Performance Benchmarking for Resource Allocation Optimization in GeoNode Ecosystems on Kubernetes Clouds

Marcel Wallschläger, Igo Silva de Almeida, Xenia Specka

Leibniz Centre for Agricultural Landscape Research (ZALF)

FOSS4G BELÉM,2024

Data Analysis & Simulation

Abstract

This study addresses the increasing demand on Spatial Data Infrastructures (SDI) for scalability, reliability, and performance by migrating GeoNode environments to Kubernetes-based cloud systems. It proposes performance benchmarking and resource optimization strategies using GeoNode-benchmark, a load-testing tool developed with Locust, to optimize CPU and memory utilization, minimize costs, and enhance reliability.

Introduction and Context

- **Spatial Data Infrastructures (SDIs)** facilitate access to and management of geospatial data for public, governmental, and research use cases.
- Growing data volumes, such as those generated by the Sentinel-2 system (1.6 TB/day), highlight the need for scalable solutions.
- Kubernetes offers dynamic resource allocation but poses challenges related to cost and configuration.
- **GeoNode**, an open-source geospatial data management platform, serves as a testbed for cloud optimization.

Objectives

- Optimize resource usage for GeoNode in Kubernetes.
- Develop benchmarking strategies to provide cost-effective, reliable, and scalable resource allocation tailored to real-world user scenarios.

Methodology

- **GeoNode-benchmark Tool:** Based on Locust, simulates user interactions (uploading/viewing data, metadata access).
- **Kubernetes Setup:** GeoNode deployed in low, base, and high resource configurations.
- **Components Evaluated:** GeoServer, Celery, GeoNode web interface, Postgres database, and others.
- **Load Simulation:** Incremental user load testing from 0 to 200 concurrent users.

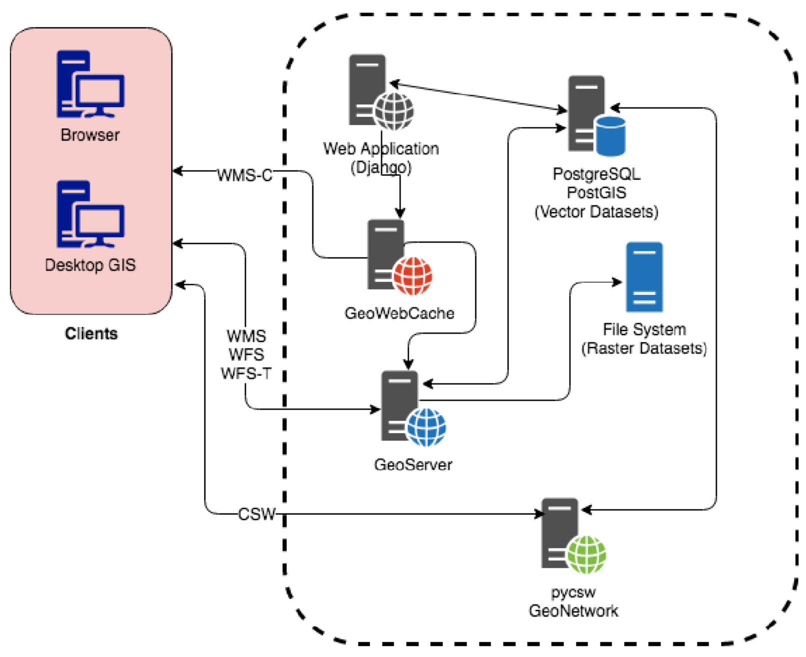

Figure 1: GeoNode components overview, published in [4]

component	Memory Low | Base | High	CPU cores Low | Base | High
GeoNode	4096Mi, 8192Mi, 16384Mi	2, 4, 8
Celery	1024Mi, 2048Mi, 4096Mi	1, 2, 4
GeoServer	4096Mi, 8192Mi, 16384Mi	2, 4, 8
Postgres	1024Mi, 2048Mi, 4096Mi	1, 2, 4
Pycsw	1024Mi, 2048Mi, 4096Mi	1, 1, 2
Nginx	1024Mi, 1024Mi, 2048Mi	1, 1, 2
Rabbitmq	1024Mi, 1024Mi, 2048Mi	1, 1, 2
Memcached	1024Mi, 1024Mi, 2048Mi	1, 1, 2

Table 1: resources limits and requests of GeoNode Kubernetes deployment

Results and Discussions

- **Memory Utilization:** GeoServer reached memory limits under high load, causing system instability (out-of-memory restarts). Memory allocation needs optimization to prevent service disruption.
- **CPU Utilization:** CPU consumption was relatively stable across configurations. Celery and GeoNode components required the most CPU resources.
- **Load Testing Insights:** Demonstrated bottlenecks in memory allocation and highlighted areas for resource tuning.
- **Resource Allocation:** Recommended balanced memory provisioning to maintain performance without over-provisioning.

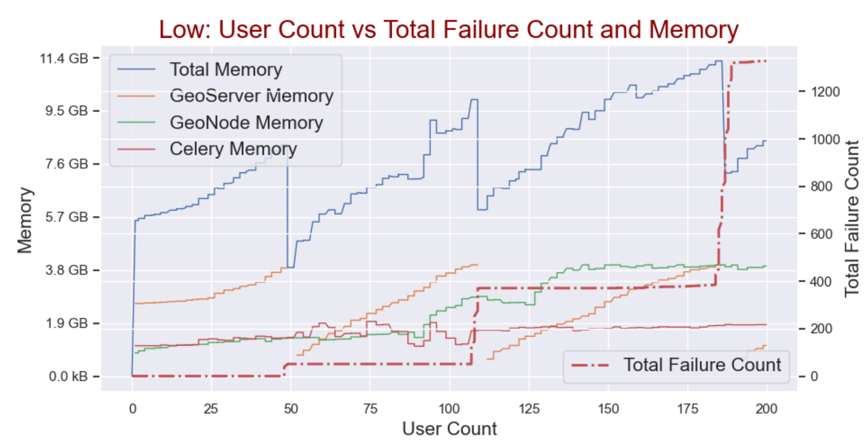

Figure 2: Memory usage of components and total failure count for resource configuration: Low

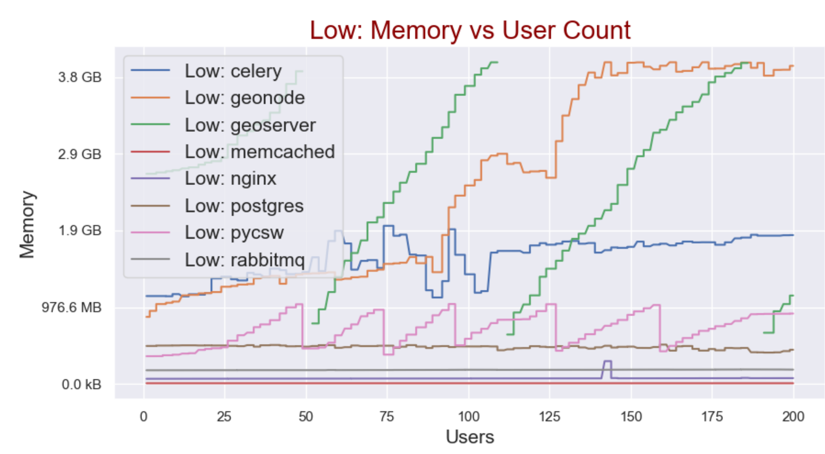

Figure 3: Memory usage of components for resource configuration: Low

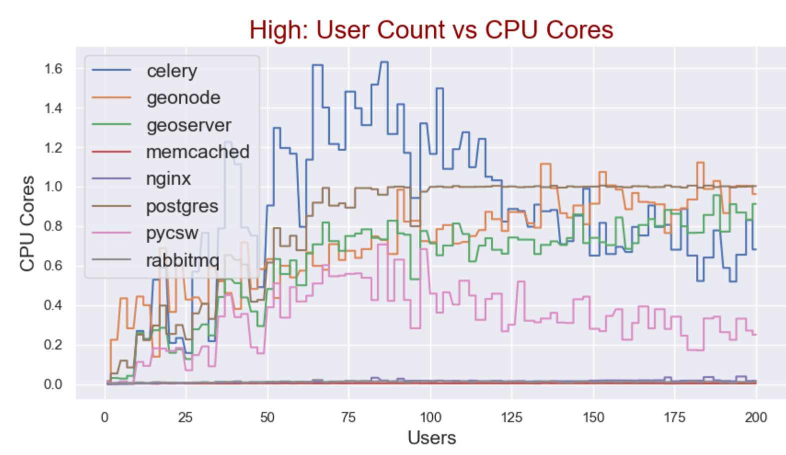

Figure 4: CPU Cores usage of components for configuration High

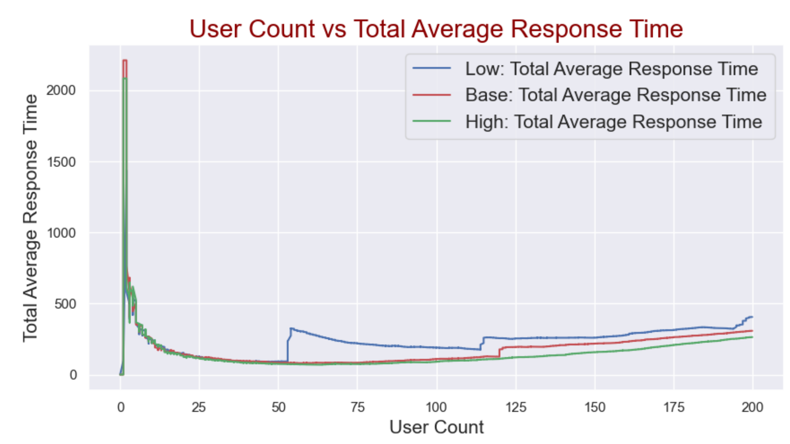

Figure 5: Average Response Time between configurations: Low, Base and High

Conclusions and Recommendations

- Optimize GeoServer memory configuration to avoid OOM issues.
- Ensure efficient CPU allocation, focusing on critical components (e.g., Celery, GeoNode).
- Continued load testing using tools like Locust is key to dynamic resource management and cost control.

Future Work

- Expand benchmarking tasks to include complex user interactions (e.g., metadata editing, data harvesting).
- Integrate real-time monitoring with Prometheus for dynamic adjustment of resource allocation.
- Evaluate scalability of GeoNode deployments, refining auto-scaling policies to respond to fluctuating workloads.

References

1. Copernicus Sentinel-2 Data, European Space Agency, 2021.
2. GeoNode Documentation and Development Team, 2024.
3. Wallschläger et al., GeoNode-k8s Development, 2024.
4. Balbo, S., Boccardo, P., Dalmasso, S., & Pasquali, P. (2014). A public platform for geospatial data sharing for disaster risk management. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 40, 189-195.

Leibniz Centre for Agricultural LandscapeResearch (ZALF) | Eberswalder Straße 84 | 15374 Müncheberg| Germany
Contact: Igo Silva de Almeida [igo.silvadealmeida@zalf.de]
Date: 04.12.2024

Federal Ministry of Food and Agriculture
LAND BRANDENBURG
Leibniz Association