

Informatics Institute of Technology

In Collaboration With

University of Westminster, UK

“MicroWise”

Autonomic Java Virtual Machine Performance Optimization
Framework for Microservices

A dissertation by

Sri Ramachandra Barathi Namage Suleka Helmini

Supervised By

Mr. Saman Hettiarachchi

Submitted in partial fulfilment of the requirements for the
B. Eng (Hons) Software Engineering
Department of Computing

May 2020

© The copyright for this project and all its associated products resides with Informatics
Institute of Technology.

Abstract

Microservices is an architectural style that was largely inspired by service-oriented computing which was first formally introduced in 2011. It has since been widely adopted in the software development domain. From the different technologies utilized in building Microservices, JVM-based Microservices make up the majority of the domain. Considering performance in Microservices, the latency introduced by inter-communication and latency introduced by in-memory processing are considered as the two main factors that affect their performance. Among these two factors, this research focuses on mitigating performance issues stemming from in-memory processing in Microservices.

In mitigating in-memory processing related performance issues, we focus on optimizing the performance of the immediate underlying layer of a JVM-based Microservice, which as the name suggests is the Java Virtual Machine (JVM). When it comes to the JVM, poor Garbage Collection (GC) is one of the main causes of its performance degradation. However, manual optimization of JVM parameters that affect GC is very time consuming and tedious due to the overwhelming number of parameter configurations that can be defined. Also, to perform this, an individual requires a good understanding of the parameters and their value bounds. Hence, we propose MicroWise, a framework employing a unique optimization pipeline in optimizing JVM parameters that affect GC with the end goal of optimizing the performance of a given Microservice. In the introduced framework we automate the complete process of running performance tests, identifying best performing GC algorithm for the Microservice, optimizing JVM parameters that affect the relevant GC algorithm and finally produce a JVM parameter configuration based on GC that can improve the performance of the Microservice. Our work is the first of its kind that provides an automated solution in obtaining JVM parameter configurations that affect GC tailored to optimize the performance of a given Microservice. Whereas other work mostly has contributed towards optimizing JVM parameters that affect its different components in optimizing various monolith applications. Using MicroWise we were able to achieve an impressive average performance improvement of 45% in a memory-bound use case, and a 26% and a 46% improvement in CPU bound and I/O bound use cases respectively. With our solution, we strive to make the JVM optimization process in Microservices more accessible, convenient, reliable and efficient.

Keywords: Performance, Microservices, Java Virtual Machine, Performance optimization