INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER, UK

University of Westminster, Coat of Arms

Evaluation of Caching Strategies for RESTful API/s and Proposing a Hybrid Caching Strategy based on Spring Boot & Redis

A dissertation by

Mr. J. A. C. L. Rajapakse

w1762231 / 2019515

Supervised by

Ms. Malsha Fernando

September 2023

Submitted in partial fulfillment of requirements for the MSc in Advanced Software Engineering degree at the University of Westminster.

Abstract

The discrepancy between processor speed and disk access time is continuing to expand as a result of the ongoing, significant rise in processor speeds and the relatively consistent disk speed. System applications with frequent memory access and memory-intensive programs are heavily impacted by this factor and their execution times are barred. Integration of cache memory and the execution of smart strategies to utilize it would address this disparity of memory access speed and I/O latency. Cache memory access mechanism/s are also dependent on the nature of the application for which it is developed. Therein the selection and development of an appropriate caching strategy and cache-access mechanism is crucial for the execution performance of almost all software applications.

This dissertation is the result of development and fine-tuning of a hybrid caching strategy known as the ‘HRA-Caching strategy’ built with the intention of decreasing application execution time of memory-intensive application software. The developed prototype system is integrated with the novel hybrid caching strategy to measure and showcase the performance in comparison to traditional caching mechanisms. This new caching algorithm is a fusion of horizontally scaled caching strategies to provide reduced latency for frequently accessed data fragments by the application.

HRA-Caching strategy and the supported microservice system APIs in telecommunication domain are novel outcomes produced in this research. It performs better with read-intensive applications and performs equally or slightly higher with write-intensive systems in comparison to existing algorithms. The developed system is integrated with a graphical user interface to support the user by eliminating the requirement of handling the logic behind. It is available in Java Spring Boot framework with Redis in-memory cache database support and is extensible for new applications as a plug-n-play component once it is production ready.

Keywords:

Microservices, Cache, Redis, SQL, Spring Boot, Node.js

Subject Descriptors:

CCS → Information systems → Information retrieval → Retrieval models and ranking → Combination, fusion, and federated search