Algorithms and code

performance optimization

About

Wave Access is a results focused software development company that provides high quality software outsourcing services to hundreds of emerging and established companies globally. We use our technical expertise to increase business efficiencies, optimize slow or unreliable systems, recover projects that have gone off track and bring ambitious ideas to life.

18
years of delivering successful outcomes for customers

250+
talented & passionate professionals

4
global R&D centers

9
industry verticals from banking to healthcare

280+
successful projects delivered and counting

72%
of our customers are repeat business

Las Vegas
headquarters

USA, UK, Denmark and Easten Europe
sales offices

Microsoft SILVER CERTIFIED Partner

ISO 9001 CERTIFIED

Microsoft Partner Network™

2011 PARTNER OF THE YEAR
Microsoft Dynamics Professional Services
CRM4Legal for Client Profiles
Winner

Microsoft Partner
Microsoft

2017 Partner of the Year Winner
Business Analytics Award

Academy Award-winning
Mocha for Imagineer Systems

Project Overview

Algorithms performance tuning projects appear when the code complexity and data flow size become an issue to achieve results within a reasonable time. The most popular technologies that we tune codebase on:

- Java
- R
- Python
- .Net
- C++
- SQL

Test Coverage

One major aspect of the WaveAccess delivery process that we follow is to cover the code with tests for every case – general, exceptions, extremes, etc. Development team does not start to make any changes until it does not have enough tests to make sure that our optimizations do not affect the algorithm. It is not a rare situation when one the algorithm has more than 100 test data sets. The validation criteria is often a subject to discuss because some algorithms are discrete with strict input and output data, but there are cases with fractions number computing that has epsilons and mistake thresholds.

The tests are implemented as a unit test module and integrated into a continuous integration environment. The test environment is usually a build server like a TeamCity that compiles every commit into a code repository and runs all the tests. This approach helps us find issues at the early stages and to ensure that we can delivery a stable version at any time.

Performance Optimization Options

WaveAccess uses a rich set of options to increase the algorithms computing speed:

1. Change common sub algorithms to be more efficient

As example is to change some part with a sorting code to be a more efficient one – many common algorithms have disadvantages in some special cases that cause computing degradation. Our job is to find the best algorithm that works faster in most cases.

2. Add caching

This is a popular approach to find places in the code that have a good hit cache rate and put the cachable data into memory, shared cache (Redis for example), or disk.

3. Utilize CPU fastest instructions

Processors evaluated enough to perform several instructions per cycle. Sometimes the computing hot spot can be changed to use different data structure and operations that are more native for CPU and can be executed much faster.

4. Utilize memory better

The whole idea of this case is to use cache CPU L1-3 memory cache better. This means that the way code iterates on simple arrays (sequence or randomly) are important for the CPU memory cache hit rates.

5. Multi-threading and clustering

The whole idea of this case is to use cache CPU L1-3 memory cache better. This means that the way code iterates on simple arrays (sequence or randomly) are important for the CPU memory cache hit rates.

6. Increase efficiency on code execution

Based on reverse engineering analysis, it is possible to find places that can be re-implemented better – code style, SQL queries, or layer architecture. After refactoring, the app will have improved SQL queries execution plans, fewer IO operations/queries to the database, fewer code cycles, etc. As a result, a better execution speed is achieved.

7. Porting one faster technology

When it appears that the current technology executes code slower than other ones, in this case it is always a solution to implement a small module on Java/C++ and seamlessly integrate it with R keeping global app interfaces the same.

More details in our article: R performance optimization using Java.

8. Use GPU acceleration

This radical approach can dramatically increase computing speed 6-100 times in some particular cases. The whole idea that >200 weak cores on GPU are much faster than 4-16 core on CPU especially on highly parallelized algorithms with heavy and random memory access. The WaveAccess team uses CUDA SDK and OpenCL for GPU accelerated code implementation.

More details can be found in our article: Breakthrough in CUDA data compression.

The Result

Algorithms performance tuning by the WaveAccess team delivers computation results in a reasonable time and thereby drives and improves the efficiency of processes.

WaveAccess

We Make IT Easy

If you have a project for us, please get in touch

scientific@wave-access.com

Skype: wave_access

+1 818 731-1279

wave-access.com