



Сборка мусора в JVM Битва алгоритмов

Passion to Perform

Алексей Рагозин

JPoint 2014

Новый претендент в OpenJDK



Shenandoah

Новый алгоритм сборки мусора в OpenJDK

- <http://icedtea.classpath.org/shenandoah/>
- <http://openjdk.java.net/jeps/189>
- <http://rkennke.wordpress.com/>
- <http://christineflood.wordpress.com/>
- <http://rkennke.files.wordpress.com/2014/02/shenandoahtake4.pdf>



Цели

Паузы < 10 ms на куче 100 GiB и больше

- Single space
- Инкрементальный
- Фоновая маркировка
- Фоновое копирование
- SATB барьер
- ~~Таблица карт~~
- ~~Remembered sets~~

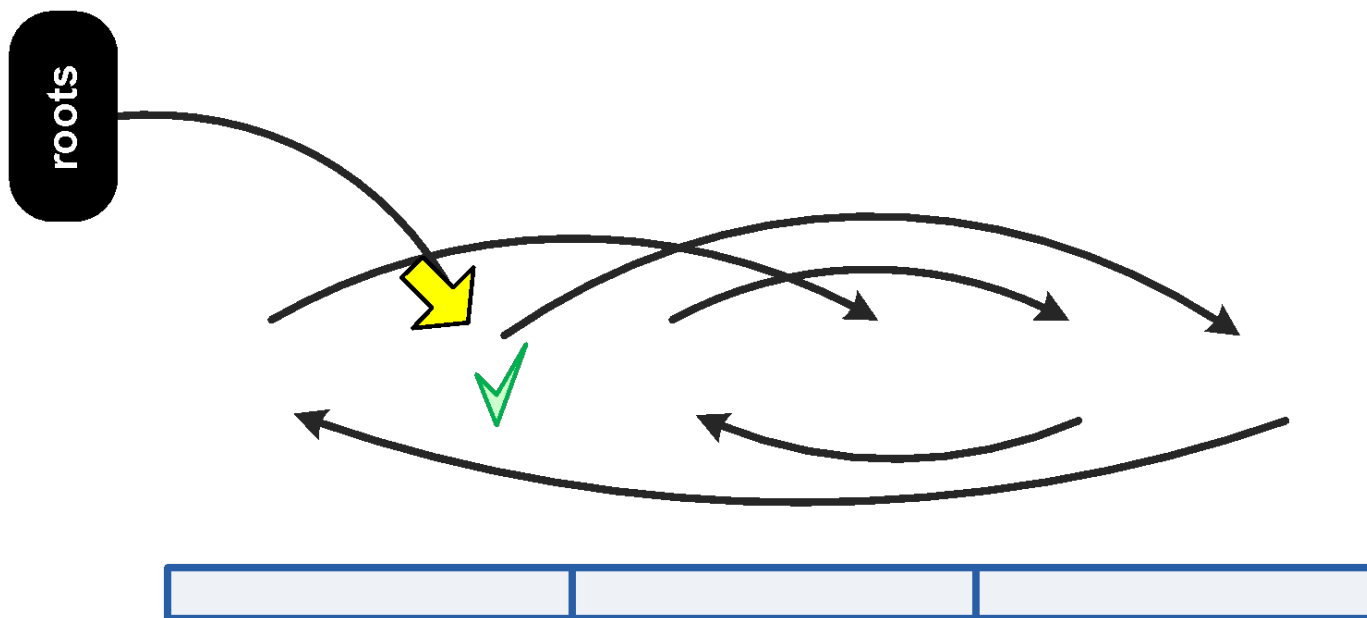


Обоснование

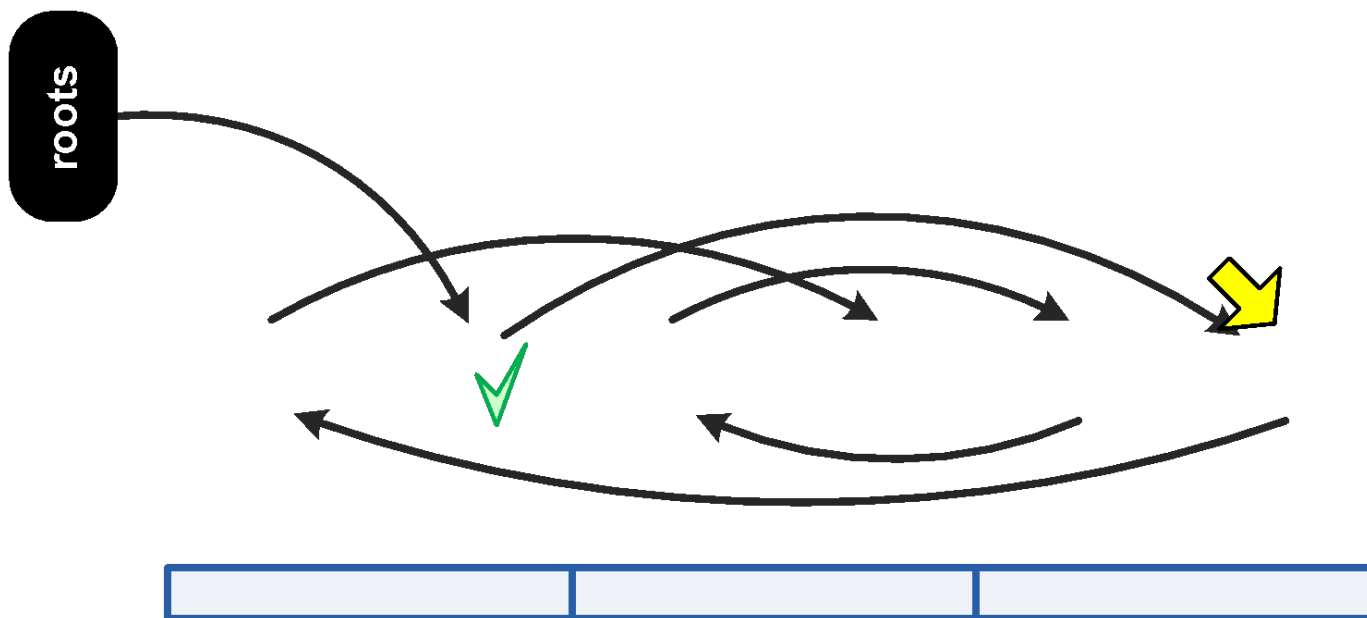
Сборка поколений требует отслеживания
ссылок между поколениями

- Таблица карт – false sharing
- Remembered set – требует много памяти
- Нет поколений - нет проблем

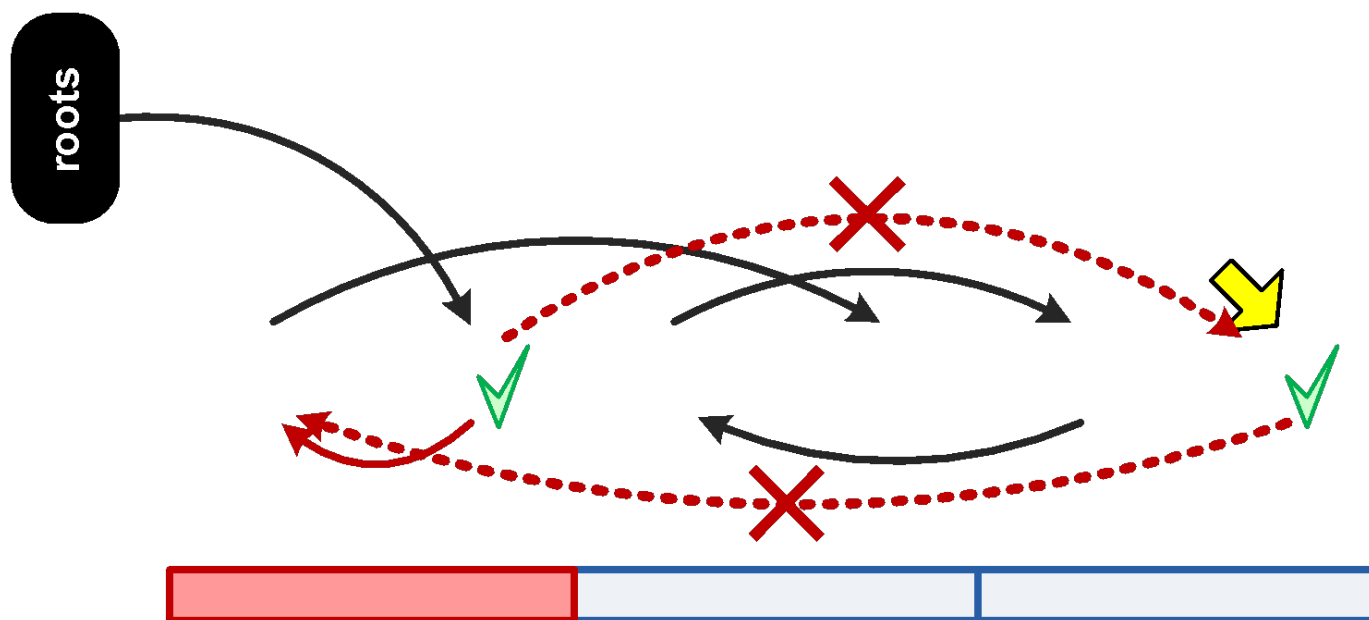
Карточный барьер



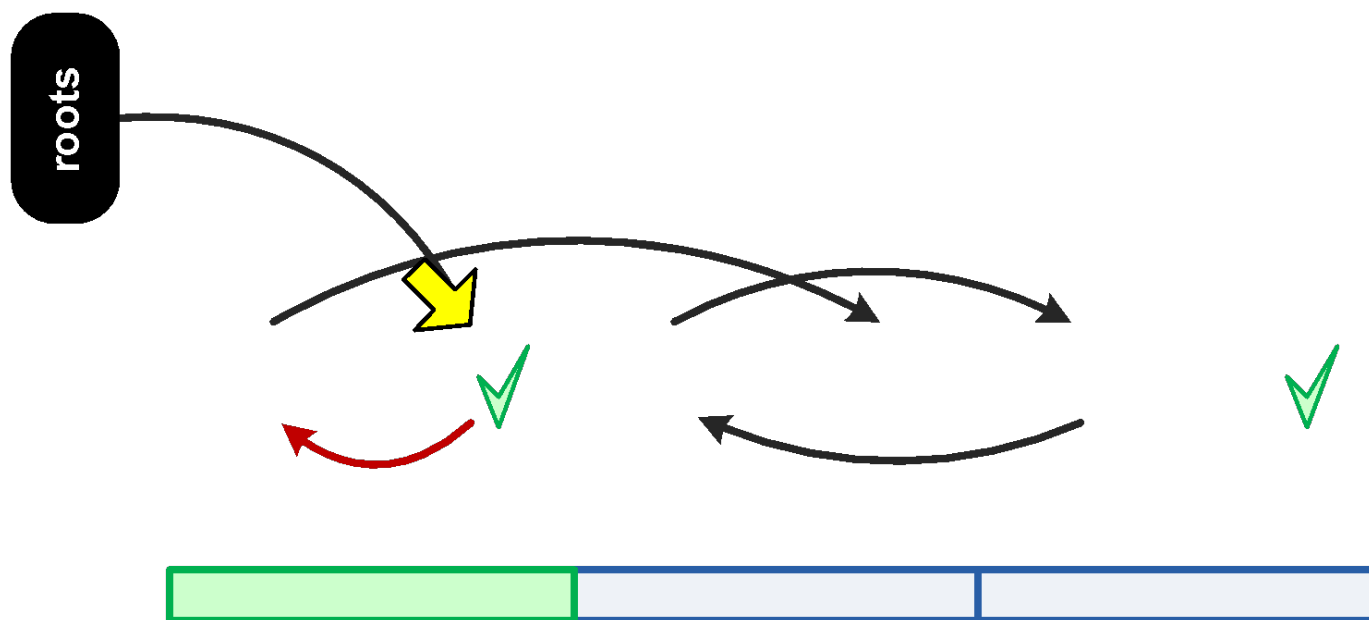
Карточный барьер



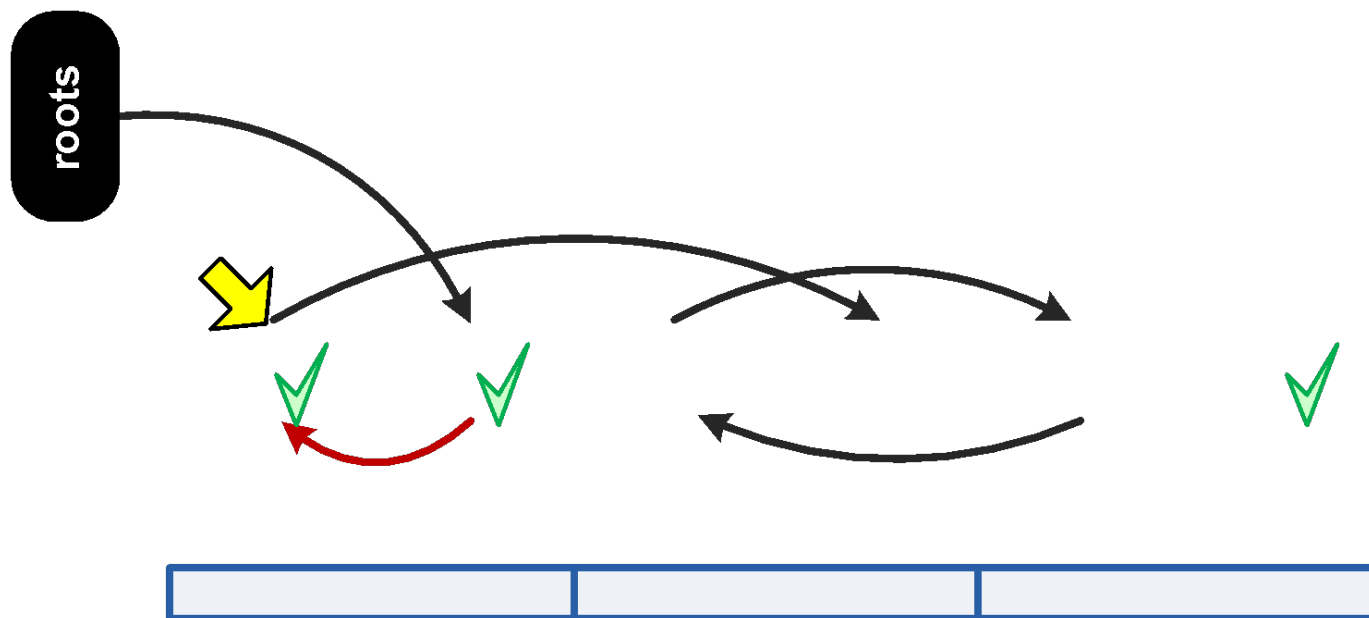
Карточный барьер



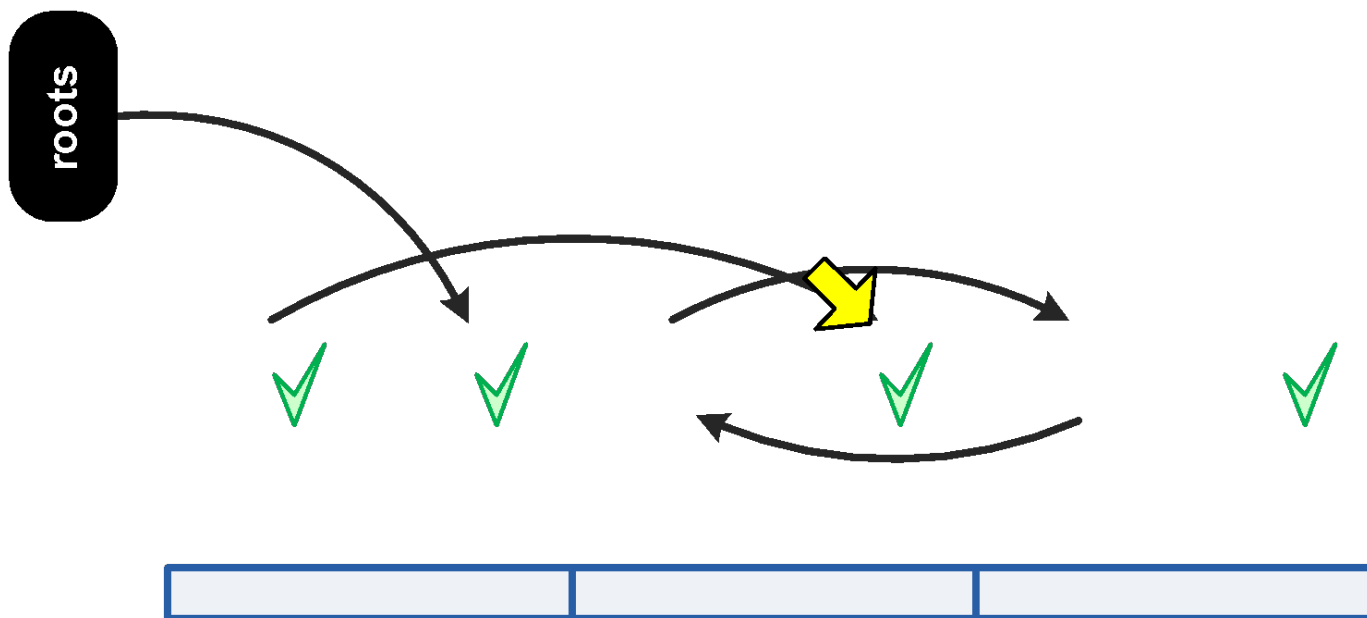
Карточный барьер



Карточный барьер



Карточный барьер



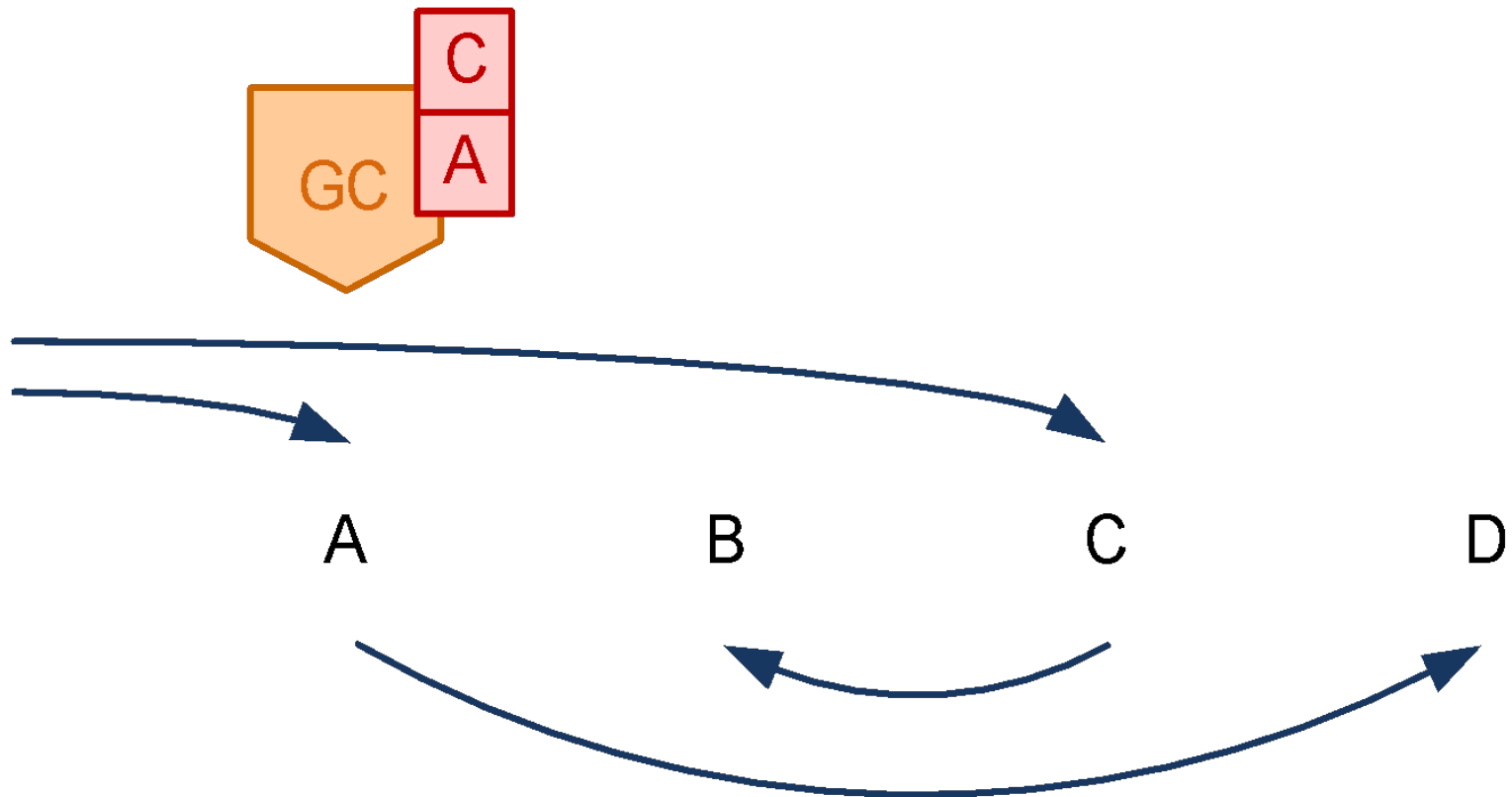


Обоснование

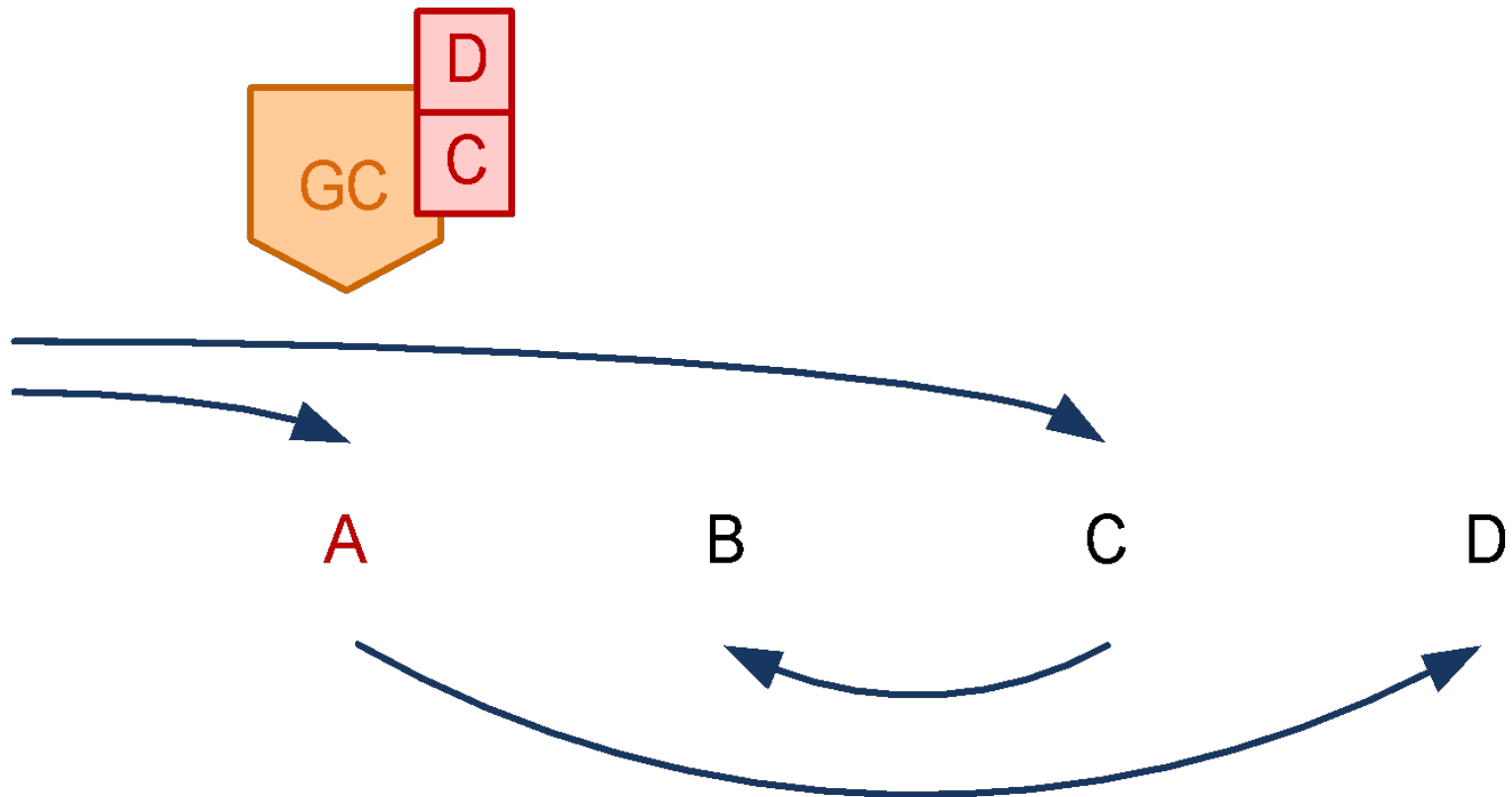
SATB барьер для фоновой маркировки

- Таблица карт
 - false sharing
 - STW пауза для ремаркировки
- SATB – маркировка без пауз

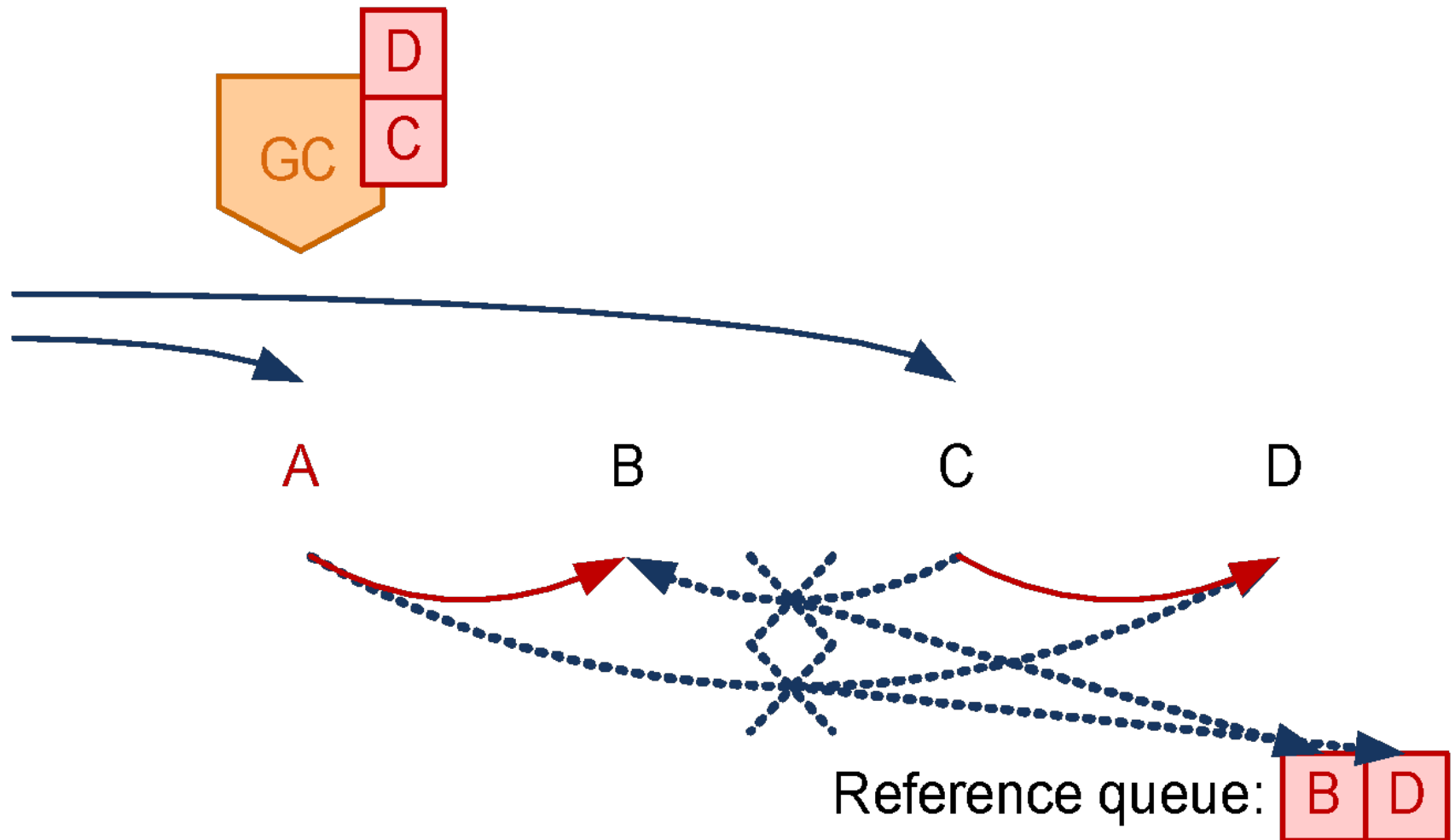
SATV барьер



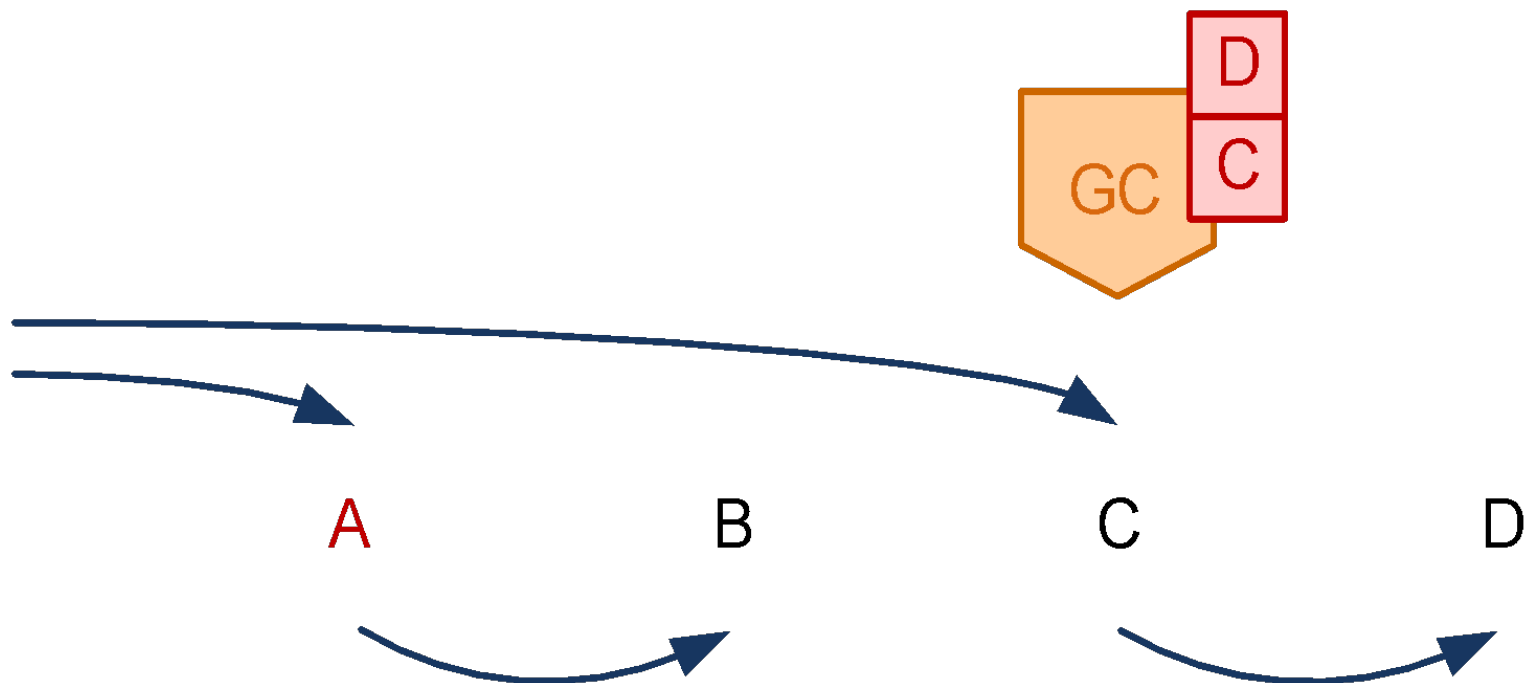
SATB барьер



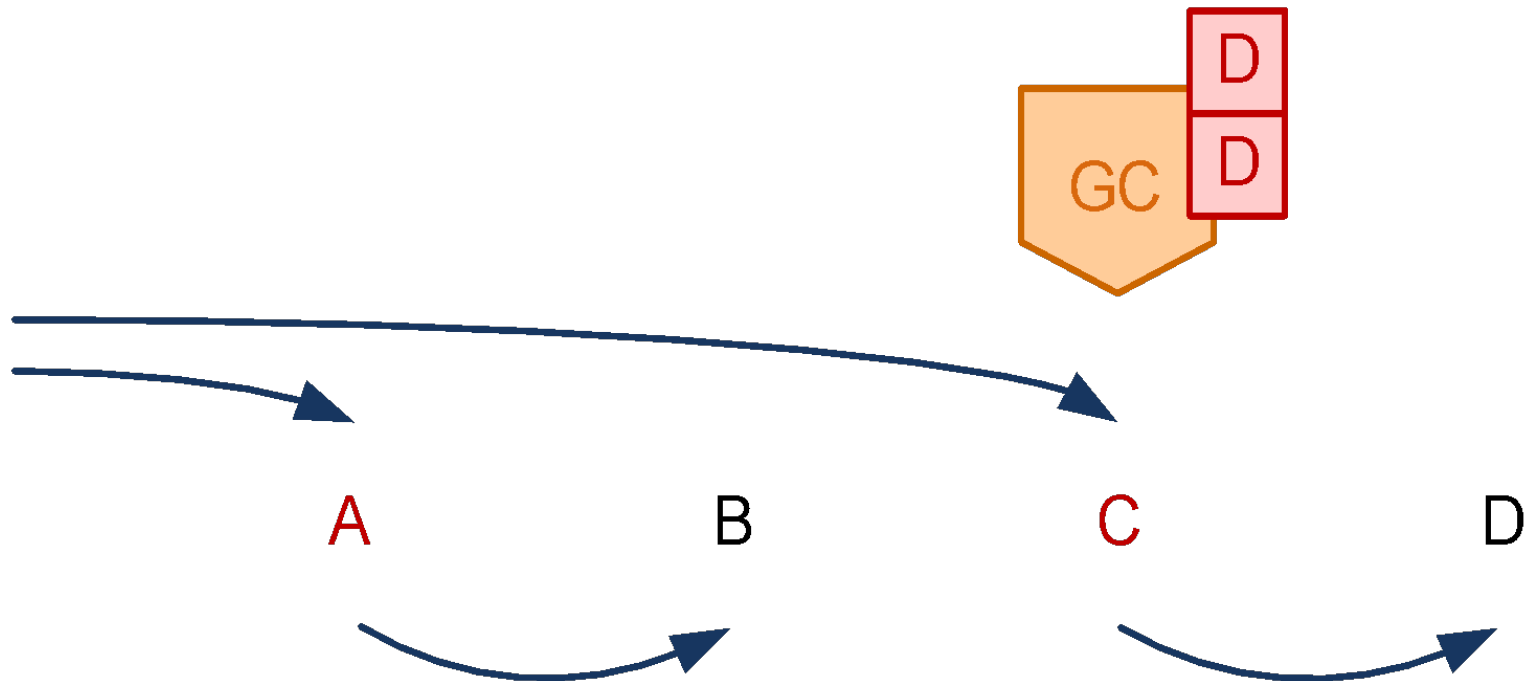
SATB барьер



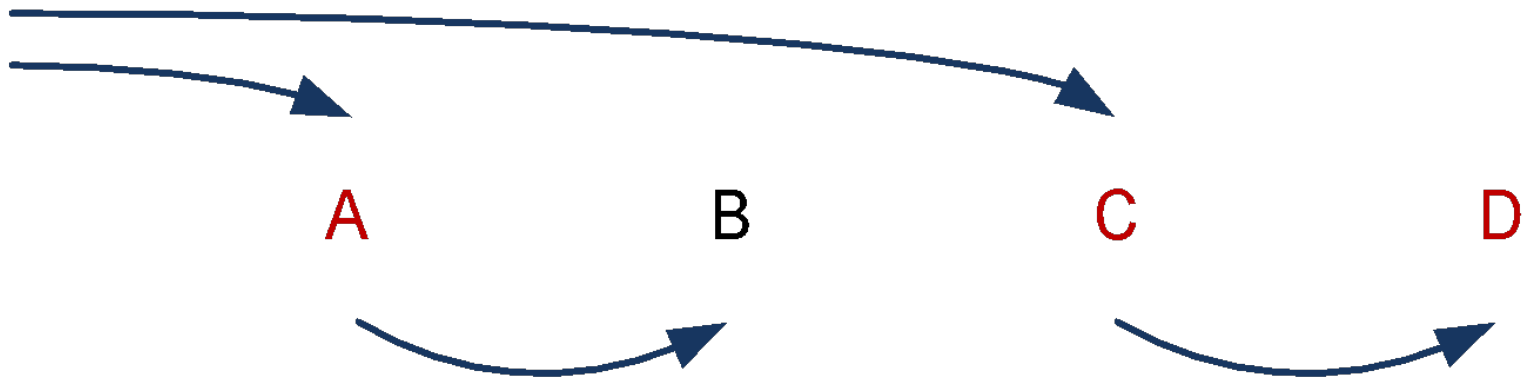
SATB барьер



SATB барьер



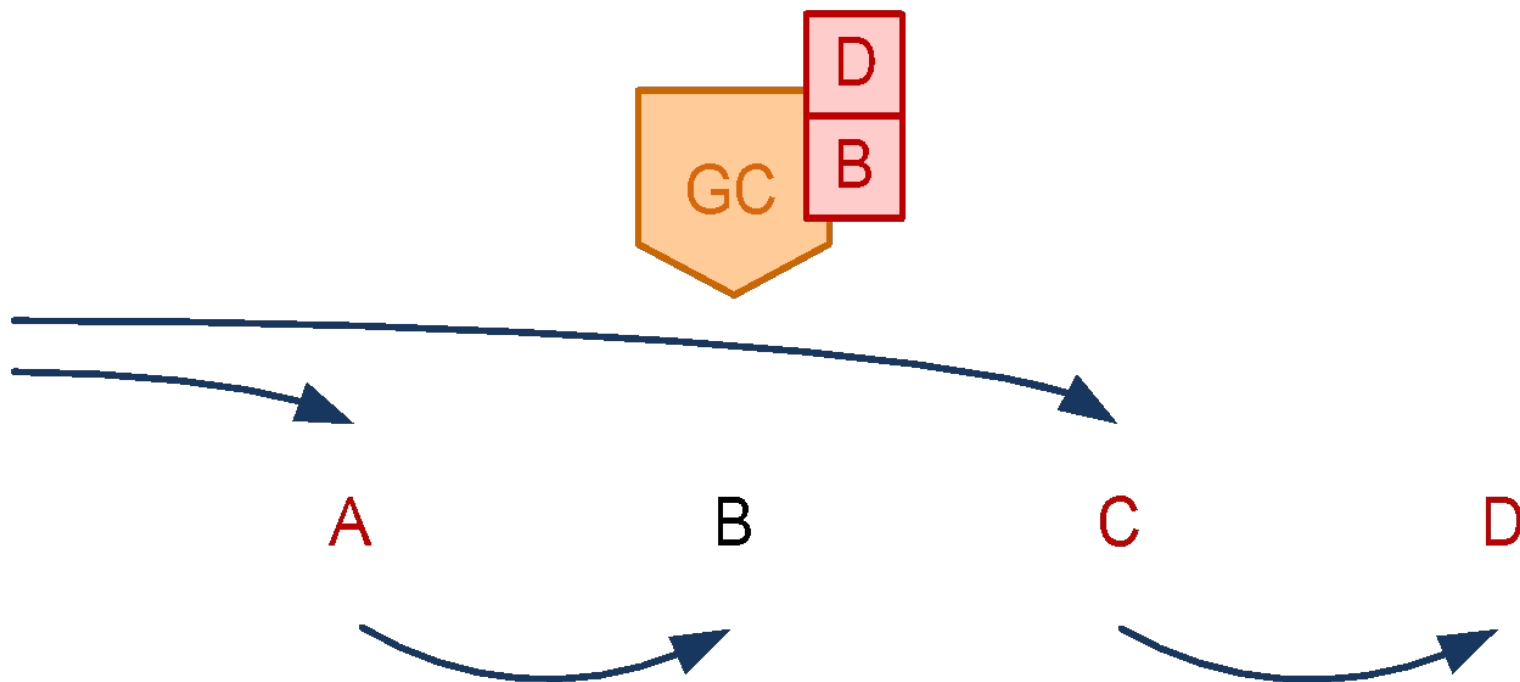
SATB барьер



Reference queue:

B	D
---	---

SATB барьер



Reference queue: empty



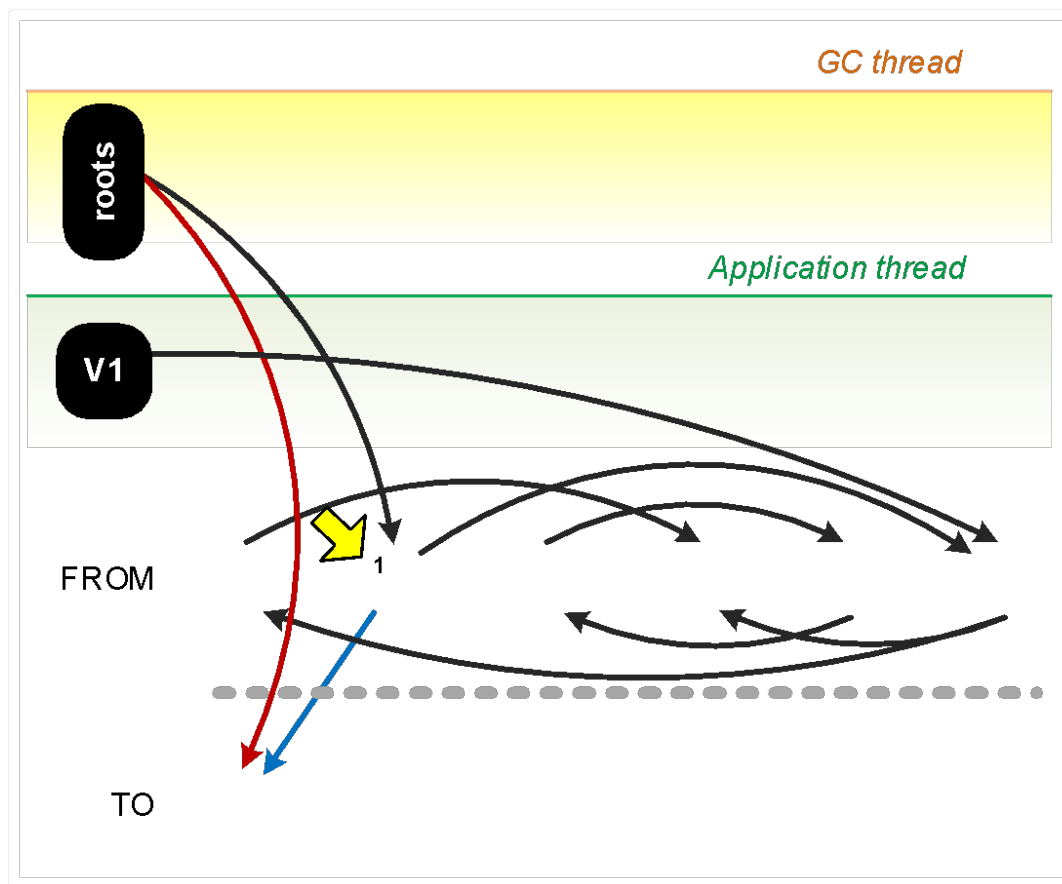
Обоснование

Фоновое копирование

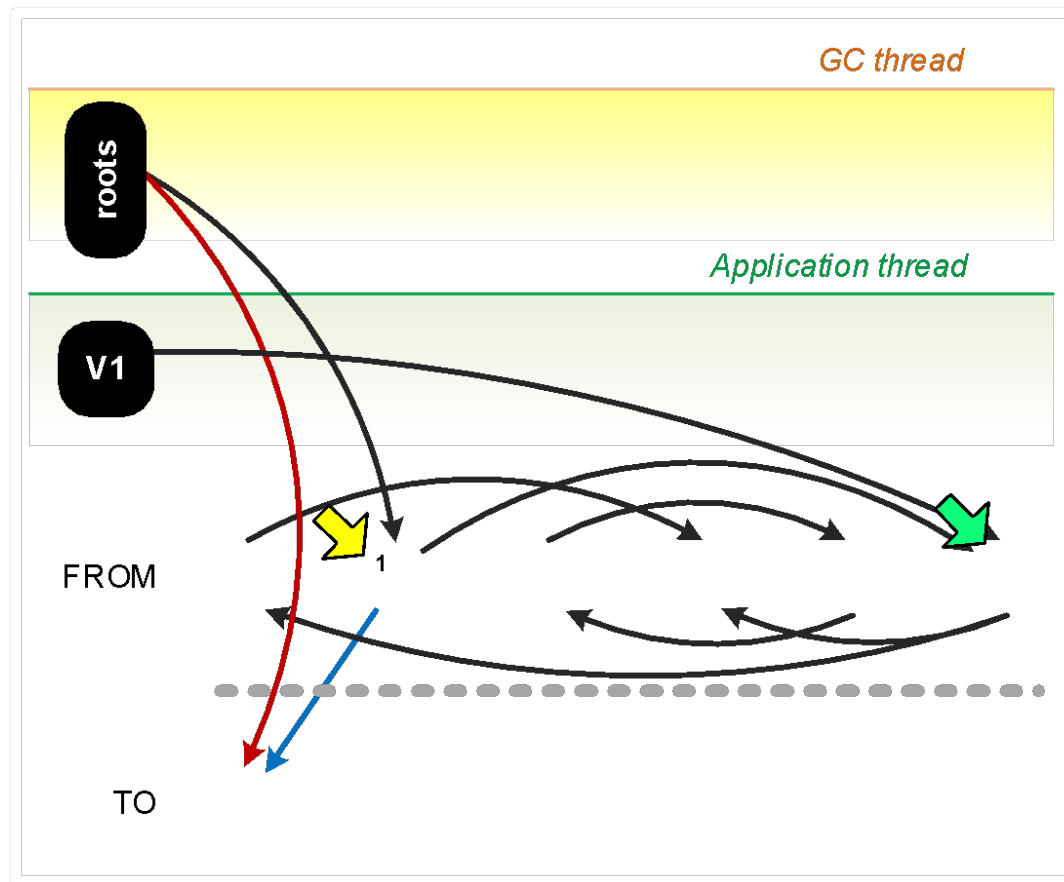
- Brooks^[1] style barrier – барьер на чтение
- Forwarding pointers
- Копирование при записи
- **Фоновое копирование**
- **Меньшая нагрузка на прикладные потоки**

[1] Rodney A. Brooks “Trading Data Space for Reduced Time and Code Space in Real-Time Garbage Collection on Stock Hardware” 1984 Symposium on Lisp and Functional Programming

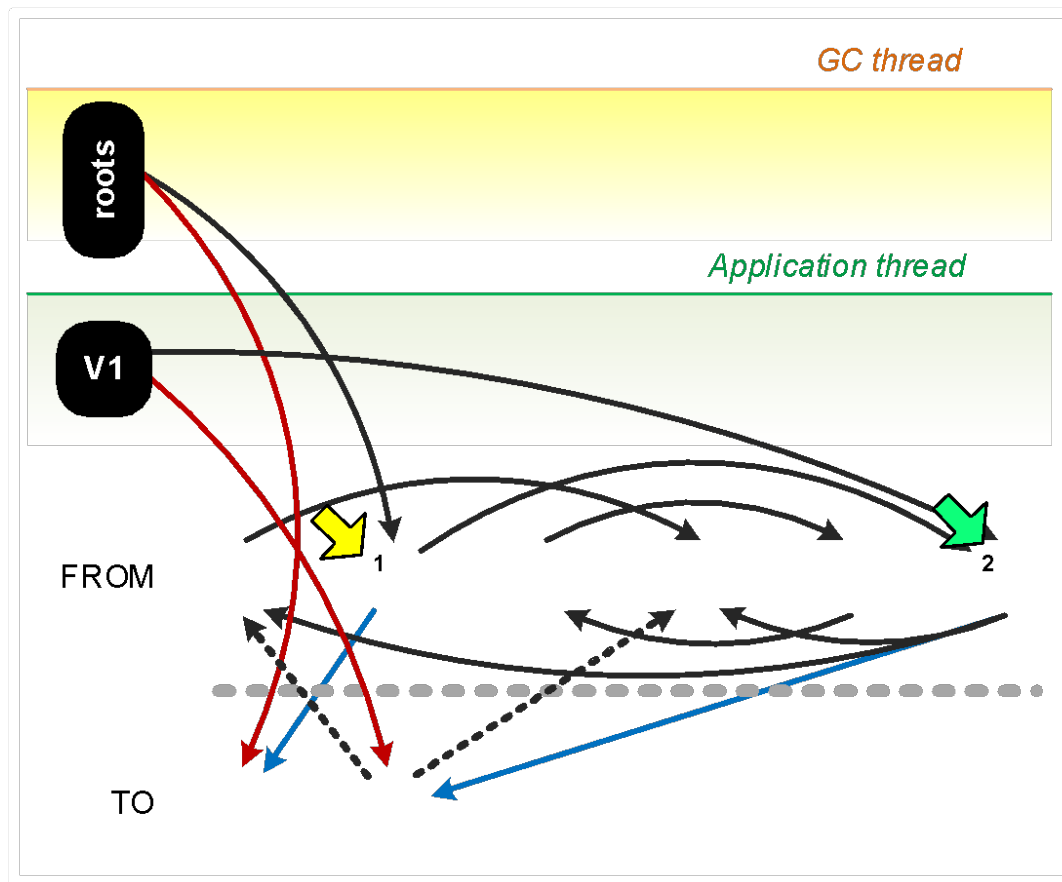
Алгоритм “метрономом”



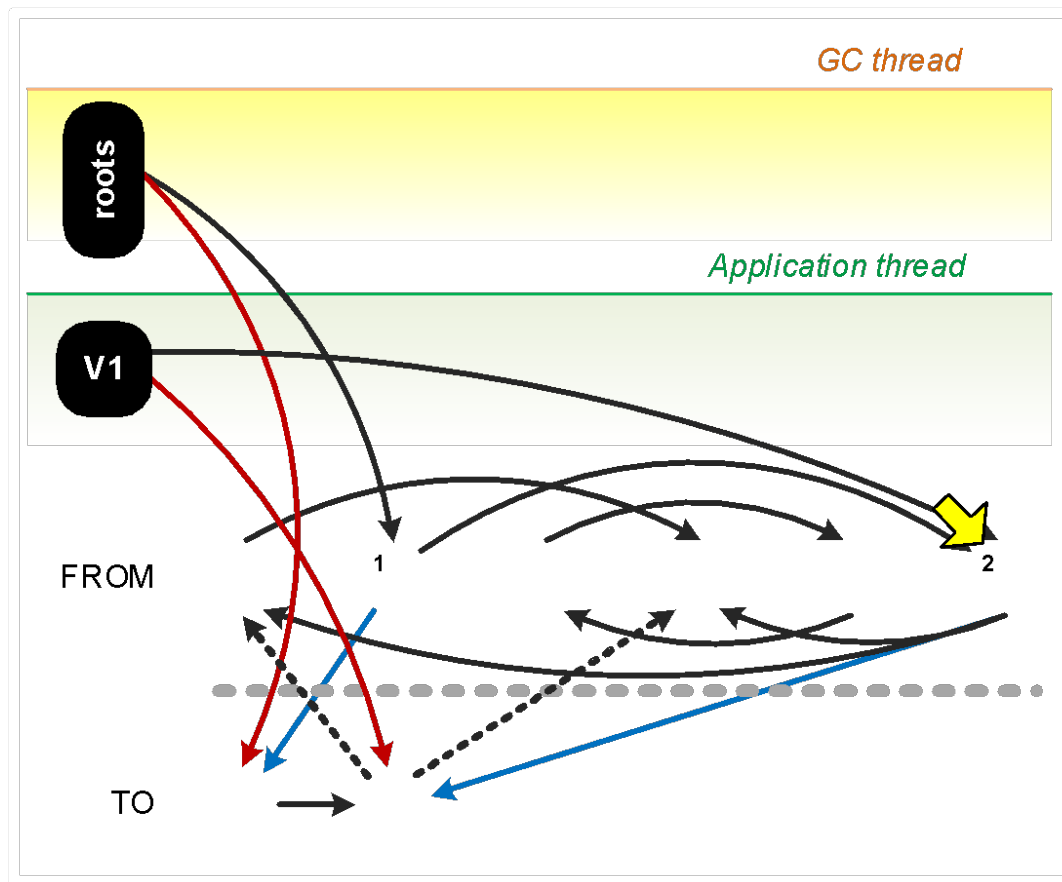
Алгоритм “метрономом”



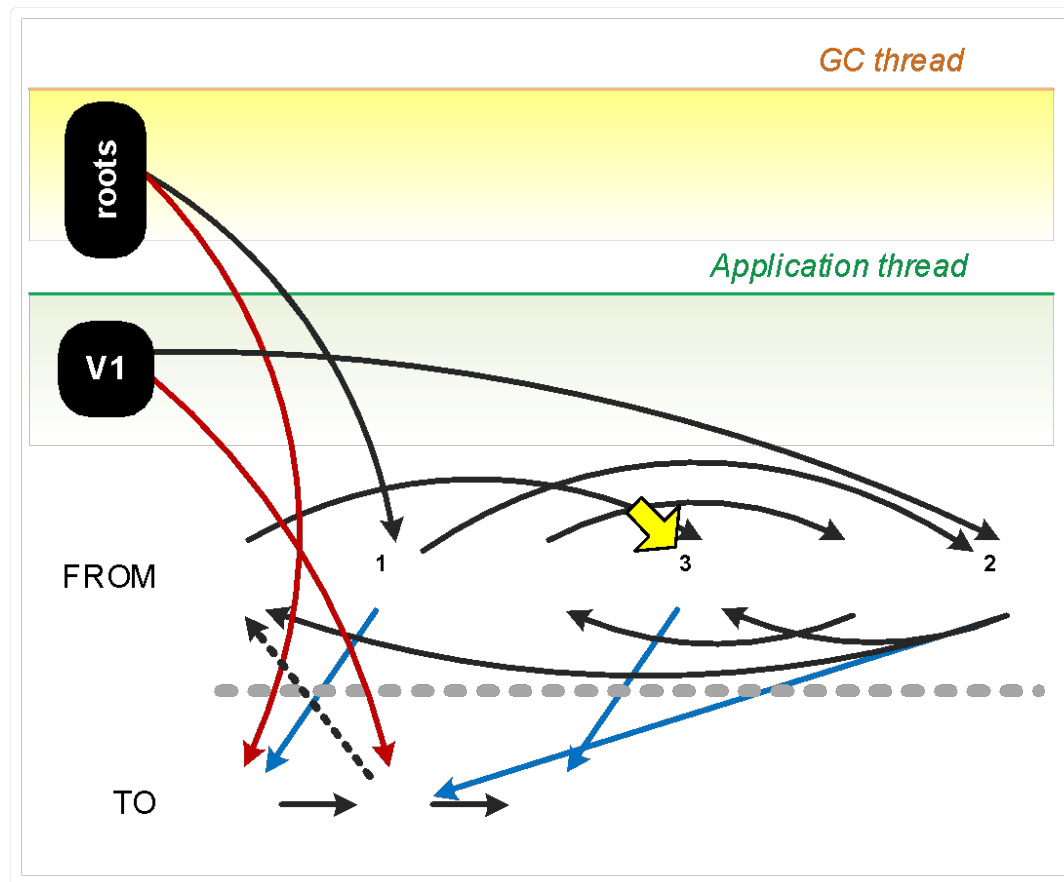
Алгоритм “метрономом”



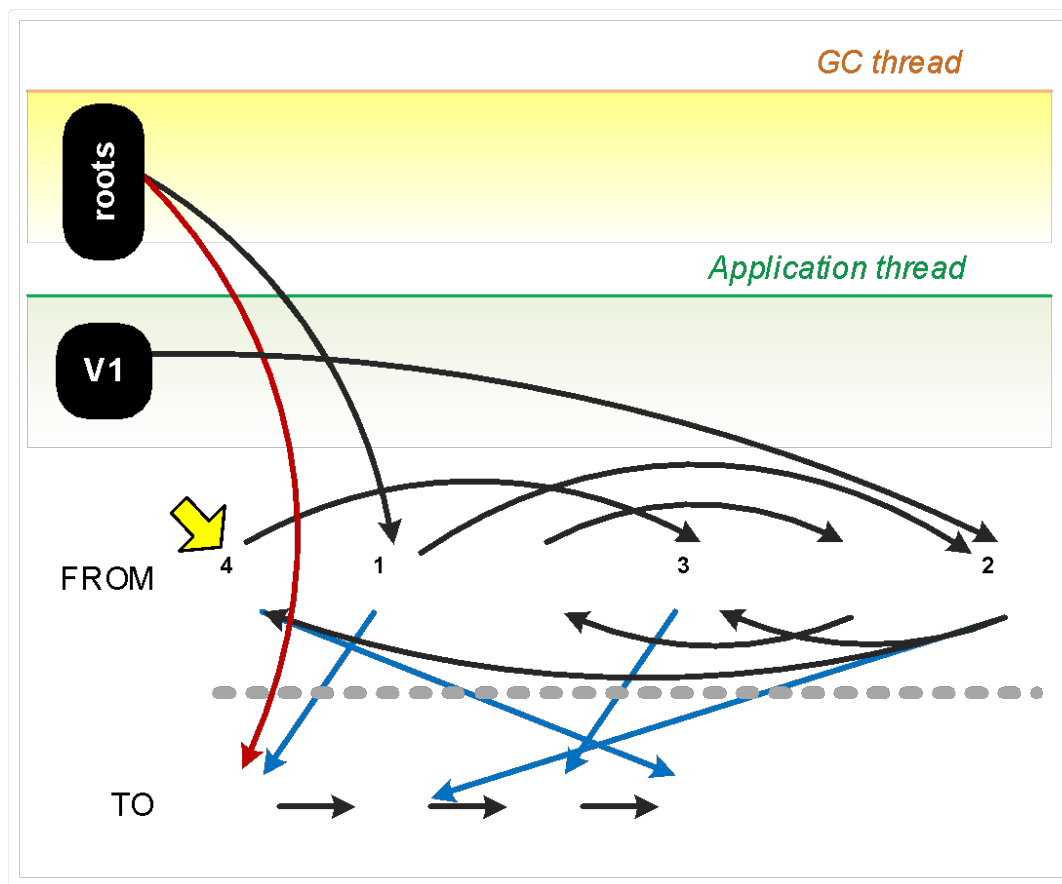
Алгоритм “метрономом”



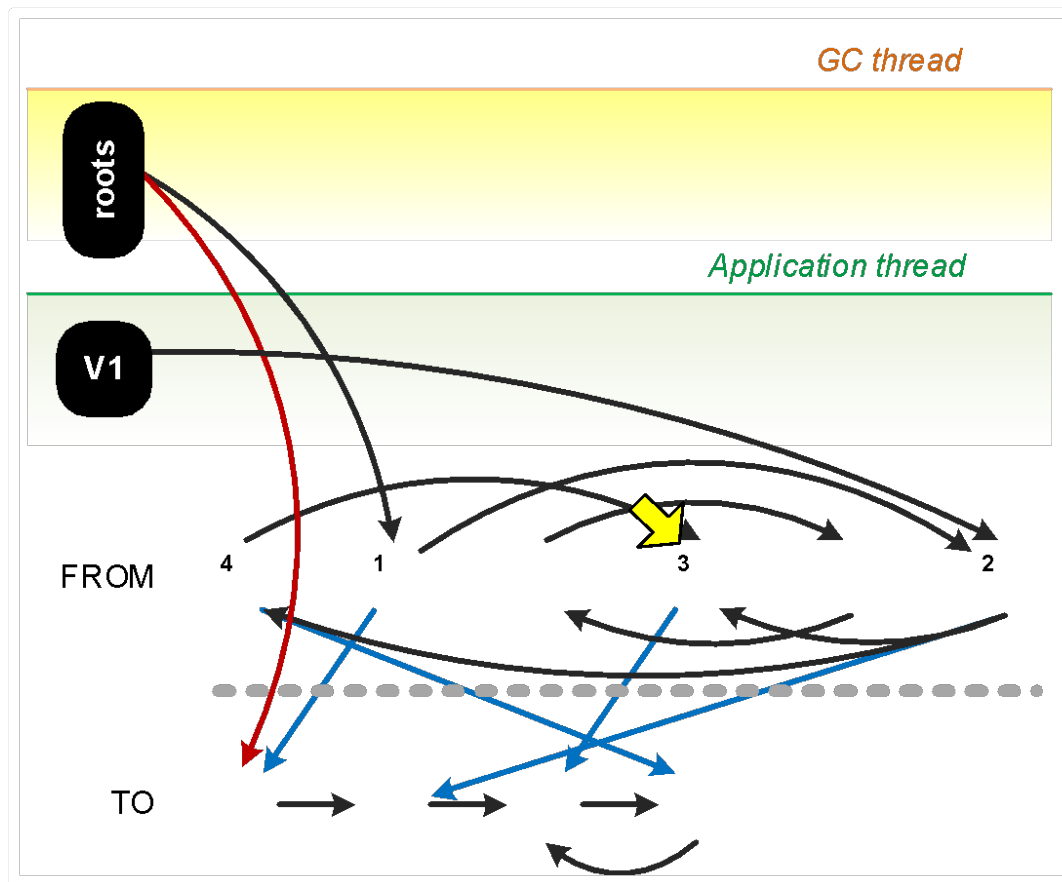
Алгоритм “метрономом”



Алгоритм “метрономом”



Алгоритм “метрономом”

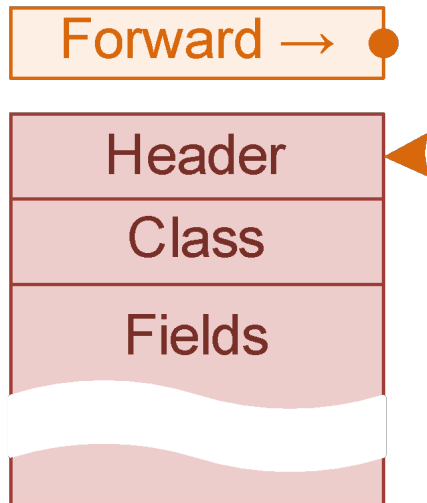




Фоновое копирование

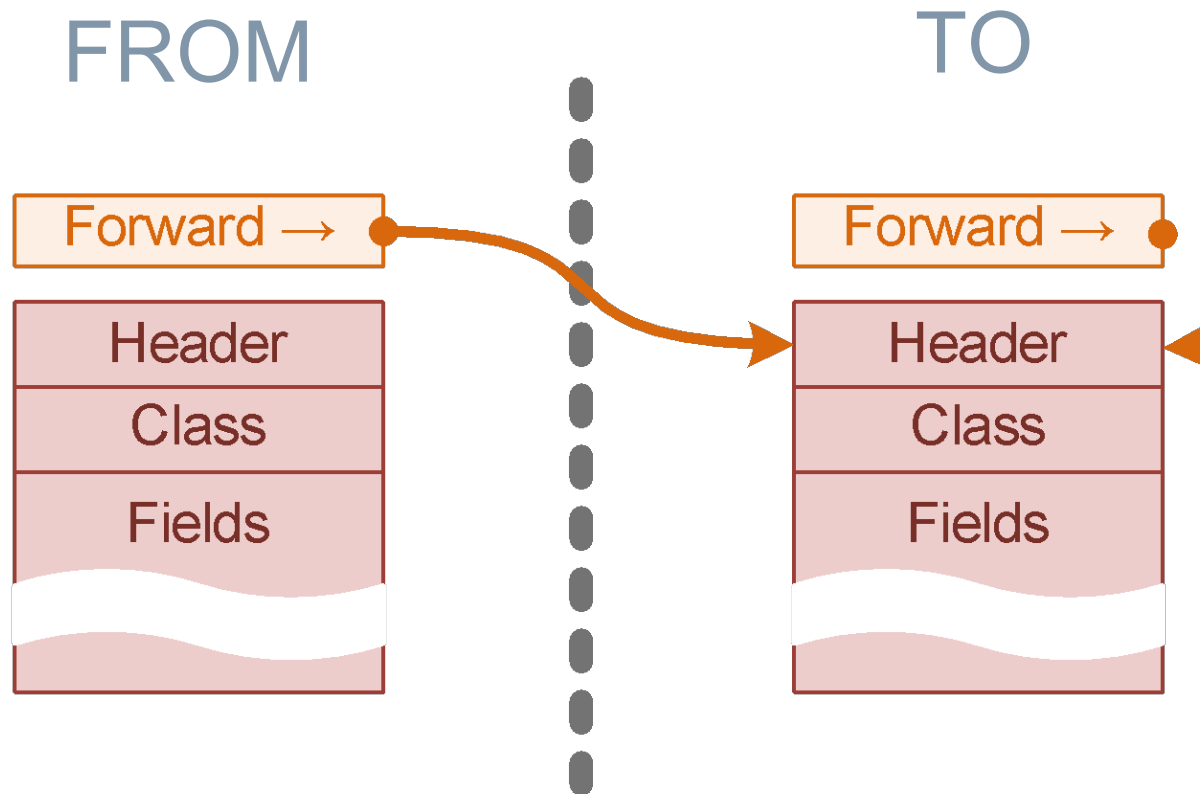
FROM

TO





Фоновое копирование





Алгоритм “Shenandoah”

- Сборка корневых ссылок - STW
- Маркировка
- Выбор регионов для сборки
- Копирование при записи
- Копирование
 - Сборка корневых ссылок
 - Маркировка
 - Выбор регионов для сборки
 - Копирование при записи

...



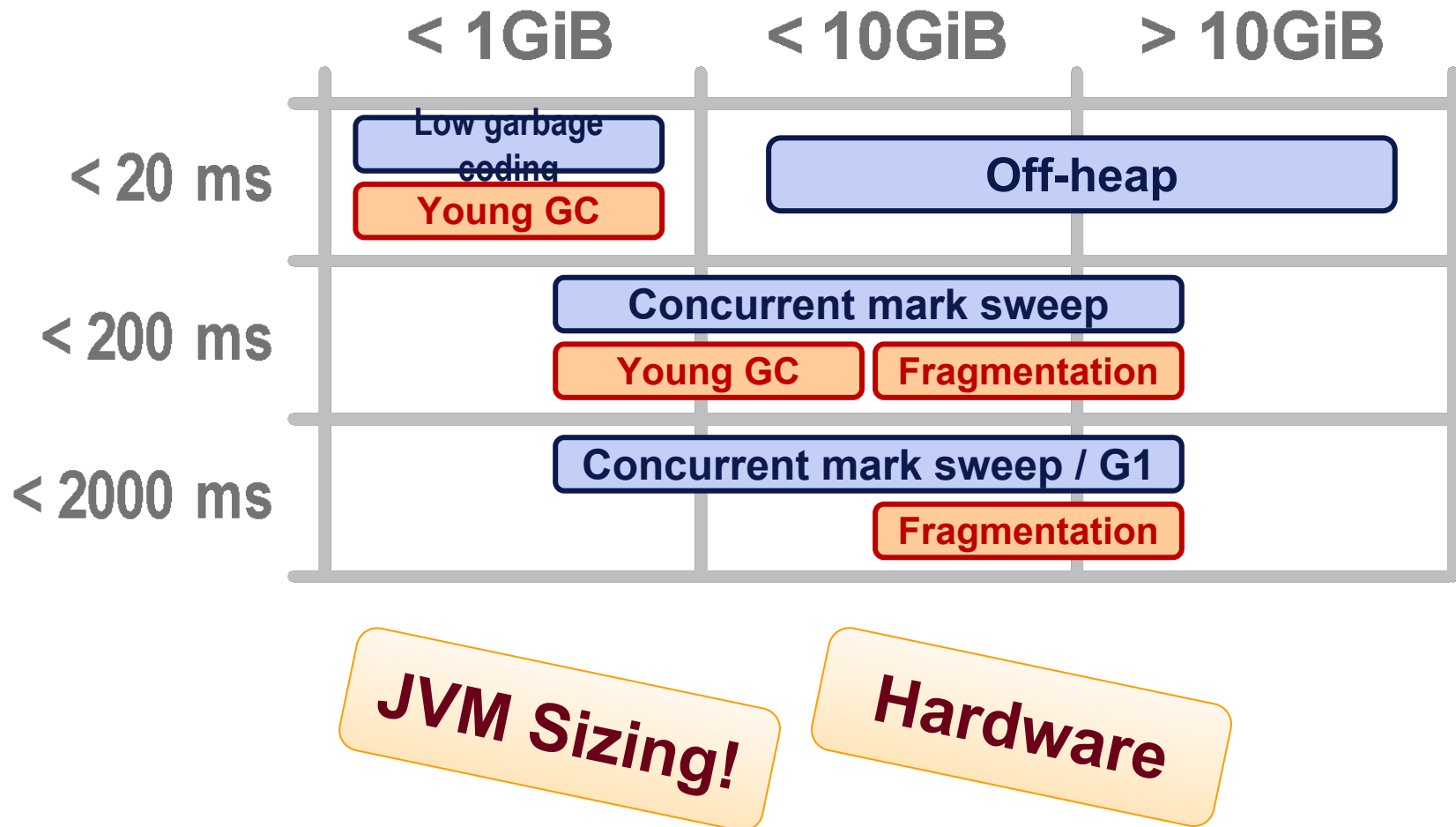
Алгоритм “Shenandoah”

Критика

- **Single space**
 - ✓ Работает только если приложение не создаёт мусор
- **Инкрементальная сборка**
 - ✓ Фрагментация больших объектов
 - ✓ Фрагментация малых средних объектов не актуальна на больших (~100 GiB) кучах
- **Производительность**
 - ✓ Read barrier + write barrier

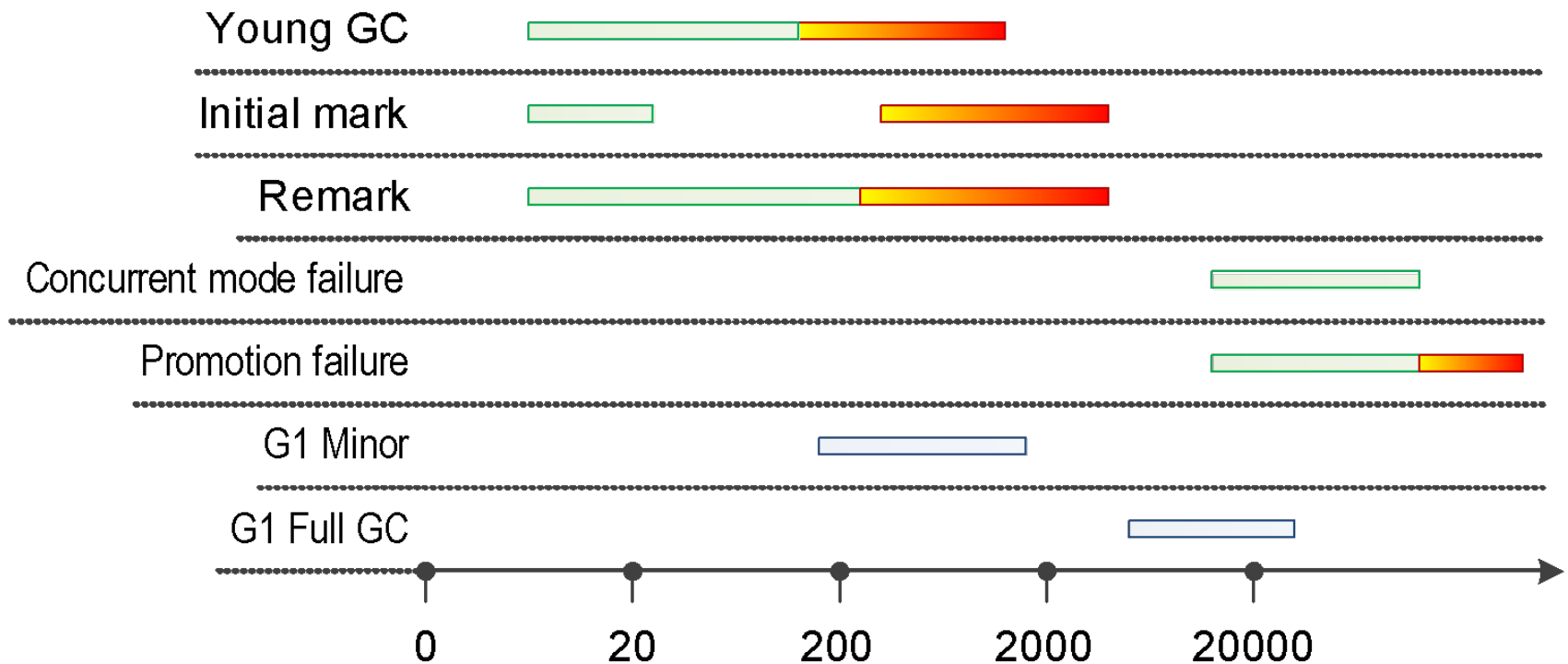


HotSpot JVM и паузы





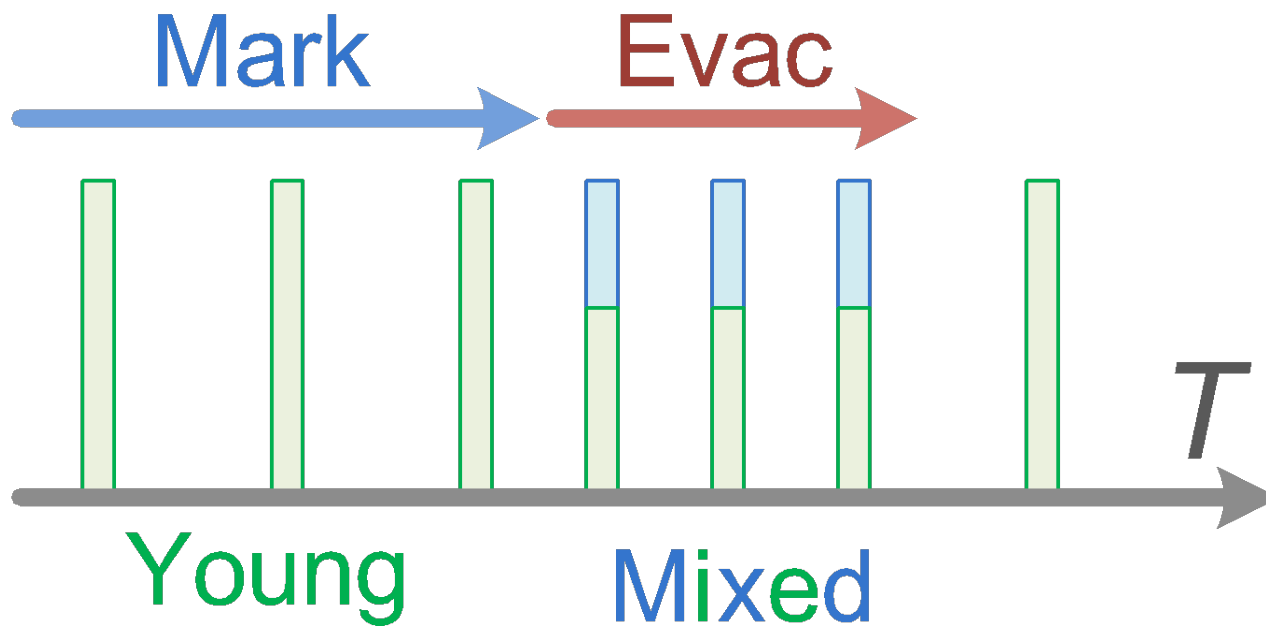
HotSpot JVM и паузы





G1 плюсы и минусы

G1 – адаптивное управление паузами





G1 плюсы и минусы

Плюсы

- Адаптивная стратегия сборок
- Параллельный Full GC

Минусы

- **STW копирование** – длинные паузы
- Фрагментация (объекты больше региона)
- Пропускная способность
- **Всё ещё в процессе разработки!**



CMS плюсы и минусы

Плюсы

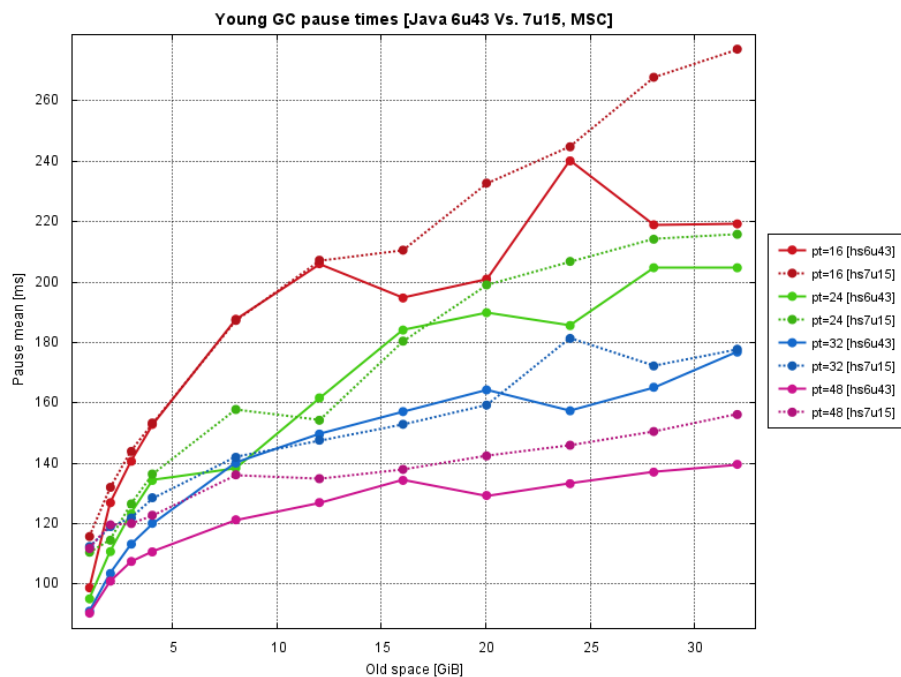
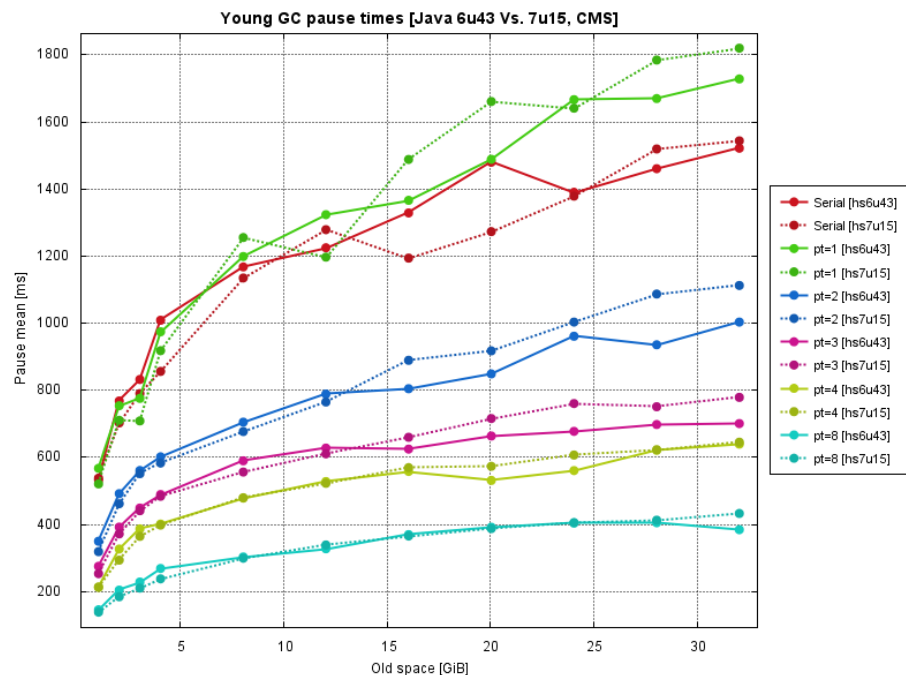
- Фоновая сборка
- Малые паузы
- Хорошая пропускная способность
- Малый оверхед по памяти

Минусы

- *Фрагментация – легко лечится*
- Однопоточный Full GC
- Синдром “аномальных пауз”
- **Заброшен**



CMS: Реальная проблема



<http://blog.ragozin.info/2013/06/java-gc-in-numbers-parallel-young.html>



CMS: Реальная проблема

Young
collection

Initial
mark

Remark

Scan
thread
stacks

Scan dirty cards

Copy
live
objects

Scan
thread
stacks

Scan
young
space

Scan
thread
stacks

Scan
young
space

Scan dirty cards

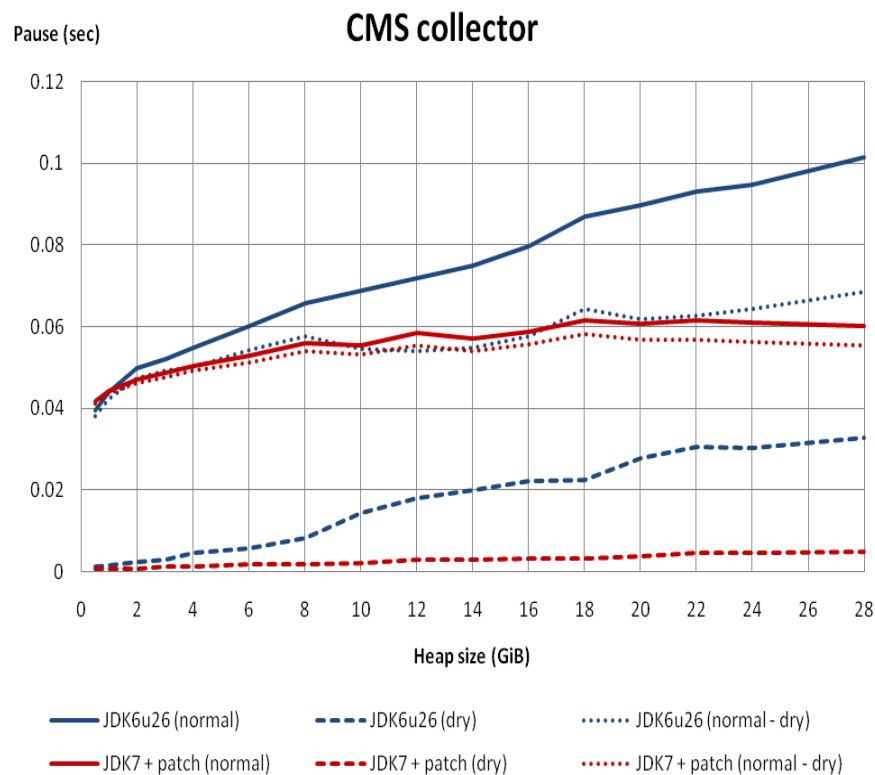
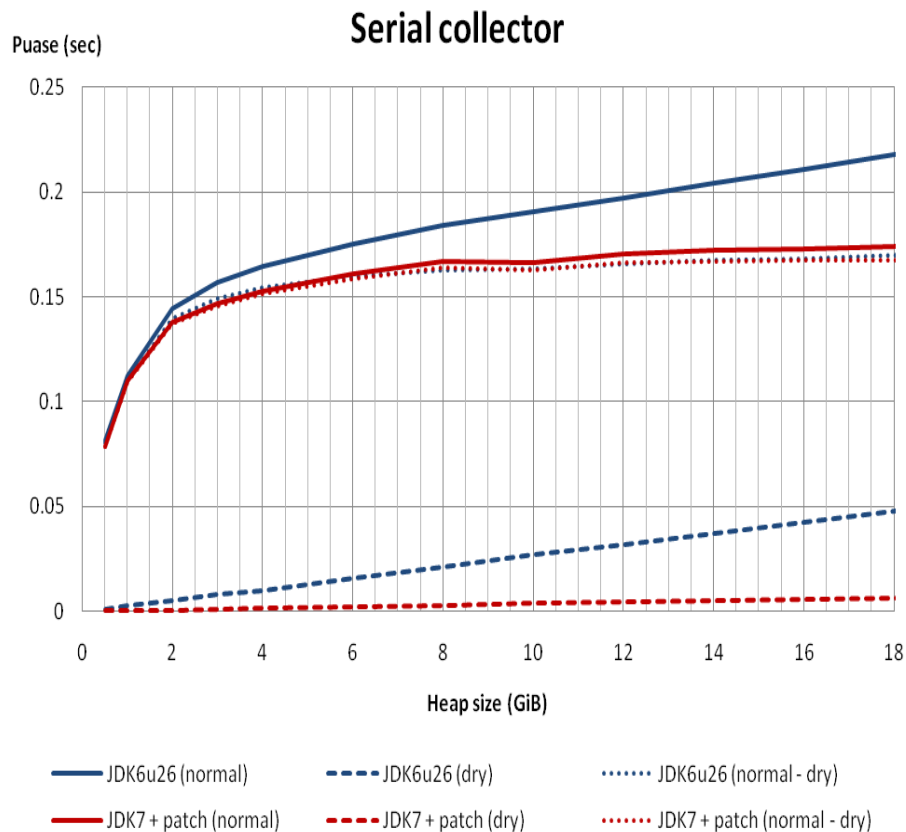
Read
card
table

Scan
dirty
pages

Read
card
table

Scan
dirty
pages

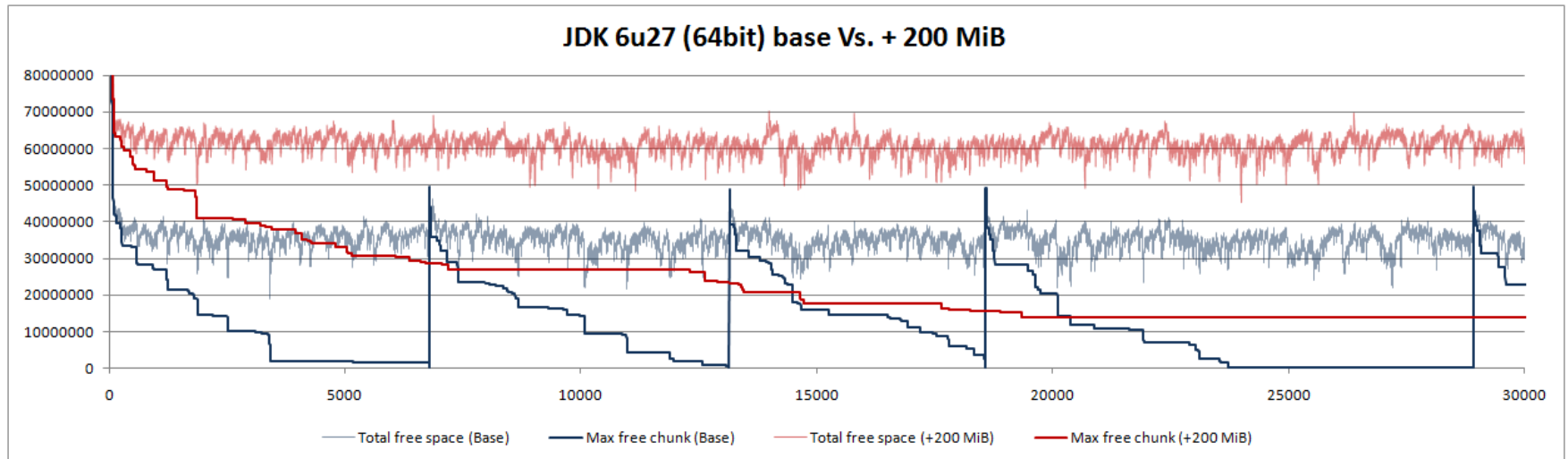
CMS: Реальная проблема



<http://blog.ragozin.info/2011/07/openjdk-patch-cutting-down-gc-pause.html>



CMS: А как насчёт фрагментации?



<http://blog.ragozin.info/2011/10/cms-heap-fragmentation-follow-up-1.html>

Секрет борьбы с паузами GC



Секрет тюнинга GC

Чтобы освободить **память** от мусора
– нужна **память** чтобы её замусорить!

-Xmn<N> -Xms<M> -Xmx<M>



Спасибо!

Алексей Рагозин
alexey.ragozin@gmail.com
<http://blog.ragozin.info>