



Ускоряем Apache.JMeter в 5 раз

Вячеслав Смирнов

Специалист по тестированию
производительности, Райффайзенбанк

Бываю в Омске





APACHE
JMeter™



JMeter удобно профилировать

Курс по профилированию JVM

Что профилировать?

Замедляю **Apache.JMeter**

Курс по Apache.JMeter, чаты

Как его ускорить?

Ускоряю **Apache.JMeter**

Сапожник с сапогами



Два следствия "тормозов"

1. Недовес, мало нагрузки

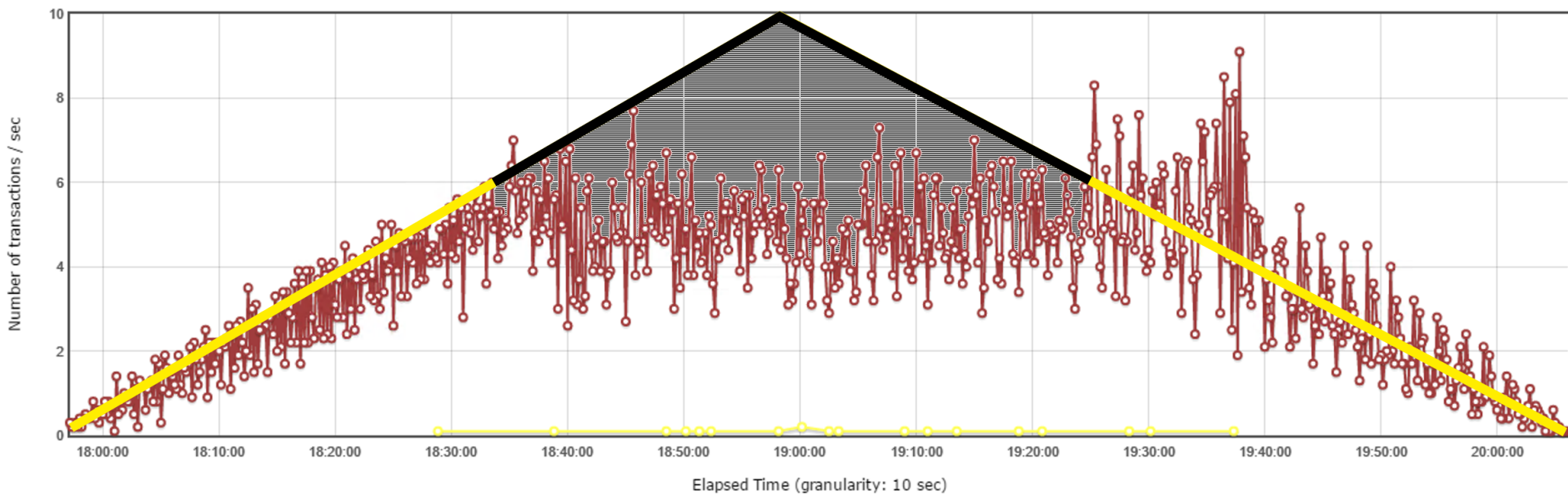
- Хотим 10 операций в секунду
- Получаем 6

2. Перевес, мало ресурсов

- Хотим уложиться в 4 ГБайт ОЗУ
- Получаем `OutOfMemoryError`

Недовес, мало нагрузки

Transactions Per Second



Перевес, мало ресурсов

01. summary +	1454	in 00:00:30	=	48.5/s	Avg:	0	Min:	0	Max:
02. summary =	5155	in 00:03:02	=	28.3/s	Avg:	0	Min:	0	Max:
03. summary +	1684	in 00:00:30	=	56.1/s	Avg:	0	Min:	0	Max:
04. summary =	6839	in 00:03:32	=	32.2/s	Avg:	0	Min:	0	Max:
05. summary +	488	in 00:00:30	=	16.1/s	Avg:	2	Min:	0	Max:
06. summary =	7327	in 00:04:03	=	30.2/s	Avg:	0	Min:	0	Max:
07.	java.lang.OutOfMemoryError: Java heap space								
08.	Killed								

Три причины "тормозов"

1. Профиль нагрузки
2. Ограничения JVM и системы
3. Выбор и настройка компонент
4. Ошибки

Профиль нагрузки

Как не делать лишнюю работу

Непредсказуемая и лишняя работа

- Открытая модель нагрузки
- Шаг и потоки не рассчитан
- Нет защиты от зависания
- Нет проверки на ошибки

Открытая модель нагрузки

Задача: **100 000** операций за час

Выделяем **1** поток в JMeter:

- **36** секунд на операцию
- **100** операций в час
- одного потока **мало**

Открытая модель нагрузки

Задача: **100 000** операций за час

Выделяем **1 000** потоков в JMeter:

- **72** секунды на операцию
- **50 000** операций в час
- тысячи потоков **мало**

Открытая модель не для интенсивности

Задача: 100 000 операций за час

Выделяем 10 000 потоков в JMeter:

- система пятисотит, не отвечает
- JMeter падает с `java.lang.OutOfMemoryError: Java heap space`

Открытая модель для количества

Задача: Сколько операций за час при 1 000 потоках?

CI/CD метрика

Контролируемая интенсивность и ступени

Задача: **100 000** операций за час ~ **28** операций/транзакций в секунду

Плавный разгон

— **7**

— **+7**

— **+7**

— **+7 (28 tps)**

— **+7**

Контролируемая интенсивность и ступени

Задача: **100 000** операций за час ~ **28** операций в секунду

Большая первая ступень:

— **22**

— **+2**

— **+2**

— **+2 (28 tps)**

— **+2**

Thread Group задаёт частоту запуска

Задача: **100 000** операций за час ~ **28** операций в секунду **начинаются**

Система может отвечать **долго**

A Thread Group продолжит создавать потоки по расписанию

Что приведёт к **взрывному росту** количества активных потоков

Прерываем запросы, ответы и транзакции

- Запросы:

Connect timeout

- Ответы:

Response timeout

- Транзакции:

Duration Assertion

Будут фиксироваться **ошибки**, на них надо **реагировать**

JMeter получает ошибку

По умолчанию проигнорируется :

- обработку ошибок не пишут — ради скорости;
- Action to be taken after sampling error: **Continue** ;
- получаем **нарушение** логики сценария и **ненужные** запросы.

На ошибки надо **реагировать**:

- Action to be taken after sampling error: **Stop Thread** .

Сколько стоит создать поток?

TODO: посчитать, профилирование SJK, JFR

Один поток на 10 итераций — кешируем

TODO: посчитать, профилирование SJK, JFR

1. Меняем:

- Action to be taken after sampling error: Start Next Thread Loop.

2. Уменьшаем:

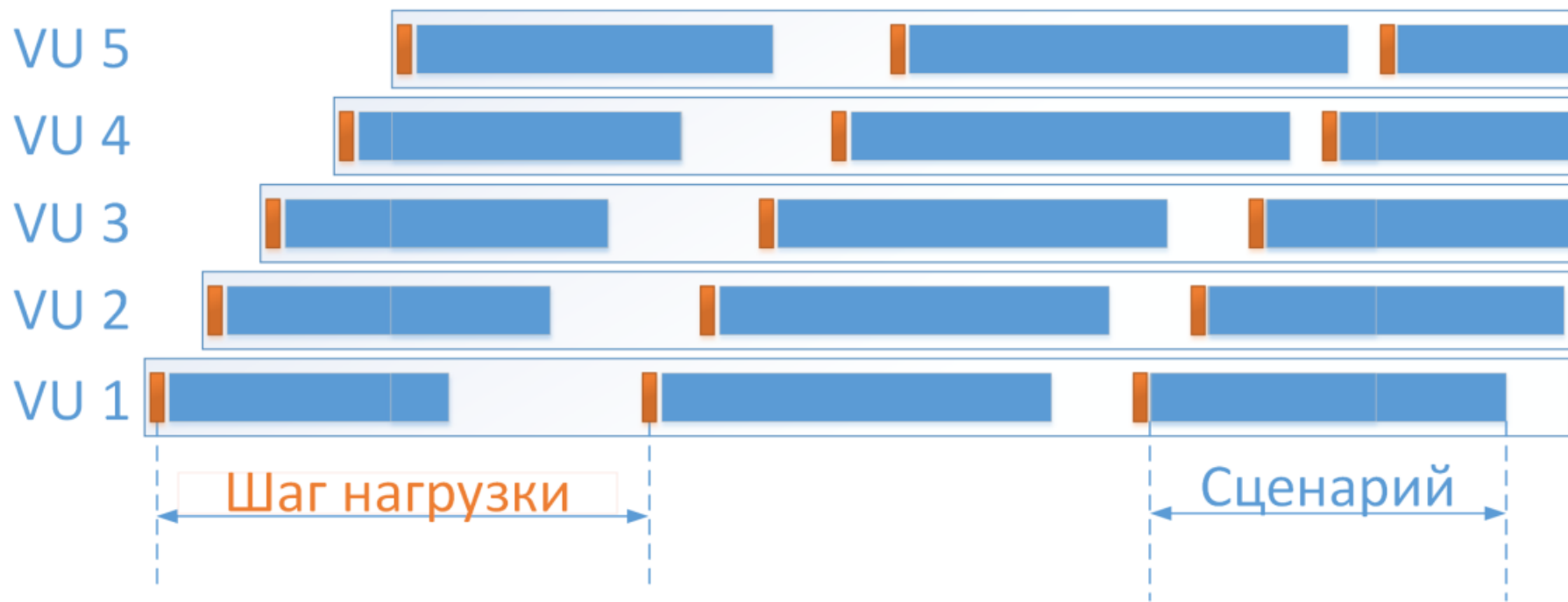
- Number of Threads: в 10 раз.

3. Увеличиваем:

- Loop count: в 10 раз.

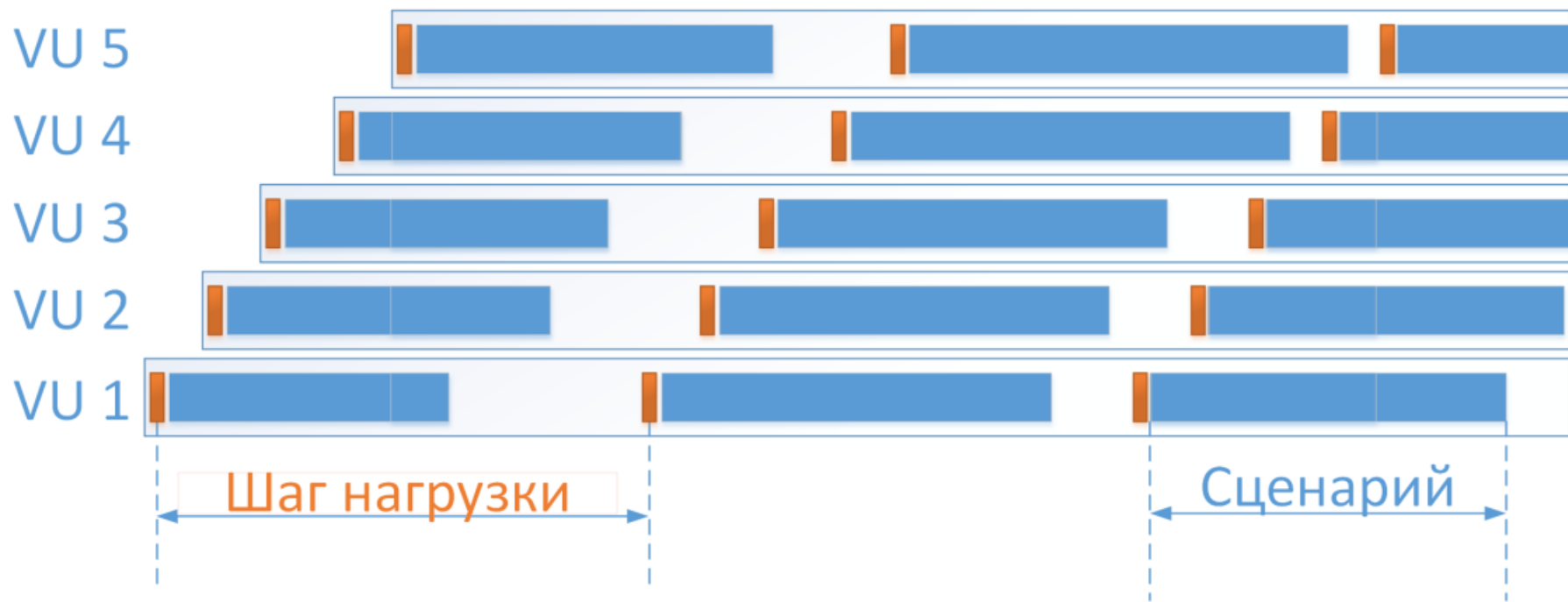
Ultimate Thread Group — постоянные потоки

Шаг нагрузки: Test Action + Constant Throuthput Timer (не меньше указанного)



Ultimate Thread Group — постоянные потоки

Шаг нагрузки: Test Action + Precise Throuthput Timer (равен указанному)



Способы задания интенсивности

Статья: loadtestweb.info/2017/08/23/pacing/

TODO: написать новую

	A	B	C	D	E	F
1	Предполагаемый шаг работы сценария (секунд):	80	1. Ввести предполагаемый размер шага в секундах			
2	Предполагаемый шаг работы сценария (минут):	1,333333333	Расчёт размера шага в минутах			
3	Предполагаемая интенсивность работы одного пользователя (сценариев в минуту):	0,75	Расчёт интенсивности работы пользователя			
4						
5	Требуемая интенсивность (сценариев/в минуту):	2,5	2. Ввести необходимую интенсивность работы теста			
6						
7	Предполагаемое количество пользователей:	3,333333333	Расчёт предполагаемого количества пользователей			
8						
9			3. Ввести или выбрать количество пользователей			
10			Расчёт шага для разного количества виртуальных пользователей			
11	Дельта:	*	-1	+0	+1	
12	Количество виртуальных пользователей:	4	2	3	4	Значение указывается в Thread Group или в jp@gc - Ultimate Thread Group
13	Расчётная интенсивность работы одного пользователя (сценариев в минуту):	0,625	1,25	0,833333333	0,625	Значение указывается в Constant Throughput Timer
14	Расчётный шаг работы сценария (минут):	1,60	0,80	1,20	1,60	
15	Расчётный шаг работы сценария (секунд):	96,00	48,00	72,00	96,00	Значение сравнивается с предполагаемым шагом работы сценария
16			Если фон зелёный, то такого количества виртуальных пользователей достаточно, чтобы выдержать шаг нагрузки. Если фон красный, то при таком количестве пользователей шаг будет недостаточно большим.			
17	Рекомендуемые значения Startup Time или Rump-Up Period (секунд):	96	48	72	96	
18	Рекомендуемые значения Startup Time или Rump-Up Period (секунд):	192	96	144	192	
19	Рекомендуемые значения Startup Time или Rump-Up Period (секунд):	288	144	216	288	

Ограничения JVM и системы

Знай свои потребности

Выбор и настройка компонент

Инструмент под задачу

Ошибки

Нет предела совершенству



Райффайзен
БАНК

Вячеслав Смирнов

polarnik.github.io, @qa_load

Спасибо!