Improve Python Performance with AWS C5 Instances Featuring Intel® Xeon® Scalable Processors and Granulate

Python

Boost Python Performance by 9% by Adding Granulate to AWS c5.large Instances Featuring Intel Xeon Scalable Processors

vs. c5.large Instances without Granulate

Boost Python Performance by 37% by Selecting AWS c5.large Instances Featuring Intel Xeon Scalable Processors with Granulate

vs. c6g.large Instances with Graviton Processors

Using Intel Xeon Scalable Processors with Granulate Real-Time Continuous Optimization Shrinks Execution Time Compared to C6g Instances with Graviton Processors

By choosing cloud instances that compile and run Python scripts faster, your organization can improve overall application performance. We used the PyPerformance benchmark to test the Python performance that multiple Amazon Web Services (AWS) EC2 cloud instance types can deliver. PyPerformance is designed to showcase Python performance, detect performance regression in Python implementations, and validate optimization changes.

First, we tested C5 instances featuring Intel Xeon Scalable processors two ways: with and without Granulate Real-Time Continuous Optimization (Granulate). Granulate offers "autonomous, continuous workload optimization," which, according to Granulate, "improves application performance and cuts costs by up to 63%."[i] Next, we compared the C5 instance with Granulate to a C6g instance using Graviton processors.

The C5 instance with Granulate outperformed both its C5 counterpart without Granulate and the C6g instance with Graviton processors. These results suggest that businesses can enjoy greater performance for Python implementations by choosing C5 instances enabled by Intel Xeon Scalable processors with Granulate.

Granulate Improved PyPerformance

As Figure 1 shows, on c5.large instances, using Granulate improved execution speed of the PyPerformance benchmark by 9%.

Relative Performance of AWS C5 Instances with and without Granulate
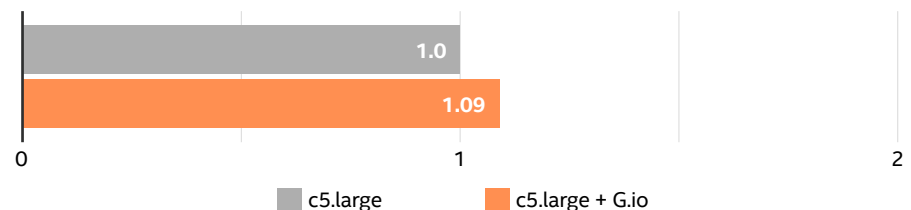
Execution time | Higher is better

Figure 1. Relative PyPerformance execution time of c5.large instances, featuring Intel Xeon Scalable processors, both with and without Granulate. Higher is better.

Intel Workload Proof Series: PyPerformance on AWS C5 vs. Graviton

See backup for workloads and configurations. Results may vary.

A C5 Instance with Granulate Outperformed a Graviton Processor-Based C6g Instance

As Figure 2 shows, the c5.large instance enabled by Intel® Xeon® Scalable processors with Granulate delivered 37% better performance than the c6g.large instance based on Graviton processors.
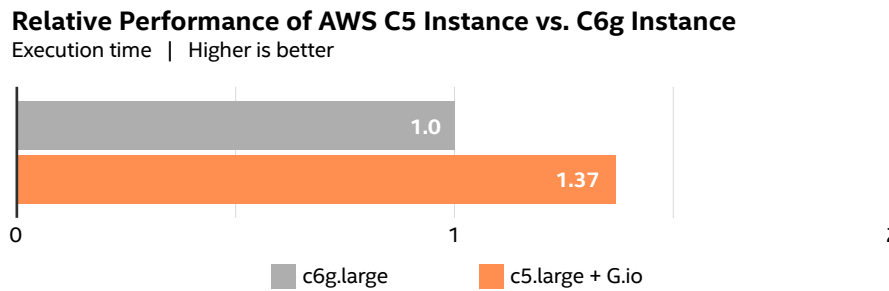
Relative Performance of AWS C5 Instance vs. C6g Instance
Execution time | Higher is better

Figure 2. Relative PyPerformance execution time of a c5.large instance with Intel Xeon Scalable processors and Granulate vs. a c6g.large instance with Graviton processors. Higher is better.

Conclusion

The faster your Python scripts run, the better performance your applications will have. In tests, an AWS C5 instance featuring Intel Xeon Scalable processors with Granulate Real-Time Continuous Optimization outperformed the same instance without Granulate as well as an AWS C6g instance based on Graviton processors. Adding Granulate increased performance by 9% on the C5 instance we tested, and the Granulate-enabled C5 instance outperformed the C6g instance we tested by 37%.

Learn More

To begin running your Python workloads on Amazon C5 instances with Intel Xeon Scalable processors with Granulate, visit https://aws.amazon.com/ec2/instance-types/c5/.