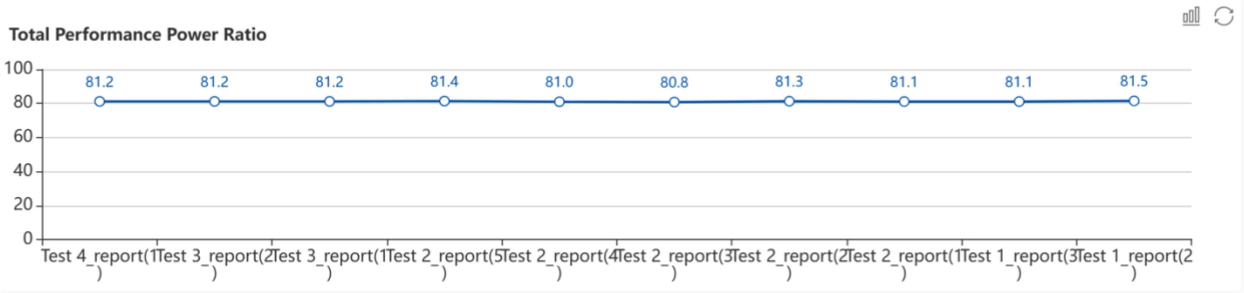
02 Performance Power Ratio Comparison of Each Component

Component	Test 4_report (1)	Test 3_report (2)	Test 3_report (1)	Test 2_report (5)	Test 2_report (4)	Test 2_report (3)	Test 2_report (2)	Test 2_report (1)	Test 1_report (3)	Test 1_report (2)
CPU	148.9	148.6	149.1	**149.4** Max	149.0	147.8	148.8	148.7	148.2	149.2
Memory	16.6	16.6	16.5	16.6	16.4	16.6	**16.7** Max	16.5	16.6	**16.7** Max
Storage	424.4	423.6	**425.5** Max	424.4	420.9	425.0	423.7	424.3	424.6	424.4
Total Performance Power Ratio	81.2	81.2	81.2	81.4	81.0	80.8	81.3	81.1	81.1	**81.5** Max









Appendix C

Description of workloads used in testing:

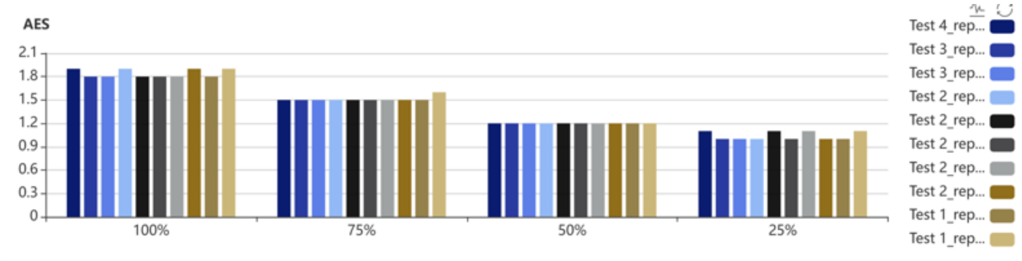
1- The workloads of CPU components under test are AES, COMPRESS, LU, OLTP, SHA256, SOR, and SORT;
2- The workloads of memory components under test are CACHE and STREAM.
3- The workloads of storage components under test are RANDOM and SEQUENTIAL.
4- The IDLE workload indicates the server's idle load, which is used to measure the server's power consumption when the server is idle.

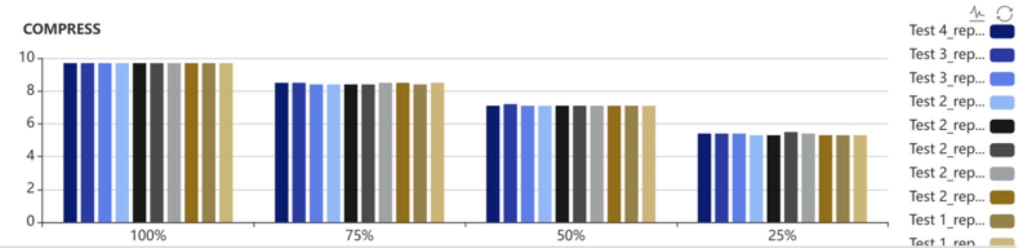
Energy Efficiency Ratio Comparison of Each Workload

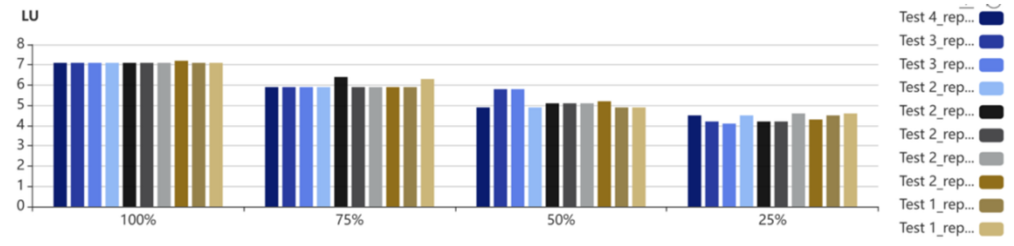
Workload	Level	Test 4_report (1)	Test 3_report (2)	Test 3_report (1)	Test 2_report (5)	Test 2_report (4)	Test 2_report (3)	Test 2_report (2)	Test 2_report (1)	Test 1_report (3)	Test 1_report (2)
AES	100%	1.9	1.8	1.8	1.9	1.8	1.8	1.8	1.9	1.8	1.9
AES	75%	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.6
AES	50%	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2
AES	25%	1.1	1.0	1.0	1.0	1.1	1.0	1.1	1.0	1.0	1.1
COMPRESS	100%	9.7	9.7	9.7	9.7	9.7	9.7	9.7	9.7	9.7	9.7
COMPRESS	75%	8.5	8.5	8.4	8.4	8.4	8.4	8.5	8.5	8.4	8.5

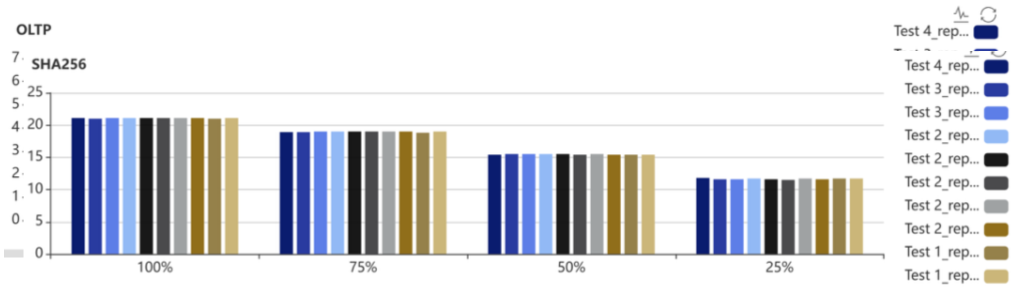
Workload	Level	Test 4_report (1)	Test 3_report (2)	Test 3_report (1)	Test 2_report (5)	Test 2_report (4)	Test 2_report (3)	Test 2_report (2)	Test 2_report (1)	Test 1_report (3)	Test 1_report (2)
COMPRESS	50%	7.1	7.2	7.1	7.1	7.1	7.1	7.1	7.1	7.1	7.1
COMPRESS	25%	5.4	5.4	5.4	5.3	5.3	5.5	5.4	5.3	5.3	5.3
LU	100%	7.1	7.1	7.1	7.1	7.1	7.1	7.1	7.1	7.2	7.1
LU	75%	5.9	5.9	5.9	5.9	6.4	5.9	5.9	5.9	6.3	5.9
LU	50%	4.9	5.8	5.8	4.9	5.1	5.1	5.1	5.2	4.9	4.9
LU	25%	4.5	4.2	4.1	4.5	4.2	4.2	4.6	4.3	4.5	4.6
OLTP	100%	6.2	5.8	6.1	6.2	6.2	6.2	6.2	6.2	6.2	6.1
OLTP	87.5%	5.5	5.5	5.5	5.5	5.5	5.6	5.5	5.6	5.6	5.5
OLTP	75%	4.9	4.8	4.9	4.9	4.9	4.9	4.9	4.9	4.9	4.9
OLTP	62.5%	4.3	4.2	4.3	4.3	4.3	4.3	4.3	4.4	4.3	4.3
OLTP	50%	3.7	3.6	3.7	3.7	3.7	3.7	3.7	3.7	3.6	3.7
OLTP	37.5%	3.3	3.0	3.0	3.4	3.4	3.3	3.3	3.3	3.4	3.1
OLTP	25%	2.2	2.2	2.3	2.7	2.4	2.4	2.4	2.3	2.4	2.2
OLTP	12.5%	1.7	1.7	1.7	1.7	1.8	1.7	1.7	1.8	1.8	1.7
SHA256	100%	21.1	21.0	21.1	21.1	21.1	21.1	21.1	21.1	21.0	21.1
SHA256	75%	18.9	18.9	19.0	19.0	19.0	19.0	19.0	19.0	18.8	19.0
SHA256	50%	15.4	15.5	15.5	15.5	15.5	15.4	15.5	15.4	15.4	15.4
SHA256	25%	11.8	11.6	11.6	11.7	11.6	11.5	11.7	11.6	11.7	11.7
SOR	100%	8.8	8.8	8.8	8.8	8.8	8.8	8.8	8.8	8.8	8.8
SOR	75%	8.6	8.5	8.6	8.5	8.5	8.7	8.6	8.5	8.6	8.5
SOR	50%	7.3	7.4	7.3	7.3	7.3	7.3	7.4	7.3	7.2	7.3
SOR	25%	5.3	5.4	5.3	5.3	5.4	5.4	5.3	5.4	5.3	5.3
SORT	100%	5.8	5.9	5.8	6.0	5.7	5.8	5.7	5.8	5.9	5.9
SORT	75%	4.8	4.9	4.8	5.0	4.7	4.8	4.8	4.9	4.9	4.9
SORT	50%	3.7	3.8	4.2	3.8	4.1	3.7	3.7	3.8	3.8	3.8
SORT	25%	2.8	2.8	2.6	2.9	2.6	2.4	2.6	2.7	2.6	2.8
CACHE	high	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
CACHE	low	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
STREAM	100%	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3
STREAM	50%	1.7	1.7	1.7	1.7	1.7	1.7	1.7	1.7	1.7	1.7
RANDOM	100%	25.7	25.4	25.5	25.6	25.5	25.6	25.5	25.6	25.4	25.5
RANDOM	50%	12.9	12.8	12.8	12.9	12.8	12.8	12.8	12.8	12.8	12.8

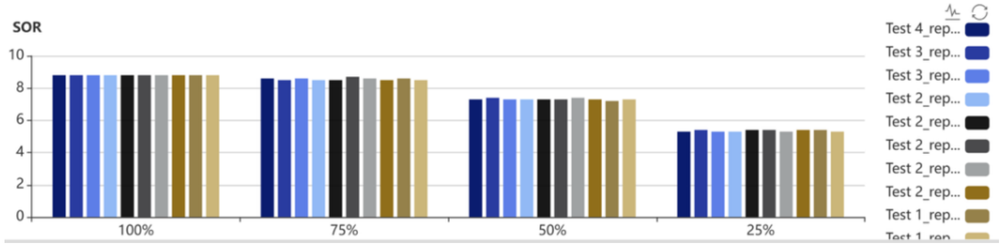
Workload	Level	Test 4_report (1)	Test 3_report (2)	Test 3_report (1)	Test 2_report (5)	Test 2_report (4)	Test 2_report (3)	Test 2_report (2)	Test 2_report (1)	Test 1_report (3)	Test 1_report (2)
SEQUENTIAL	100%	18.0	18.0	18.1	18.0	17.8	18.0	18.0	18.0	18.1	18.1
SEQUENTIAL	50%	9.2	9.3	9.3	9.2	9.1	9.3	9.2	9.2	9.3	9.2

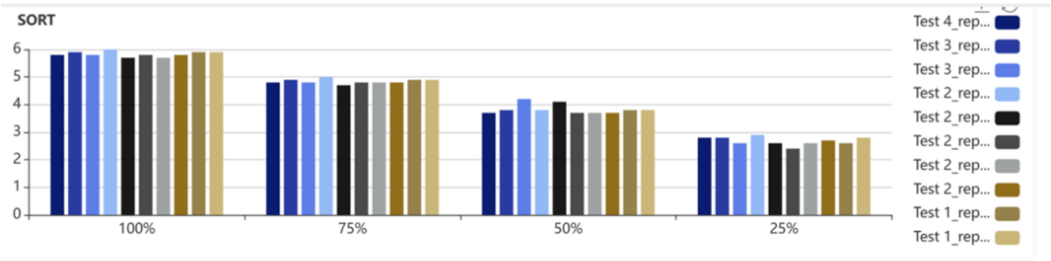


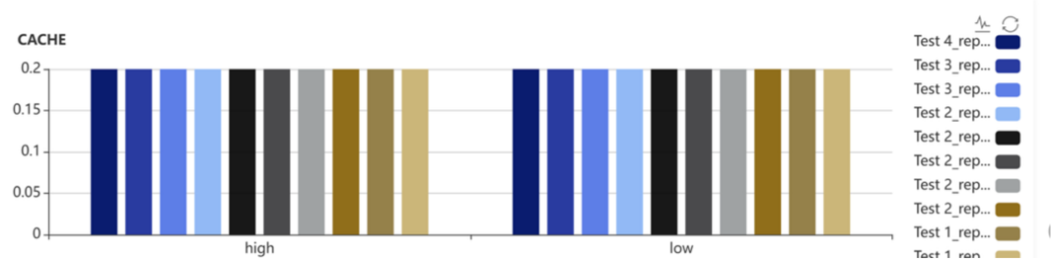


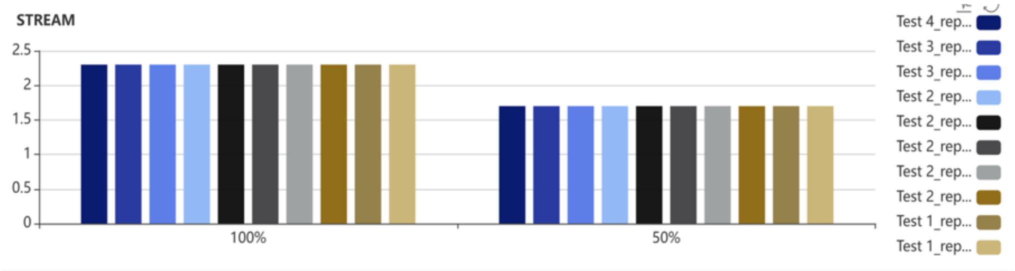


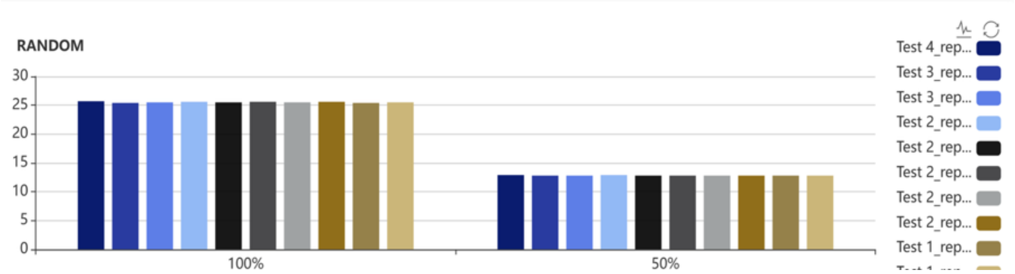


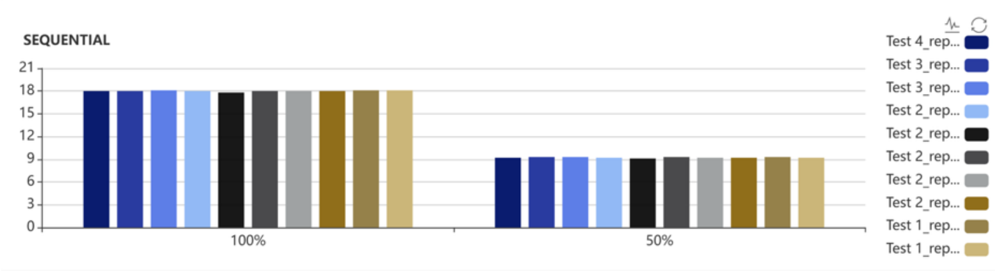



SORT
0
1
2
3
4
5
6
100%
75%
50%
25%
Test 4_rep...
Test 3_rep...
Test 3_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 1_rep...
Test 1_rep...


CACHE
0
0.05
0.1
0.15
0.2
high
low
Test 4_rep...
Test 3_rep...
Test 3_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 1_rep...
Test 1_rep


STREAM
0
0.5
1
1.5
2
2.5
100%
50%
Test 4_rep...
Test 3_rep...
Test 3_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 1_rep...
Test 1_rep...


RANDOM
0
5
10
15
20
25
30
100%
50%
Test 4_rep...
Test 3_rep...
Test 3_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 1_rep...


SEQUENTIAL
0
3
6
9
12
15
18
21
100%
50%
Test 4_rep...
Test 3_rep...
Test 3_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 2_rep...
Test 1_rep...
Test 1_rep...