



Leistungsoptimierung von Code und Algorithmen

Über

WaveAccess ist ein ergebnisorientiertes Softwareentwicklungsunternehmen, das hochwertige Software-Outsourcing-Services Hunderten von aufstrebenden und etablierten Unternehmen weltweit bietet. Wir verwenden unsere technische Kompetenz zur Steigerung von Geschäftseffizienz, zur Optimierung langsamer und unzuverlässiger Systeme, zur Wiederherstellung von in Schieflage geratenen Projekten und zur Verwirklichung ehrgeiziger Ideen.

18

Jahre der Lieferung
erfolgreicher Ergebnisse
für Kunden

300+

talentierte & leidenschaftliche
Fachkräfte in 4 Ländern

4

globale R&D
Zentren und nahezu
jede Technologie

16+

Branchen vom Bankwesen
bis Gesundheitswesen
Verwenden
unsere Lösungen

280+

Erfolgreiche Projekte
wurden geliefert
und umgesetzt

96%

Kunden-
Zufriedenheitsindex

Auszeichnungen und Anerkennungen

Silver
Microsoft
Partner

Microsoft
Partner



2018 Partner of the Year
Artificial Intelligence Award

2017 Partner of the Year
Business Analytics Award



Microsoft
Partner Network

WORLDWIDE
DYNAMICS PARTNER
OF THE YEAR

2009, 2010, 2011 & 2



Academy Award-winning
Mocha for Imagineer Systems



| Standorte



USA
Vertriebsbüros



Dänemark
Vertriebsbüros



Osteuropa
Vertriebsbüros

| Kernkompetenzen



Implementierung von hochbelasteten & hochskalierbaren Systemen



Microsoft Dynamics CRM Entwicklung, Anpassung und Implementierung



Entwicklung von Echtzeit-Kontrolle-Systemen



Workflow and business process automation



Building AI and Machine Learning based systems



Entwicklung von Web- & mobilen Anwendungen



Blockchain-Implementierung



Wiederherstellung von Projekten



Datenimport und Datenmigration



Systemintegration & -optimierung



Aufbau von Systemen auf Basis von AI und des maschinellen Lernens

Projektübersicht

Leistungsoptimierung von Algorithmen ist in den Fällen erforderlich, wenn Codekomplexität und die Datenflussgröße für die Ausgabe von Ergebnissen in einer angemessenen Zeit problematisch werden. Die beliebtesten Technologien, mit denen wir einen Code optimieren, sind: Java, R, Python, .NET, C ++ und SQL.

| Testabdeckung

Eine der Grundregeln der Organisation des Arbeitsprozesses von WaveAccess ist die Testabdeckung des Codes für jeden Fall - allgemeine Fälle, Ausnahmen, extreme Fälle usw. Das Entwicklerteam führt keine Änderungen so lange ein, bis eine ausreichende Anzahl von Tests durchgeführt wird, um sicherzustellen, dass unsere Optimierung die Arbeit des Algorithmus nicht gestört hat. Es kommt nicht selten vor, dass ein Algorithmus mehr als 100 Test-Datensätze erfordert.

Validierungskriterien werden oft diskutiert, weil einige Algorithmen diskret, mit exakten Eingangs- und Ausgangsdaten sind, aber es kommen Berechnungsfälle mit einem flottierenden Punkt vor, die ihre Abweichungen und Fehler haben.

Die Tests werden als Unit-Test-Modul implementiert und in das System der kontinuierlichen Integration integriert.

Ein Testumfeld stellt ein Build-Server dar, wie TeamCity, der jeden Commit in Code-Repository kompiliert und alle Tests aktiviert. Diese Vorgehensweise ermöglicht es uns, Probleme auf frühen Arbeitsphasen zu erkennen und eine stabile Version zu jedem Zeitpunkt sicherzustellen.

| Parameter der Leistungsoptimierung

WaveAccess verwendet eine breite Reihe von Optionen zur Steigerung der Geschwindigkeit von Berechnungsalgorithmen.

1. Änderung gesamter eingesetzter Algorithmen zur Steigerung deren Effizienz

Man kann zum Beispiel den Sorting Code unter Einsatz eines effektiveren Algorithmus ändern – viele allgemeine Algorithmen haben Mängel in einigen besonderen Fällen, was zur Degradation von Berechnungen führt. Unsere Arbeit besteht darin, den besten Algorithmus zu finden, der in den meisten

Fällen schneller arbeitet.

2. Caching hinzufügen

Das ist eine beliebte Vorgehensweise, Stellen im Code zu finden, die eine gute Cache-Trefferrate haben und Caching-Daten in den Arbeitsspeicher zu übertragen, in den gesamten Cache (z.B. Redis) oder in die Disk.

3. Verwendung der schnellsten Befehle des Zentralprozessors

Ein Prozessor ist in der Lage mehrere Befehle pro Zyklus durchzuführen. Manchmal müssen die für den Prozessor nativen Strukturen und Operationen verwendet werden, die wesentlich schneller ausgeführt werden können.

4. Effektivere Nutzung des Arbeitsspeichers

Die Idee dieser Methode besteht in einer effektiveren Nutzung L1-L3-Cache für die CPU. Das bedeutet, dass die Methode, wie der Code die Iteration einfacher arrays (Felder) ausführt (folgerichtig oder zufällig), eine wichtige Bedeutung für die Effizienz des Caches CPU hat.

5. Multithreading und Clustering

Das ist die Methode einer horizontalen Skalierbarkeit der Codeausführung. Multithreading hat seine Spezifik – ein kohärenter Zugang zum Arbeitsspeicher, ein wettbewerblicher Zugang der Flüsse zu den Ressourcen usw.

6. Effizienzsteigerung der Codeausführung

Auf Basis von Reverse Engineering Analysis kann man Stellen herausfinden, die erneut besser implementiert werden können – Code Style, SQL-Anfragen, Schichtenarchitektur. Nach der Refaktorisierung erhält die App einen verbesserten Ausführungsplan der SQL-Anfragen, eine geringere Anzahl von Operationen der Eingabe-Ausgabe/Anfragen an die Datenbank, eine geringere Anzahl des Code-Zyklus usw. Im Ergebnis wird eine höhere Geschwindigkeit der Ausführung erreicht.

7. Portierung einer schnelleren Technologie

Wenn festgestellt wird, dass die aktuelle Technologie den Code langsamer als andere ausführt, kann man als Lösung ein kleines Modul auf Java/C++ implementieren und es mit R einfach integrieren, wobei die gesamte App-Schnittstelle erhalten bleibt.

Mehr Details über diese Vorgehensweise in unserem Artikel

[«R-Leistungsoptimierung»](#)

8. Verwendung der GPU-Beschleunigung

Diese radikale Vorgehensweise kann die Berechnungsgeschwindigkeit in einigen bestimmten Fällen um 6-100 Mal steigern.

Die Hauptidee der Vorgehensweise besteht darin, dass mehr als 200 schwacher Kerne mit GPU wesentlich schneller arbeiten, als 4-16 Kerne des Prozessors, insbesondere auf den hoch parallelisierten

Algorithmen mit schwerem und zufälligem Zugriff zum Arbeitsspeicher. Das Team von WaveAccess verwendet CUDA SDK und OpenCL zur GPU-Beschleunigung der Codeausführung.

| Ergebnis

Das Team von WaveAccess verfügt über eine Reihe von Instrumenten und Praktiken zur effektiven Leistungseinstellung von Algorithmen, was die Lieferung der Berechnungsergebnisse innerhalb angemessener Zeit und die Steigerung der Prozessausführung ermöglicht.



Wenn Sie ein Projekt für uns haben,
kontaktieren Sie uns bitte.

hello@wave-access.com

www.wave-access.com