

Эффективная реализация сопрограмм в управляемой среде исполнения

Евгений Пантелеев

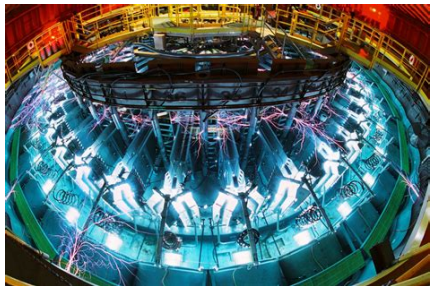
Новосибирский государственный университет

Научный руководитель: Бульонков Михаил Алексеевич,
канд. физ-мат наук
ИСИ СО РАН

Новосибирск
2021г.



(a) Серверы.



(b) Ускорители.

Существует множество задач, в которых необходимо обрабатывать много независимых событий.

- ▶ **Сопрограмма** (с англ. coroutine) - программный модуль, организованный для обеспечения взаимодействия с другими модулями по принципу кооперативной многозадачности.
- ▶ Сопрограммы способны приостанавливать свое выполнение, сохраняя *контекст* (программный стек и регистры), и передавать управление другой.

Плюсы сопрограмм

- ▶ Переключение контекста сопрограммы требует меньше накладных расходов, чем потока.
- ▶ Как правило меньший размер стека, а значит, потребление памяти так же меньше.

Минусы

- ▶ Сопрограммы не способны исполняться параллельно.



(a) C++20.



(b) C#.



(c) Go.

В языке Java сопрограммы не реализованы.

Project Loom

Fibers and Continuations



- ▶ Project Loom – проект на базе OpenJDK, целью которого является разработка сопрограмм для языка Java.
- ▶ На данный момент уже доступна ранняя версия проекта.

Цель: реализация прототипа сопрограмм в Java.

Поставленные задачи:

- ▶ Разработать тесты для сравнения производительности потоков и сопрограмм.
- ▶ Реализовать переключение сопрограмм.
- ▶ Реализовать трассировку ссылок объектов на стеках сопрограмм для сборки мусора.
- ▶ Сравнить производительности сопрограмм и потоков.

Работа проводится на базе Huawei JDK.

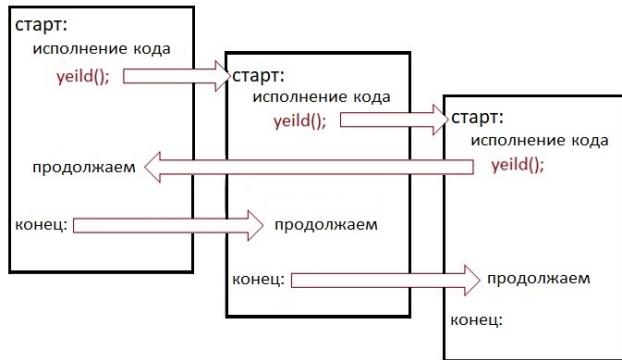
Был создан набор тестов на производительность сопрограмм для языков Go, Java (с “Loom Project”).

Тесты создавались для измерения 2 параметров.

- ▶ Скорость переключения контекста.
- ▶ Потребление памяти.

Репозиторий с тестами: <https://github.com/minium2/coroutines-benchmark>

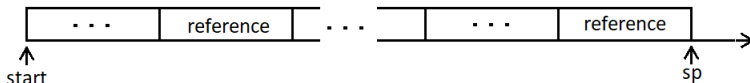
Переключение сопрограмм



Подходы к реализации:

- ▶ OpenJDK(Проект "Loom"): копирование стека сопрограммы при переключении.
- ▶ Go и HuaweiJDK: изменение указателя стека.

Трассировка стеков



- ▶ Для работы сборщика мусора необходимо найти ссылки на стеке.
- ▶ Поиск проходит путем итерации значений на стеке и проверка, что они являются ссылками.

- ▶ Создан набор тестов для сравнения производительности потоков и сопрограмаами.
- ▶ Реализовано переключение контекста сопрограмм.
- ▶ Разработана трассировка ссылок объектов на стеках сопрограмм.
- ▶ Получены результаты тестов производительности.

Результаты: скорости переключения потоков и сопрограмм

Ubuntu, Intel Core i7-8700, 31 Гб ОЗУ, HuaweiJDK

Каждое значение усреднено по 100 измерениям.

Для измерения используется только одно ядро ЦП.

Количество(???), шт.	Число переключений, 1/сек.	
	Сопрограммы	Потоки
100	1'246'756 (-/+ 12'961)	2'306'346 (-/+ 49'831)
1'000	1'199'142 (-/+ 11'803)	2'300'279 (-/+ 27'180)
5'000	1'075'559 (-/+ 59'328)	1'553'872 (-/+ 36'832)
10'000	1'016'802 (-/+ 9'990)	1'015'976 (-/+ 29'096)
50'000	756'523 (-/+ 8'232)	361'088 (-/+ 7'853)

Результаты: скорости переключения сопрограмм в управляемых средах

Ubuntu, Intel Core i7-8700, 31 Гб ОЗУ, HuaweiJDK

Каждое значение усреднено по 100 измерениям.

Для измерения используется только одно ядро ЦП.

ШТ.	Число переключений, 1/сек.		
	HuaweiJDK	OpenJDK("Loom Project")	Go
100	1'246'756 (-/+12'961)	1'900'009 (-/+19'732)	18'187'799 (-/+219367)
1000	1'199'142 (-/+11'803)	1'775'239 (-/+20'491)	17'934'078 (-/+332778)
5000	1'075'559 (-/+59'328)	1'703'631 (-/+30'498)	12'892'417 (-/+339410)
10000	1'016'802 (-/+9'990)	1'924'971 (-/+234'982)	8'307'791 (-/+79652)
50000	756'523 (-/+8'232)	1'518'349 (-/+152'899)	5'292'780 (-/+121844)

Результаты: потребление памяти

Ubuntu, Intel Core i7-8700, 31 Гб ОЗУ

Количество(???), шт.	Размер физической памяти		
	HuaweiJDK	OpenJDK	Go
100	18 Мб	130 Мб	3040 Кб
1000	23 Мб	161 Мб	3105 Кб
5000	30 Мб	187 Мб	3156 Кб
10000	35 Мб	193 Мб	3308 Кб
50000	55 Мб	202 Мб	3407 Кб

Результаты: потребление памяти

Ubuntu, Intel Core i7-8700, 31 Гб ОЗУ, HuaweiJDK

Количество, шт.	Размер физической памяти	
	Сопрограммы	Потоки
100	18 Мб	34 Мб
1000	23 Мб	35 Мб
5000	30 Мб	37 Мб
10000	35 Мб	40 Мб
50000	55 Мб	72 Мб

- ▶ Переделать функцию переключения контекста.
- ▶ Реализация возможности миграции сопрограмм с одного потока на другой(???).
- ▶ Синхронизация: поддержка synchronized блоков(???).
- ▶ Переключение сопрограммы при вызове ввода вывода.

Спасибо за внимание!