



Cyprus
University of
Technology

Department of Electrical
Engineering and Computer
Engineering and Informatics

Master thesis

KWEST: Kubernetes Workload Simulator and Re- play Tool

Matthew Paraskeva

Limassol, May 2026

CYPRUS UNIVERSITY OF TECHNOLOGY

Faculty of Engineering and Technology

Department of Electrical Engineering and Computer Science and Engineering

Master thesis

KWEST: Kubernetes Workload Simulator and Replay Tool

Matthew Paraskeva

Advisor: Herodotos Herodotou

Dr Herodotos Herodotou

Limassol, May 2026

Copyrights

Copyright © 2026 Matthew Paraskeva

All rights reserved.

The approval of the dissertation by the Department of Electrical Engineering and Computer Science and Engineering does not necessarily imply the approval by the Department of the views of the writer.

Acknowledgements

First and foremost, I would like to thank my advisor, Dr. Herodotos Herodotou, for their invaluable guidance, continuous support, and patience throughout the entire pre-development and development phase. Their insights and expertise were crucial in shaping this project.

Lastly, I wish to thank my family and friends for their unwavering encouragement and understanding.

ABSTRACT

Evaluating Kubernetes scheduling behavior for different workloads is costly and difficult to reproduce on real clusters due to infrastructure overhead, environmental noise, and inconsistent initial conditions. Although simulation provides controlled execution, simulator-based workflows commonly rely on manual file handling, incomplete provenance capture, and fragmented result inspection, which limits repeatability and auditability.

This thesis extends the KWEST simulator into a practical experimentation platform for Kubernetes-oriented workload studies. The platform introduces a workload pipeline that supports both synthetic scenario generation and trace-driven replay derived from Kubernetes pod lifecycle events, together with a cluster import mechanism that exports and ingests Kubernetes node snapshots reflecting scheduler-visible resources (e.g. allocatable CPU and memory). A web-based interface integrates configuration creation and validation, run execution, and run history navigation. To support repeatability and controlled comparisons, the platform persists run metadata, referenced inputs, and produced artifacts in Couchbase under a stable `runId`, using `gzip+base64` encoding to reduce storage overhead, while report views provide structured access to key simulator outputs.

A scenario-based evaluation assesses trace correctness, artifact completeness, reproducibility under fixed inputs, controlled cross-scheduler comparisons, and runtime/storage overhead. Results show that Kubernetes-derived inputs can be captured and reused consistently, that run provenance is sufficient to repeat experiments, and that orchestration overhead remains modest relative to simulator runtime, with compression substantially reducing stored artifact size.

Keywords: Kubernetes, Scheduling, Simulation, Trace Replay, Provenance, Node Snapshots, Reproducibility