

# Тестирование оптимизаций в кластерах баз данных

Контур

Сергей Махетов

Ведущий специалист по тестированию

## О себе

В области разработки ПО > 10 лет,  
тестирование > 5 лет.

Специализация: Инфраструктурные  
решения (интеграционные шины и  
системы хранения данных)

Стек: Java/Linux, Scala, JMeter,  
Gatling

Системы на текущий момент: Kafka,  
ES, ClickHouse



# **О системе, в поддержке и совершенствовании которой принимаю участие**

Команда телеметрии Контура – логи, метрики, распределенные трассировки

Различные хранилища – ElasticSearch, ClickHouse, Kafka, Cassandra, ...

Повышенные требования к производительности (> 3 000 000 Events per second)

Хостимся не в публичных облаках

## **Когда не хватает ресурсов БД**

Накинуть железа(ресурсов в облаке)?

Снизить нагрузку со стороны клиентов?

Оптимизировать!

# План

- Терминология
- 3 кейса оптимизации
- Оптимизация в общем
- Особенности тестирования
- Осознанность проведения оптимизаций

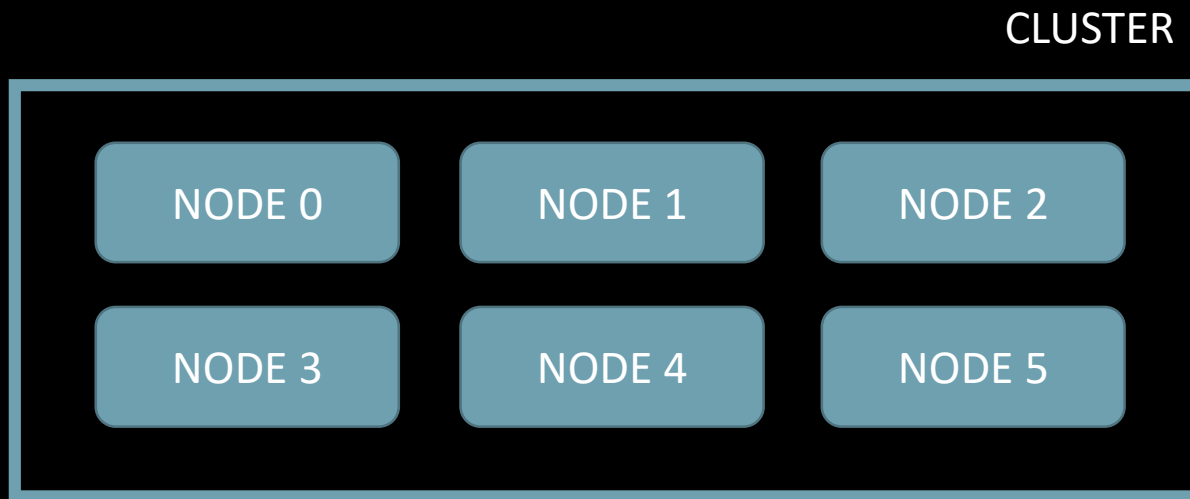
# Терминология

# Кластер

- (англ. cluster «скопление, кисть, рой») — объединение нескольких однородных элементов, которое может рассматриваться как самостоятельная единица, обладающая определёнными свойствами.

# Нода

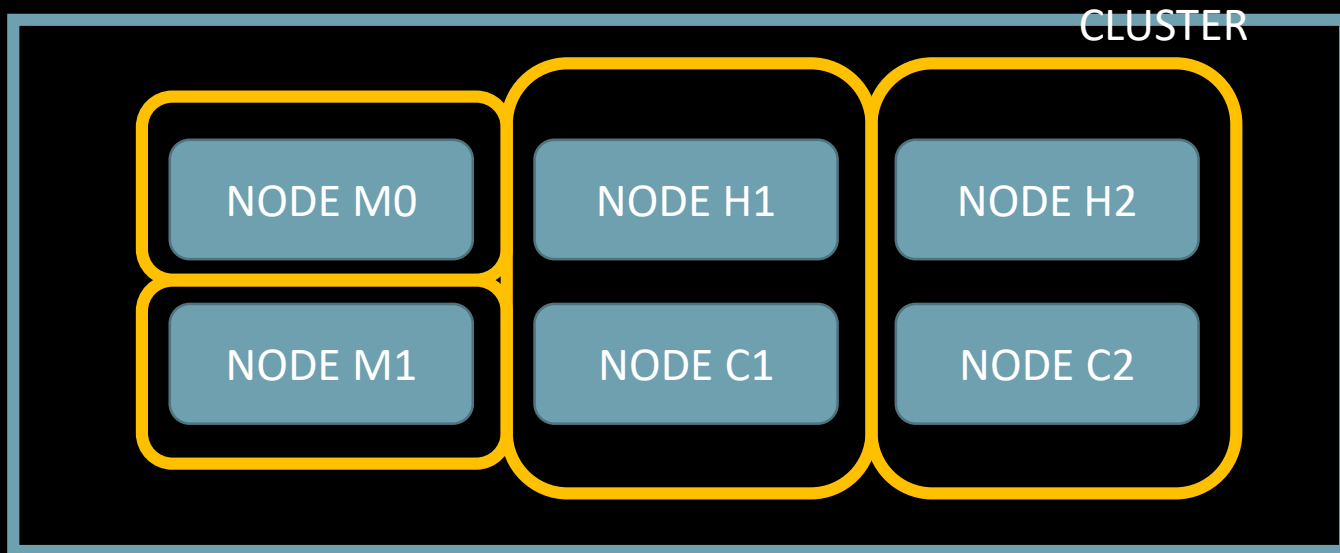
- (от лат. nodus — узел), отдельный узел кластера(сервер БД)





# Нода

- Может быть несколько на сервере
- Может иметь специализацию



An orange diagonal stripe runs from the top-left corner towards the bottom-right, separating the orange background on the left from the black background on the right.

# **Кейс 1.**

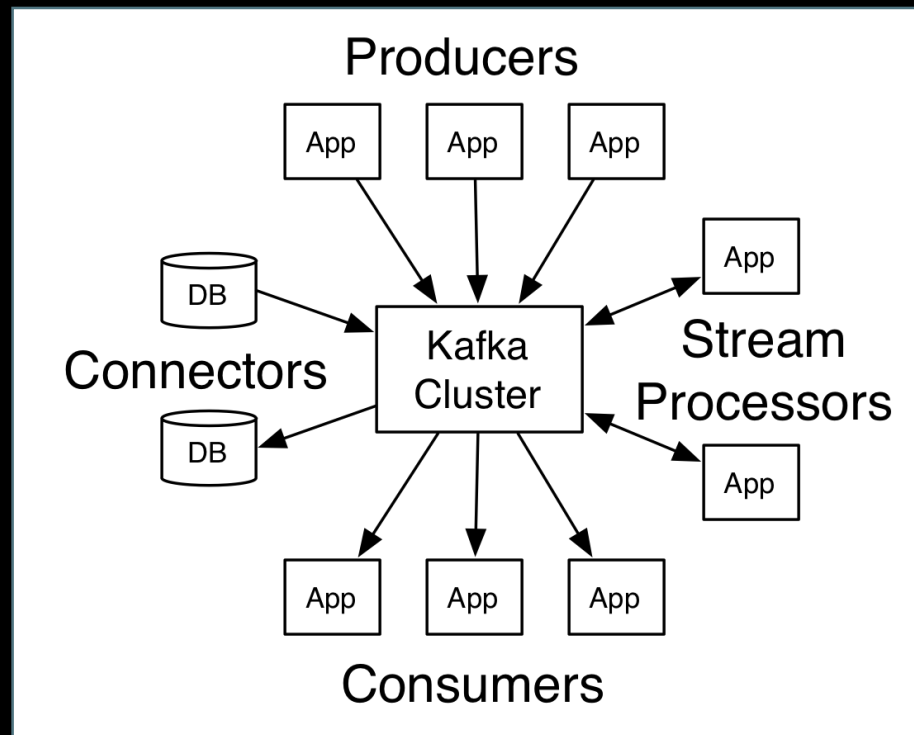
## **Нехватка ресурсов в Кафке**

# Apache Kafka

- **Распределенная стримминговая платформа**
- **Горизонтально масштабируется**
- **Отказоустойчивая**
- **Производительная**



# Apache Kafka



# Предельная утилизация сети и дисков

Высокая утилизация (до 80% )  
дисков и сети на множестве  
серверов

Нет возможности роста без  
закупа оборудования

Малый запас для  
эксплуатационных действий

	CPU	LAN	DISC
Node1	30	70	66

# Предельная утилизация сети и дисков

Высокая утилизация (до 80% )  
дисков и сети на множестве  
серверов

Нет возможности роста без  
закупа оборудования

Малый запас для  
эксплуатационных действий

	CPU	LAN	DISC
Node1	30	70	66
Node2	28	67	80

# Предельная утилизация сети и дисков

Высокая утилизация (до 80% )  
дисков и сети на множестве  
серверов

Нет возможности роста без  
закупа оборудования

Малый запас для  
эксплуатационных действий

	CPU	LAN	DISC
Node1	30	70	66
Node2	28	67	80
Node3	43	72	70

# Предельная утилизация сети и дисков

Высокая утилизация (до 80% )  
дисков и сети на множестве  
серверов

Нет возможности роста без  
закупа оборудования

Малый запас для  
эксплуатационных действий

	CPU	LAN	DISC
Node1	30	70	66
Node2	28	67	80
Node3	43	72	70
Node N	30	81	66



# Изменение настроек сжатия

The screenshot shows the 'Update Config' interface in Kafka Manager. The breadcrumb trail is 'Clusters / hercules\_prod / Topics / legacy\_logs\_elk\_c2 / Update Config'. The 'Update Config' section has a title bar with a plus icon. Below it, the 'Topic' field is set to 'legacy\_logs\_elk\_c2'. The 'cleanup.policy' field is empty. The 'compression.type' field is highlighted with a yellow box and is currently empty. Below it, the 'delete.retention.ms' field is also empty. The page includes descriptive text for each configuration field.

**Update Config**

Topic  
legacy\_logs\_elk\_c2

cleanup.policy

A string that is either "delete" or "compact" or both. This string designates the retention policy to use on old log segments. The default policy ("delete") will discard old segments when their retention time or size limit has been reached. The "compact" setting will enable [log compaction](#) in the topic.

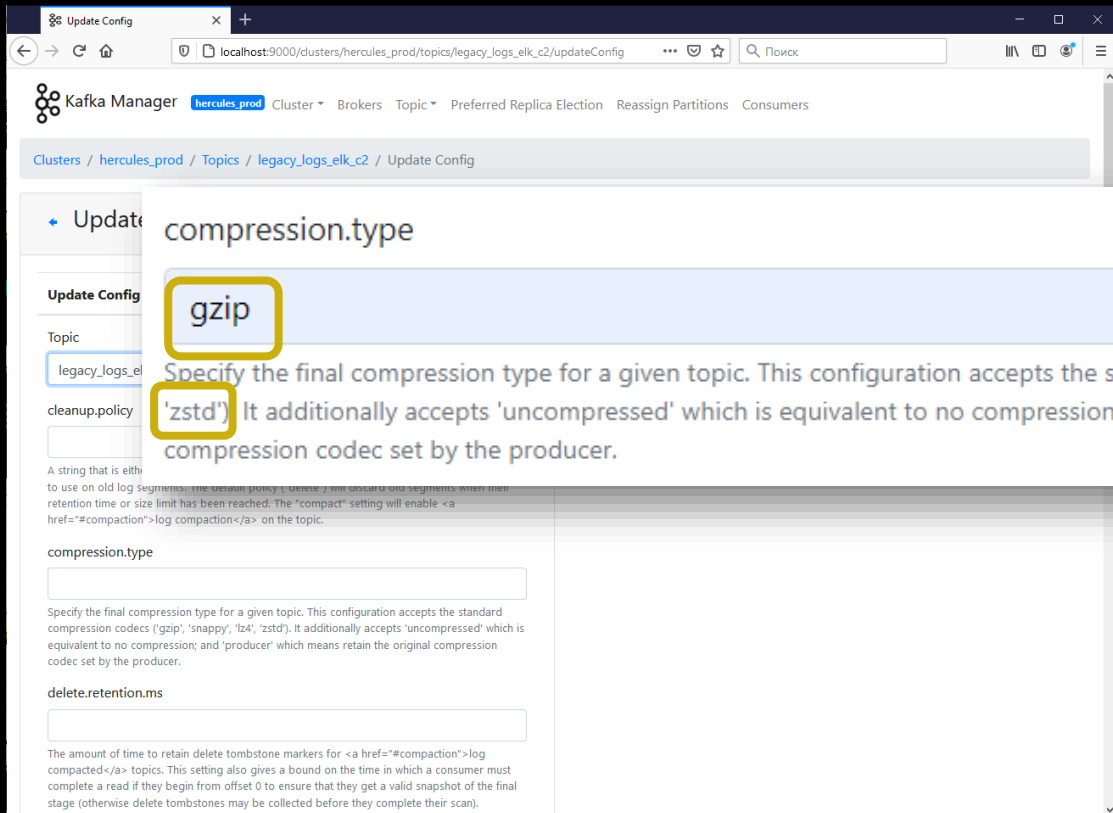
compression.type

Specify the final compression type for a given topic. This configuration accepts the standard compression codes ('gzip', 'snappy', 'lz4', 'zstd'). It additionally accepts 'uncompressed' which is equivalent to no compression; and 'producer' which means retain the original compression codec set by the producer.

delete.retention.ms

The amount of time to retain delete tombstone markers for [compact](#) topics. This setting also gives a bound on the time in which a consumer must complete a read if they begin from offset 0 to ensure that they get a valid snapshot of the final stage (otherwise delete tombstones may be collected before they complete their scan).

# Изменение настроек сжатия



Update Config

localhost:9000/clusters/hercules\_prod/topics/legacy\_logs\_elk\_c2/updateConfig

Kafka Manager hercules\_prod Cluster Brokers Topic Preferred Replica Election Reassign Partitions Consumers

Clusters / hercules\_prod / Topics / legacy\_logs\_elk\_c2 / Update Config

+ Update compression.type

Update Config

Topic

legacy\_logs\_elk\_c2

cleanup.policy

compression.type

gzip

Specify the final compression type for a given topic. This configuration accepts the standard compression codecs ('gzip', 'snappy', 'lz4', 'zstd'). It additionally accepts 'uncompressed' which is equivalent to no compression; and 'producer' which means retain the original compression codec set by the producer.

A string that is either 'delete' or 'compact'. The 'delete' policy will discard old segments when their retention time or size limit has been reached. The 'compact' setting will enable a log compaction on the topic.

compression.type

Specify the final compression type for a given topic. This configuration accepts the standard compression codecs ('gzip', 'snappy', 'lz4', 'zstd'). It additionally accepts 'uncompressed' which is equivalent to no compression; and 'producer' which means retain the original compression codec set by the producer.

delete.retention.ms

The amount of time to retain delete tombstone markers for compacted topics. This setting also gives a bound on the time in which a consumer must complete a read if they begin from offset 0 to ensure that they get a valid snapshot of the final stage (otherwise delete tombstones may be collected before they complete their scan).

# Графики и отчеты

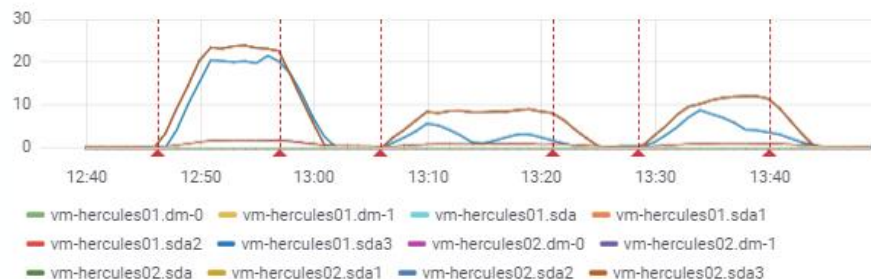
NONE

Jmeter RPS  
GZIP

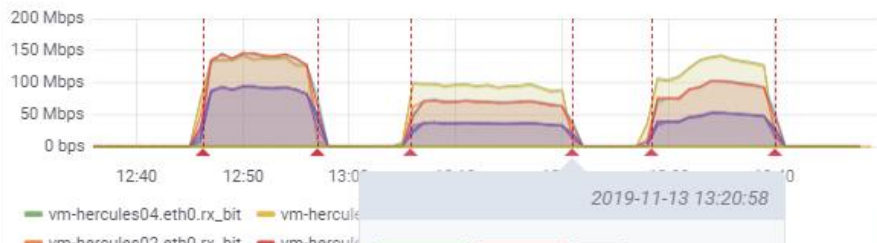
ZSTD



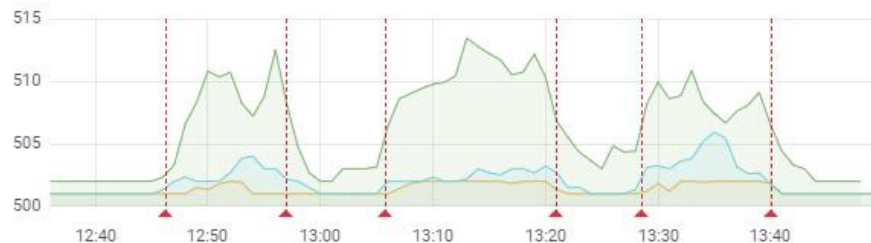
Disk Utilization



Network Utilization, Top 10



Fetch total time ms



# Положительный эффект

Дисковая и сетевая активность упала от 2-х до 4-х раз в зависимости от данных

Не надо покупать оборудование – растем еще в 2 раза

	CPU	LAN	DISC
Node1	33	30	25
Node2	32	33	43
Node3	46	35	30
Node N	36	40	40

# Издержки

## Первоначальные

- Неделя на тестирование(регрессионное, конфигурационное)

## Эксплуатационные

- Незначительно повысилось потребление CPU на клиентах
- Неудобство анализа данных (сжатый трафик похож на шифрованный)

An orange diagonal stripe runs from the top-left corner towards the bottom-right, separating the dark blue background into two sections.

# **Кейс 2.**

## **Высокая утилизация дисков в ClickHouse**

# Clickhouse от Яндекс

ClickHouse - столбцовая система управления базами данных (СУБД) для онлайн обработки аналитических запросов (OLAP).



Высокая производительность на вставку

Тонко настраиваемое сжатие данных

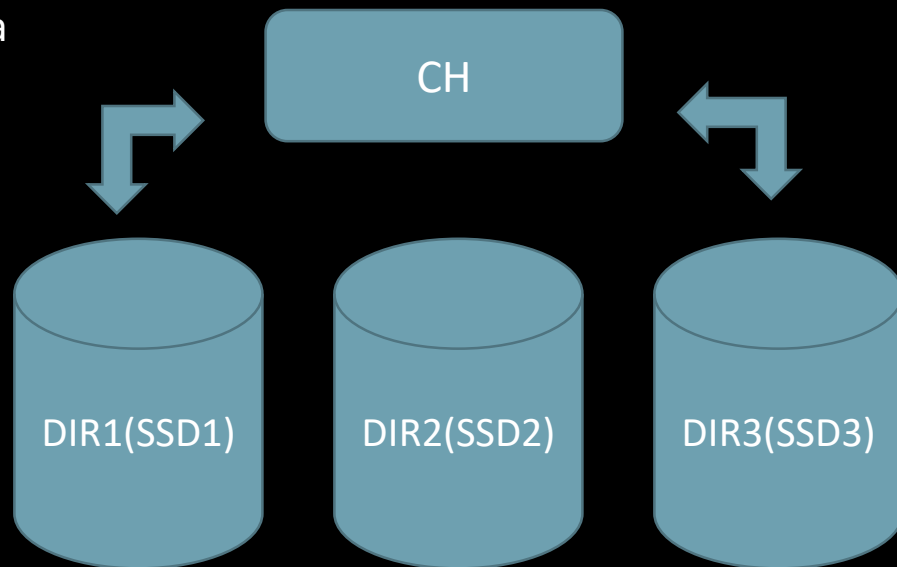
SQL - синтаксис

# Структура записи

Папки, в которых хранятся данные кликхауса, смонтированы на jbod **JBOD** (от англ. Just a bunch of disks, просто пачка дисков)

«/dir1,/dir2,/dir3»

1 файл – 1 диск





# Структура записи

Отдельные диски утилизируются на 100%

Низкая скорость вставки

Растут внутренние очереди

Отказ системы через некоторое время

	1 min
SSD1	90
SSD2	5
SSD3	7

# Структура записи

Отдельные диски утилизируются на 100%

Низкая скорость вставки

Растут внутренние очереди

Отказ системы через некоторое время

	1 min	2 min
SSD1	90	10
SSD2	5	92
SSD3	7	9

# Структура записи

Отдельные диски утилизируются на 100%

Низкая скорость вставки

Растут внутренние очереди

Отказ системы через некоторое время

	1 min	2 min	3 min
SSD1	90	10	6
SSD2	5	92	100
SSD3	7	9	12

# Структура записи

Отдельные диски утилизируются на 100%

Низкая скорость вставки

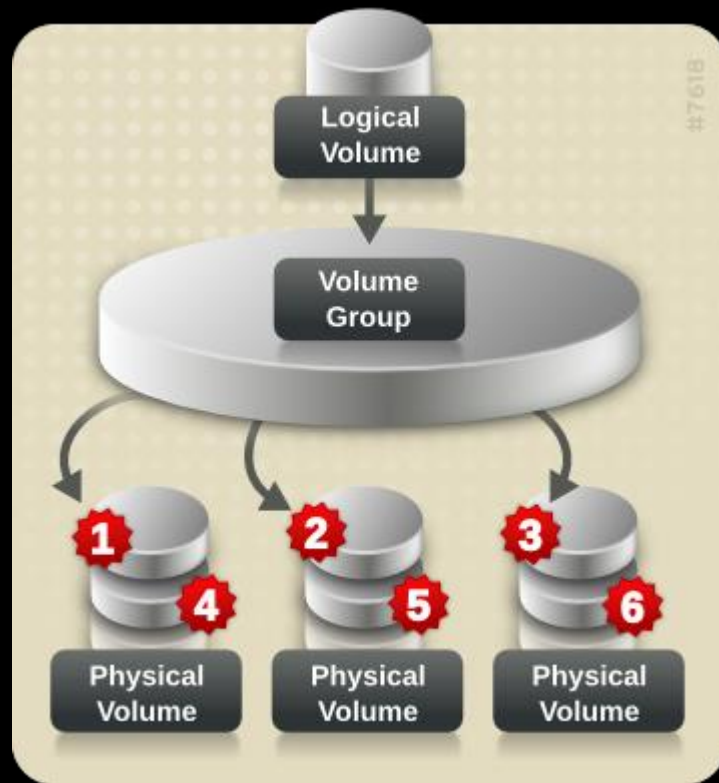
Растут внутренние очереди

Отказ системы через некоторое время

	1 min	2 min	3 min	4 min
SSD1	90	10	6	8
SSD2	5	92	100	9
SSD3	7	9	12	92

# Striped LVM/RAID0

- Логический том с чередованием
- Для приложения – один диск (директория)
- Параллельная запись на несколько дисков



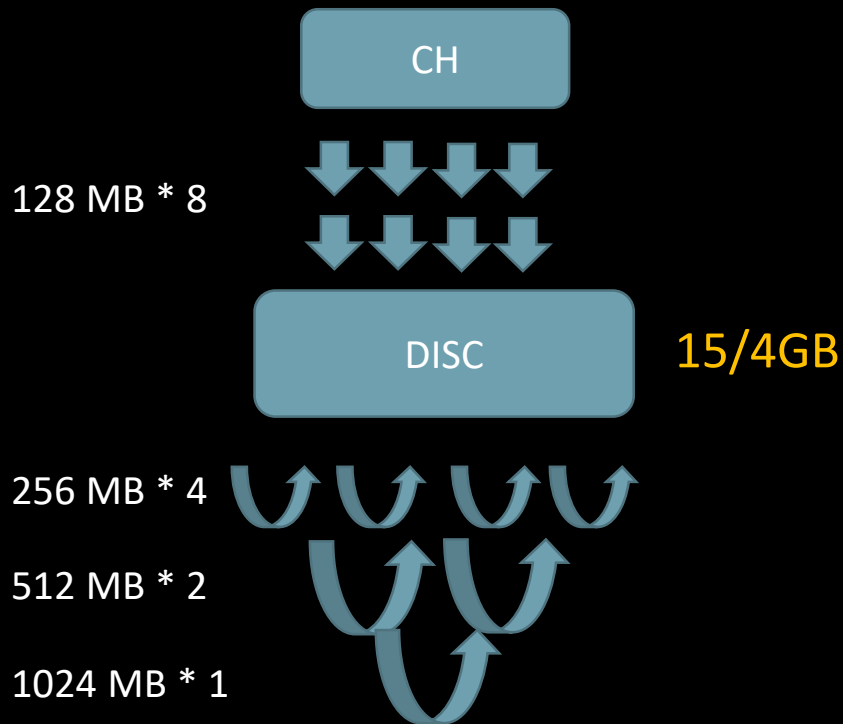
# Стало

Равномерное распределение нагрузки по дискам

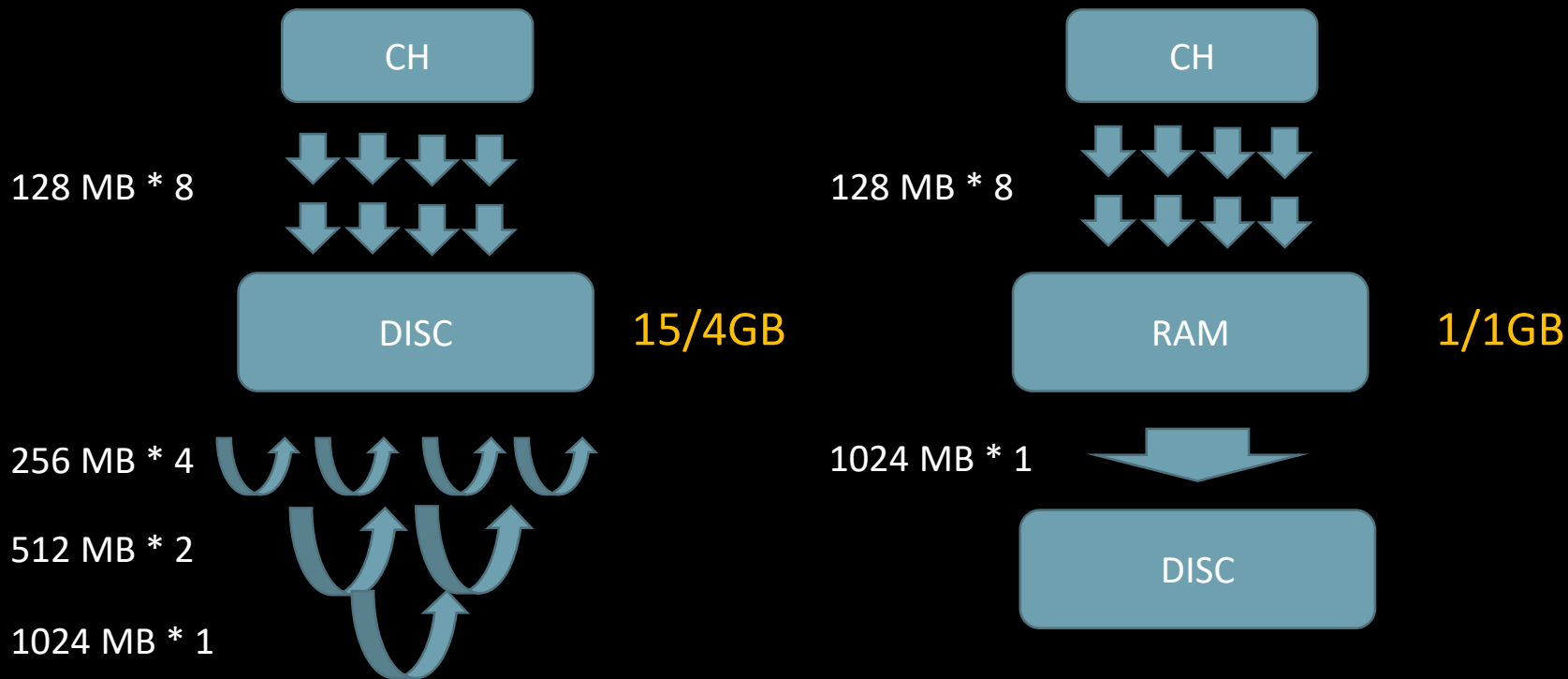
Высокая производительность на вставку – очередь не копится

	1 min	2 min	3 min	4 min
SSD1	60	55	63	65
SSD2	60	56	63	65
SSD3	60	56	63	66

# Большое количество обращений к диску



# In-memory Table(Buffer Engine)





## Уменьшение количества дисковых IO

	1 min	2 min	3 min	4 min
SSD1	60	55	63	65
SSD2	60	56	63	65
SSD3	60	56	63	66

	1 min	2 min	3 min	4 min
SSD1	25	31	19	22
SSD2	24	31	19	22
SSD3	25	31	20	21

# Выгода

Смогли поднять производительность сервера в 2 раза, освободив некоторые сервера под другие нужды

Отпала необходимость в приобретении высокопроизводительных дорогих дисков

Чтение свежих данных ускорилось, потому что они хранятся в in-memory таблице

# Издержки

## Первоначальные


- Исследование архитектуры
- Несколько недель тестирования

## Эксплуатационные

- Увеличение сложности архитектуры !!!

## Риски

- Возможная потеря данных в in-memory таблице при перезапуске
- Lvm – при отказе одного диска теряются все данные массива

An orange diagonal stripe runs from the top-left corner towards the bottom-right, separating the dark background into two sections.

# **Кейс 3.**

# **Жизненный цикл**

# **индексов**

# **ElasticSearch**

# ElasticSearch

Распределенный поисковой движок

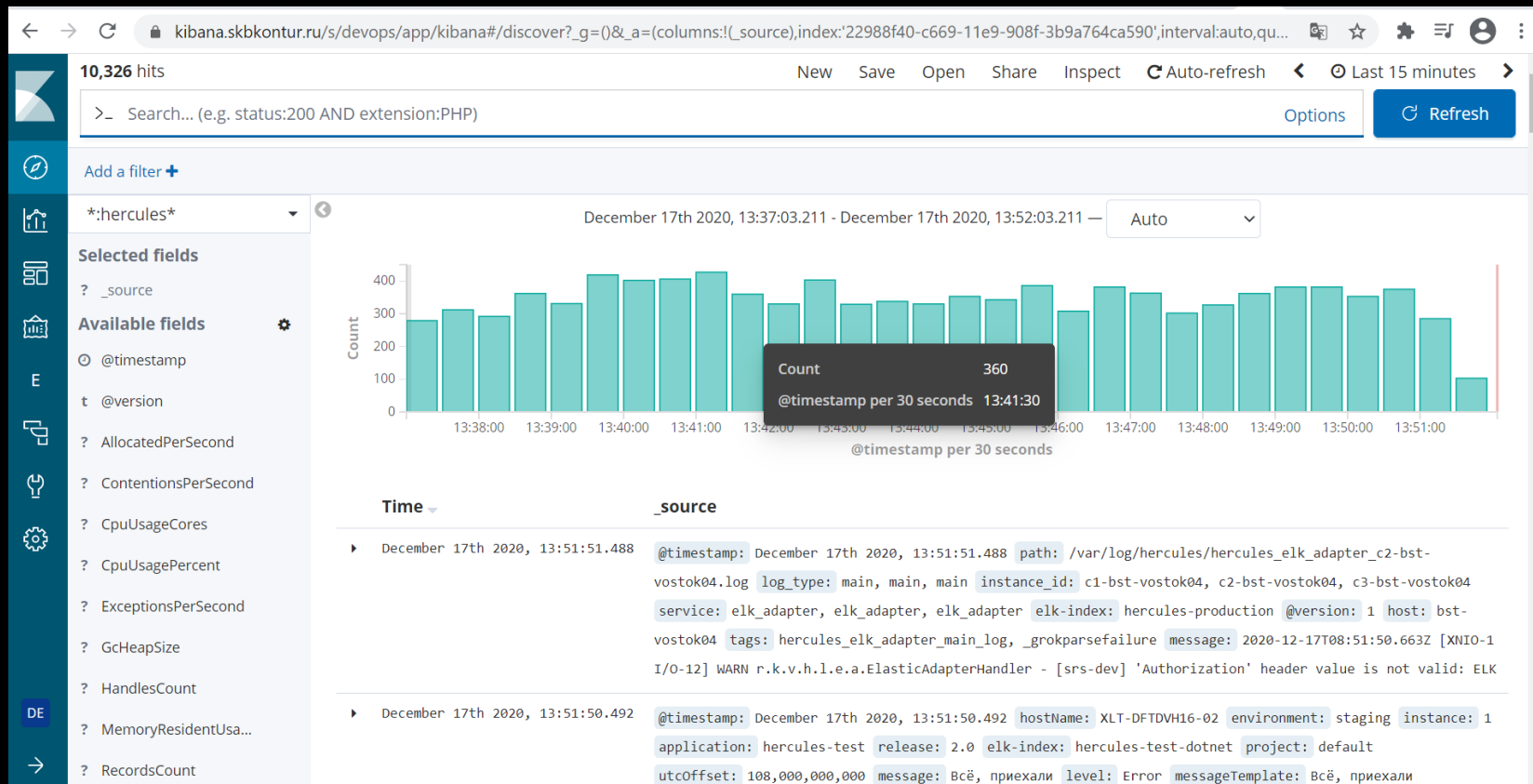
Быстрый полнотекстовый поиск

RestFull API

noSQL



# ElasticSearch



# ES Index

Индекс – как будто таблица в БД

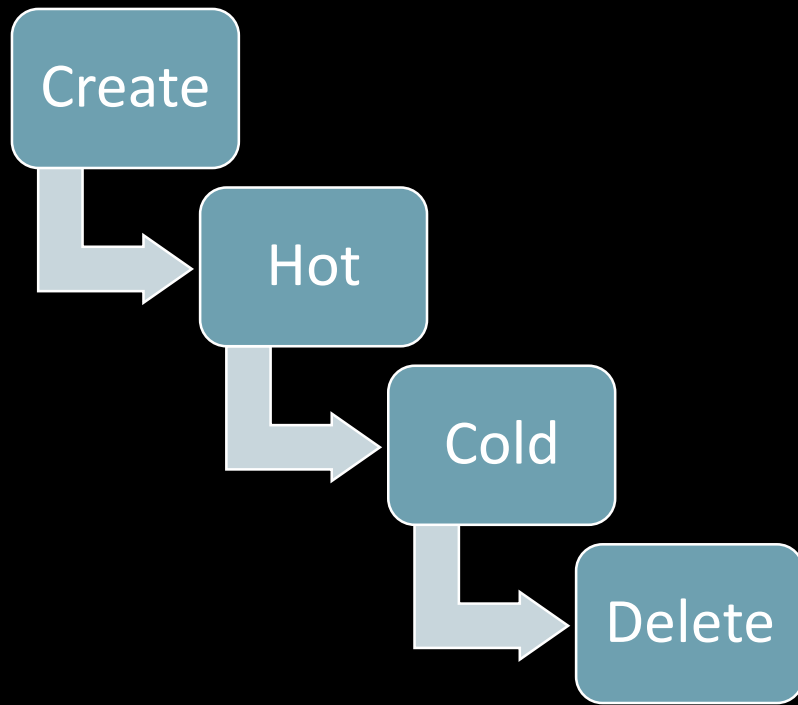
Имя

- sentry-prod-kafka-2021.06.17
- sentry-prod-kafka-2021.06.18
- elk-staging-curator-2021.06.18

Index pattern

- sentry-prod-kafka-\*
- elk-\*-curator\*

# ES index lifecycle





# Неравномерность размеров индексов

Для каждой команды каждый день создается несколько индексов.

Сотни мелких индексов(1 MB) – много накладных расходов на обслуживание и мониторинг

Огромными индексами больших команд (2 TB) - сложно управлять(дорогой переезд на разные фазы жизненного цикла, дорогое перемещение при балансировке)



# Равноразмерные индексы

Создание индексов не ежедневно а при соблюдении условий – 30 дней или 30 гигабайт

Имя

- sentry-prod-kafka-0000001
- sentry-prod-kafka-0000002
- elk-staging-curator-0000001



# Выгоды

- Более ровное распределение индексов, они одинакового размера
- Нет слишком больших и слишком маленьких
- Улучшение эксплуатационных свойств системы

# Издержки

## На внедрение

- Несколько недель на исследование и тестирование
- Переписывание скриптов для управления жизненным циклом
- Изменение логики работы со стороны клиентов на вставку

## Эксплуатационные

- Нужна длительная параллельная поддержка обоих жизненных циклов
- Усложнение сопоставления индекса и данных в них

An orange diagonal stripe runs from the top-left corner towards the bottom-right, separating the orange background on the left from the black background on the right.

# **Оптимизации и тестирование**

# Требования к кластеру

- Функциональность
- Производительность
- Отказоустойчивость
- Удобство эксплуатации

\* Цена (Железо, люди)

# Оптимизации

Оптимизация — процесс максимизации выгодных характеристик, соотношений (например, оптимизация производственных процессов и производства), и минимизации расходов.

Более эффективное использование ресурсов обычно требует компромиссов (tradeoff) — один параметр оптимизируется за счёт других.

# Оптимизации

- Настройки(сжатие, размеры буферов, очередей, количество потоков, размер пачки сообщений)
- Настройки ресурсов(выделение памяти, дисков, сети, процессора)
- Настройки шардирования и жизненного цикла(Hot-Cold фазы, размер шард, количество шард)
- Оптимизация передачи и хранения (буферные таблицы, индексы, форматы передачи данных)
- Локализация сетевого трафика



# Настройки

```
# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space

-Xms20g
-Xmx20g

#####
## Expert settings
#####
##
## All settings below this section are considered
## expert settings. Don't tamper with them unless
## you understand what you are doing
##
#####

## GC configuration
-XX:+UseG1GC
-XX:MaxGCPauseMillis=300
-XX:InitiatingHeapOccupancyPercent=30

-XX:G1ReservePercent=25
```

# Тестирование

- Определяем узкие места
- Фиксируем состояние системы до оптимизации
- Применяем оптимизацию
- Проводим тестовые испытания с фиксацией состояния системы
- Анализируем результат
- Оцениваем риски

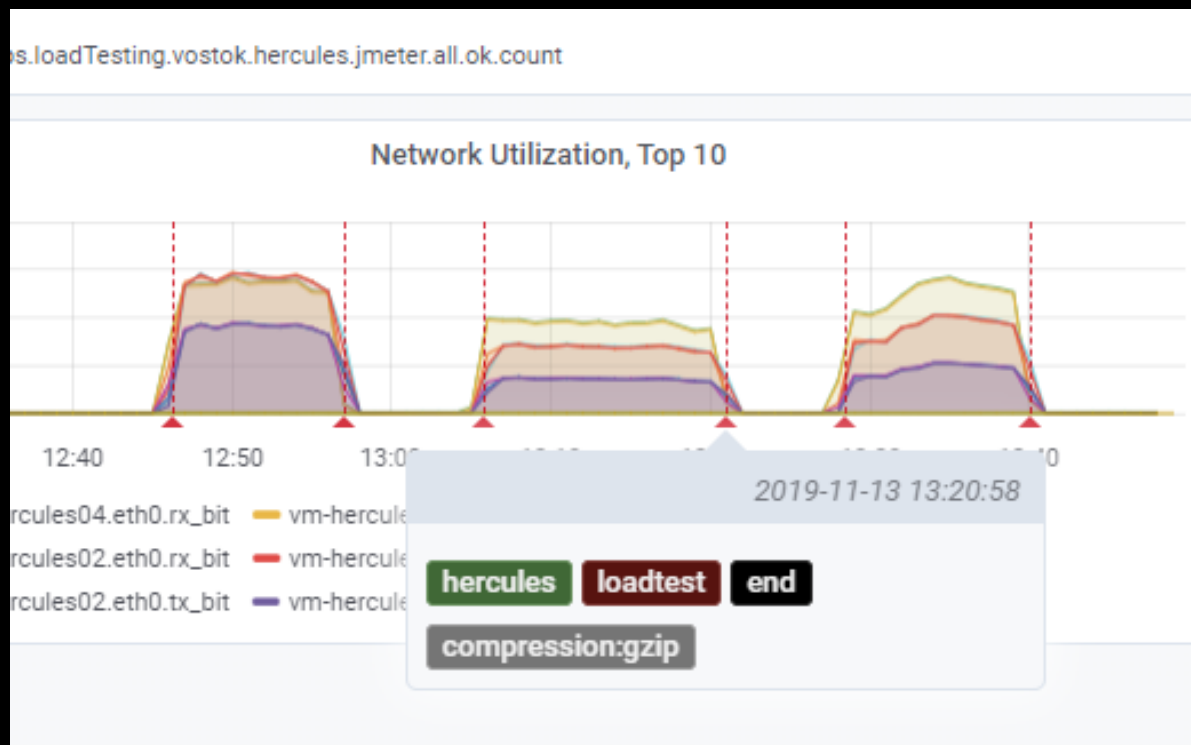
# Тестирование

- Функциональное (регрессионное)
- Производительности (нагрузочное, стресс)
- Эксплуатационное
- На отказ и восстановление
- Объемное



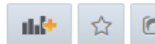
# **Три рекомендации по тестированию**

# Графики и отчеты



# Графики и отчеты

Other > Hercules Kafka Dashboard For Load Tests



Prefix hercules

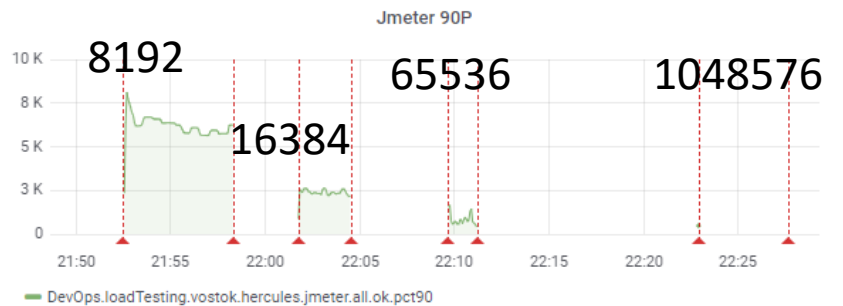
Environment testing

Host All

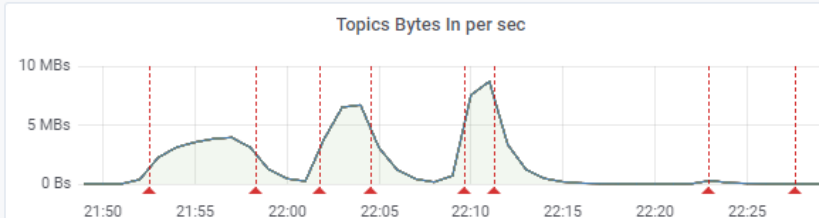
Group All

Topic All

Deploy ☒



## Topic Stats



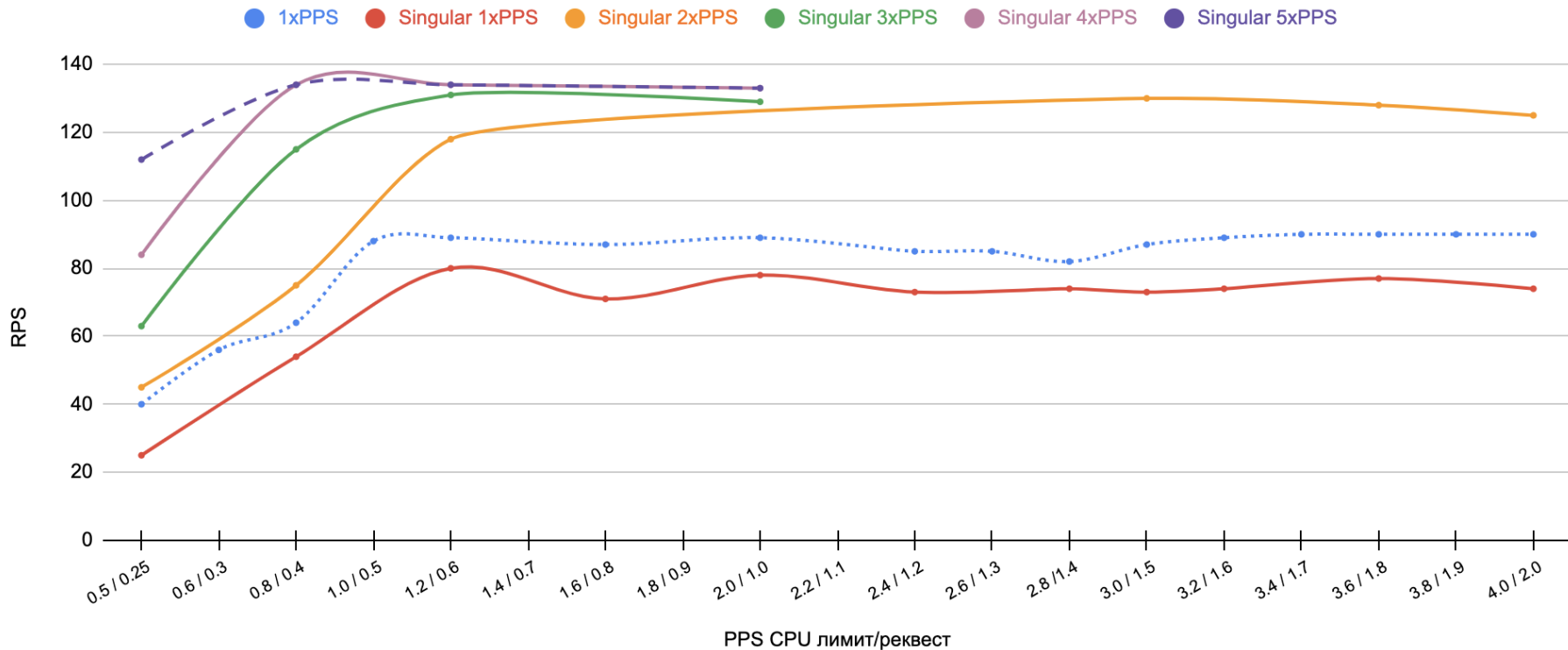
## Графики и отчеты

#31	Success	No changes	vm-tank1.sys-msk	30 Aug 18 18:39	2m:46s	2	...
#30	Success	No changes	vm-tank1.sys-msk	30 Aug 18 13:29	3m:23s	2	...
#29	Success	No changes	vm-tank1.sys-msk	30 Aug 18 13:25	2m:12s	2	...
#28	Success	No changes	vm-tank1.sys-msk	30 Aug 18 13:19	3m:18s	2	...
#27	Success	No changes	vm-tank1.sys-msk	30 Aug 18 13:12	3m:23s	2	...
#26	Canceled (Exit code 137 (Step: Run gatliting it test (Command L...	No changes	vm-tank1.sys-msk	30 Aug 18 13:11	18s		...
#25	Success	No changes	vm-tank1.sys-msk	30 Aug 18 12:35	7m:37s		...
#24 ☆	Canceled (Exit code 137 (Step: Run gatliting it test (Command L...	No changes	vm-tank1.sys-msk	30 Aug 18 11:50	41m:18s	async	undertow
#23	Success	No changes	vm-tank1.sys-msk	30 Aug 18 11:26	7m:58s		...
#22	Success	Махетов Сергей Леонидович: 1	vm-tank1.sys-msk	30 Aug 18 11:17	7m:06s		...
#21	Success	Махетов Сергей Леонидович: 1	vm-tank1.sys-msk	29 Aug 18 23:17	6m:33s		...
#20	Success	Махетов Сергей Леонидович: 1	vm-tank1.sys-msk	29 Aug 18 16:11	7m:16s		...
#19	Canceled (Exit code 137 (Step: Run gatliting it test (Command L...	Махетов Сергей Леонидович: 1	vm-tank1.sys-msk	29 Aug 18 16:09	25s		...
#18 ★	Canceled (Exit code 137 (Step: Run gatliting it test (Command L...	Махетов Сергей Леонидович: 1	vm-tank1.sys-msk	29 Aug 18 16:02	5m:14s		...

Name	Value passed to build
generator_args	-Dusers=20 -Dloops=50 -Devents=100 -Durl=http://vostok06:6307
report_text	Send sync undertow after throttling
test_class_name	ElkPipeTestIt
tmp	-Drpsshedule="line 2000 10;hold 10;line 0 10"

# Графики и отчеты

## RPS Predict Proxy





# Графики и отчеты

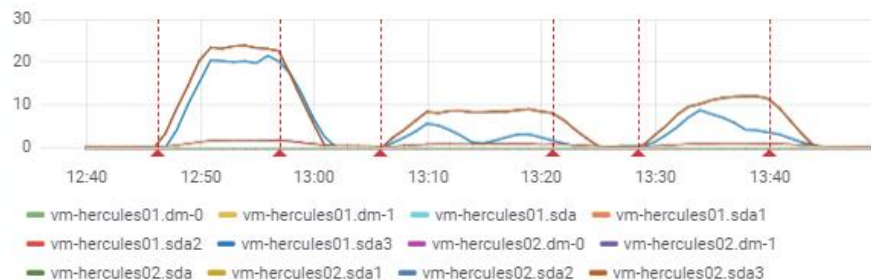
NONE

Jmeter RPS  
GZIP

ZSTD



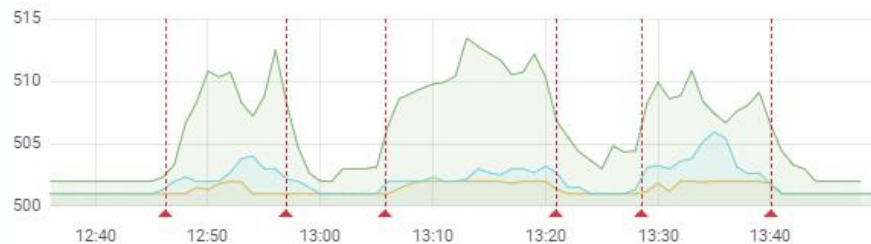
Disk Utilization



Network Utilization, Top 10



Fetch total time ms



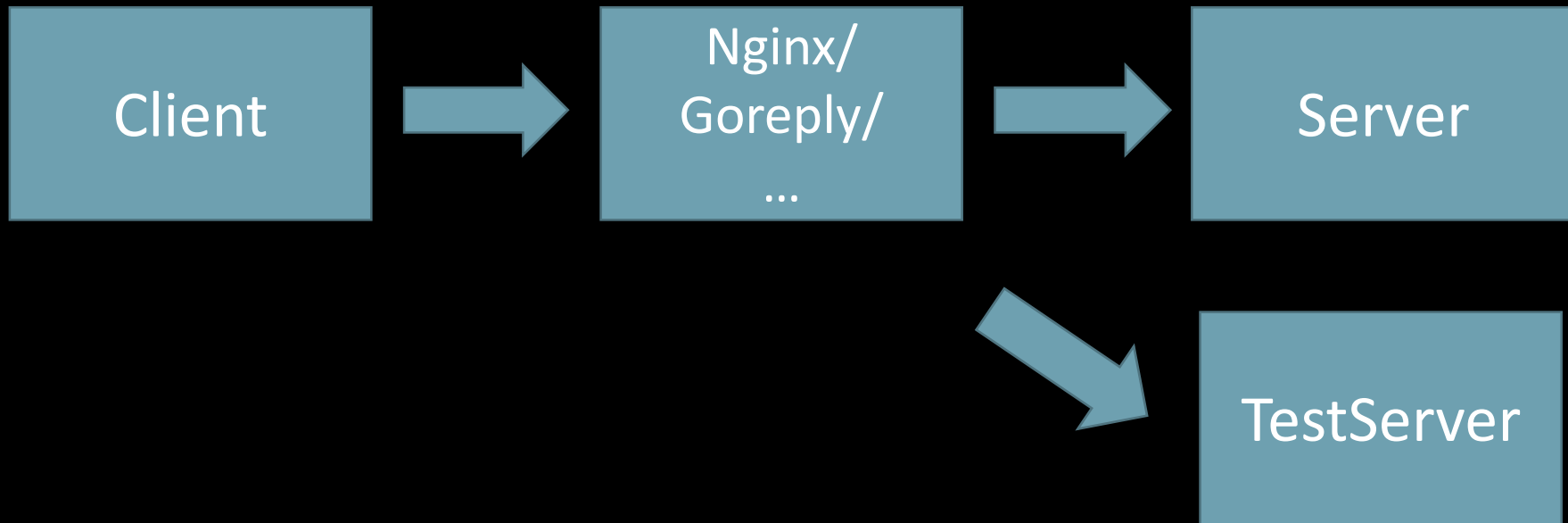




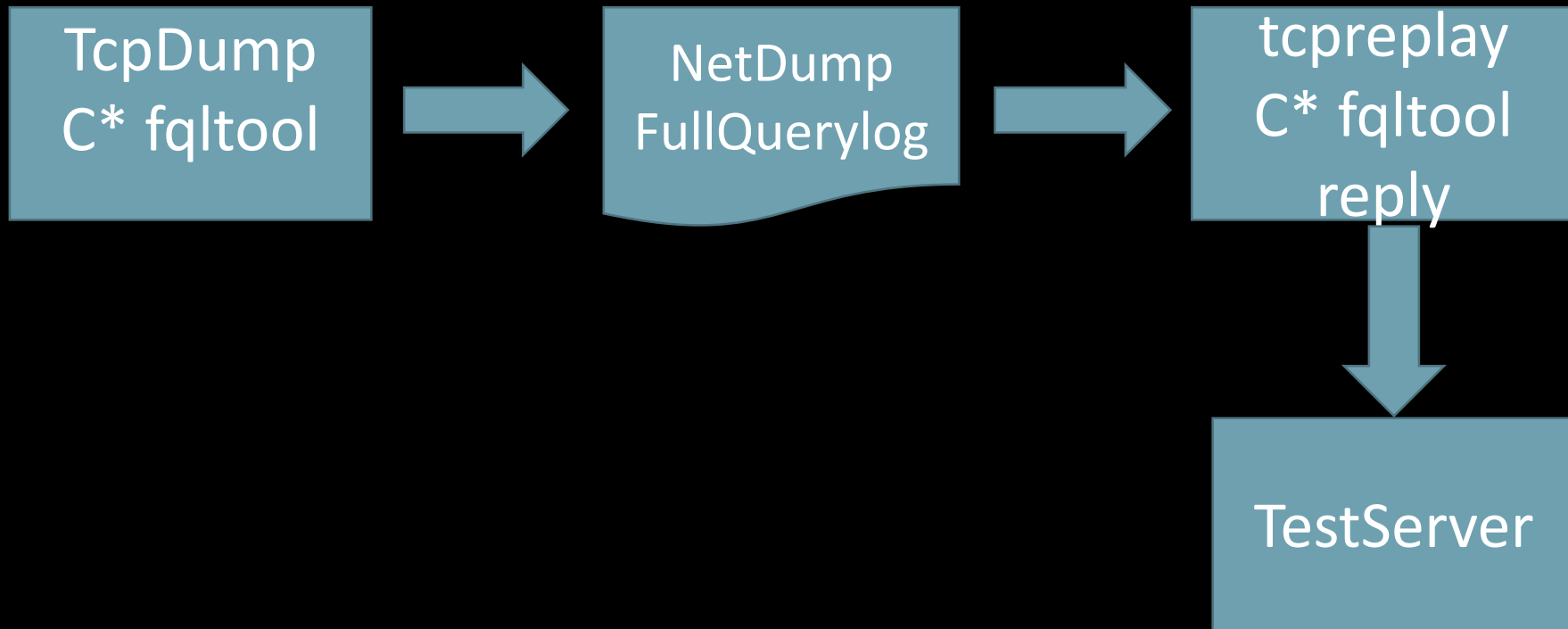
# Протоколы передачи и хранения

Текстовый	Бинарный
XML, JSON, PLAIN TEXT	Apache Avro, Google Protocol Buffers, BSON
Человекочитаемость	Высокая компактность, скорость обработки
Легкая генерация, множество инструментов	Дополнительные затраты на генерацию

## Дублируем с прода



## Повторяем с прода



# Wireshark export bytes

```
"tcp.payload_raw": [  
  "485454502f312e3120323030204f4b0d0a5365727665723a20616b6b612d687474702  
  54,  
  1460,  
  0,  
  30  
],  
"tcp.payload": "48:54:54:50:2f:31:2e:31:20:32:30:30:20:4f:4b:0d:0a:53:65",  
"tcp.reassembled_in": "128",  
"tcp.segment_data_raw": [  
  "485454502f312e3120323030204f4b0d0a5365727665723a20616b6b612d687474702  
  54,  
  1460,  
  0,  
  30  
],  
"tcp.segment_data": "48:54:54:50:2f:31:2e:31:20:32:30:30:20:4f:4b:0d:0a:"
```

# Идентичность стендов тестирования и эксплуатации

- Неожиданное поведение
- 4 виртуальных ядра != 4 реальных
- Большой объем свободной памяти – отдается под page cache
- Сеть 10 Gbps работает не так, как 1 Gbps
- ...



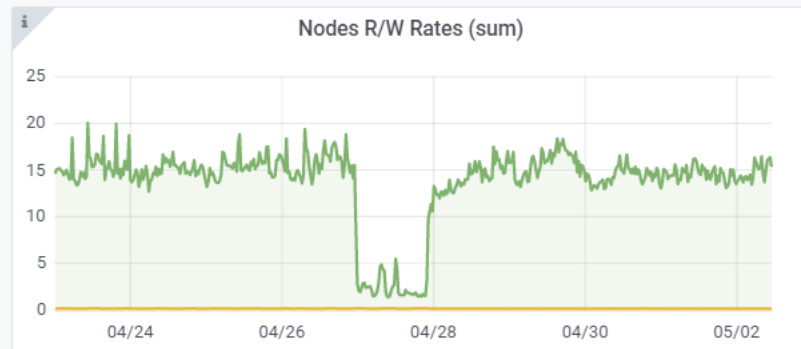
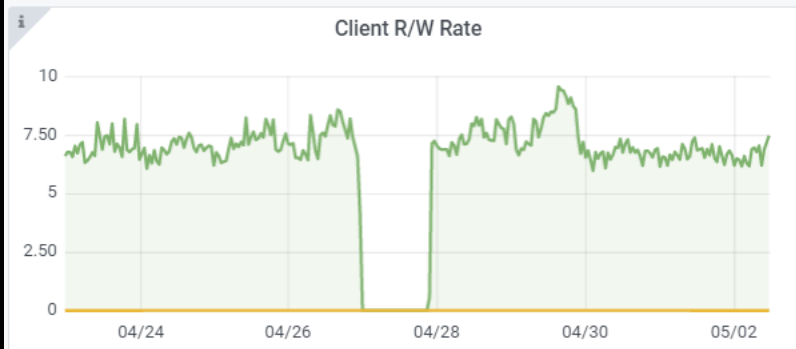
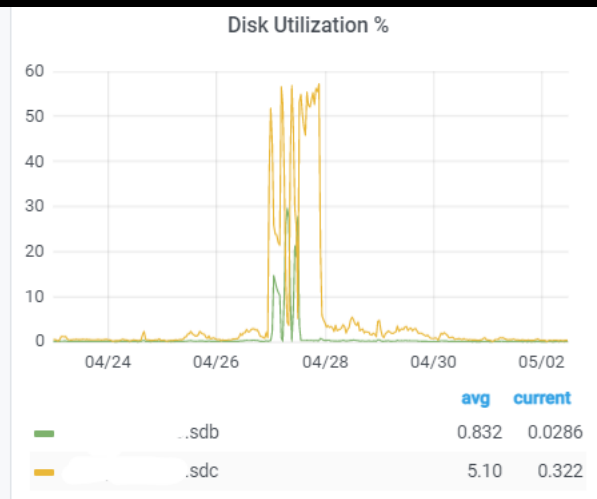
# Идентичность стендов тестирования и эксплуатации



# **Дополнительные особенности работы БД**

# Отложенные операции в БД

- Backup
- Merge, optimize, compaction
- Переезд на другие фазы

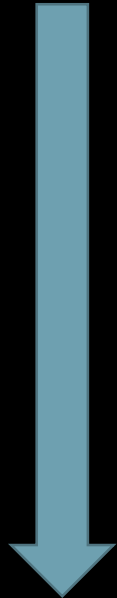


## Фазы хранения

- А что, если хранить не все данные на SSD, а редкоиспользуемые данные отправить на HDD?
- Внутри одной ноды, одного сервера, одного кластера, между разных кластеров

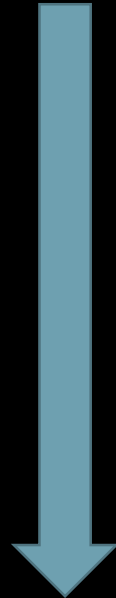
# Фазы хранения

Скорость доступа



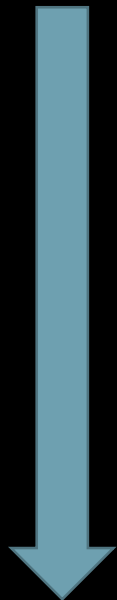
VERY HOT/RAM
HOT/SSD
HOT/SSD
WARM/HDD
WARM/HDD
COLD/HDD
COLD/HDD
COLD/HDD
FROZEN/LAN
FROZEN/LAN

Цена хранения



# Фазы хранения

Скорость доступа  
Цена хранения



VERY HOT/RAM
HOT/SSD
HOT/SSD
WARM/HDD
WARM/HDD
COLD/HDD
COLD/HDD
COLD/HDD
FROZEN/LAN
FROZEN/LAN

- Цена перемещения?
- Цена выполнения служебных операций?

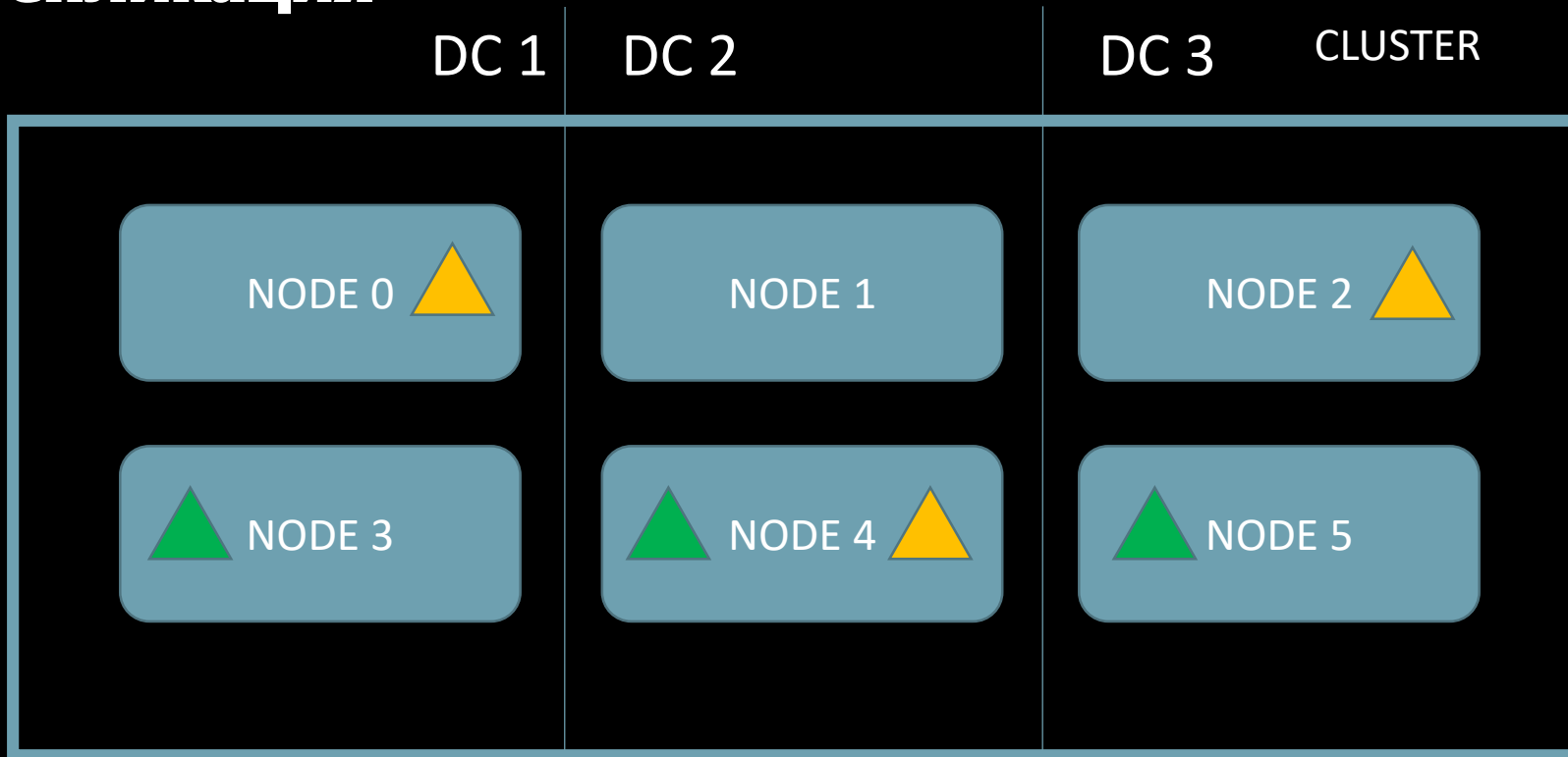
# **Репликация и шардирование**



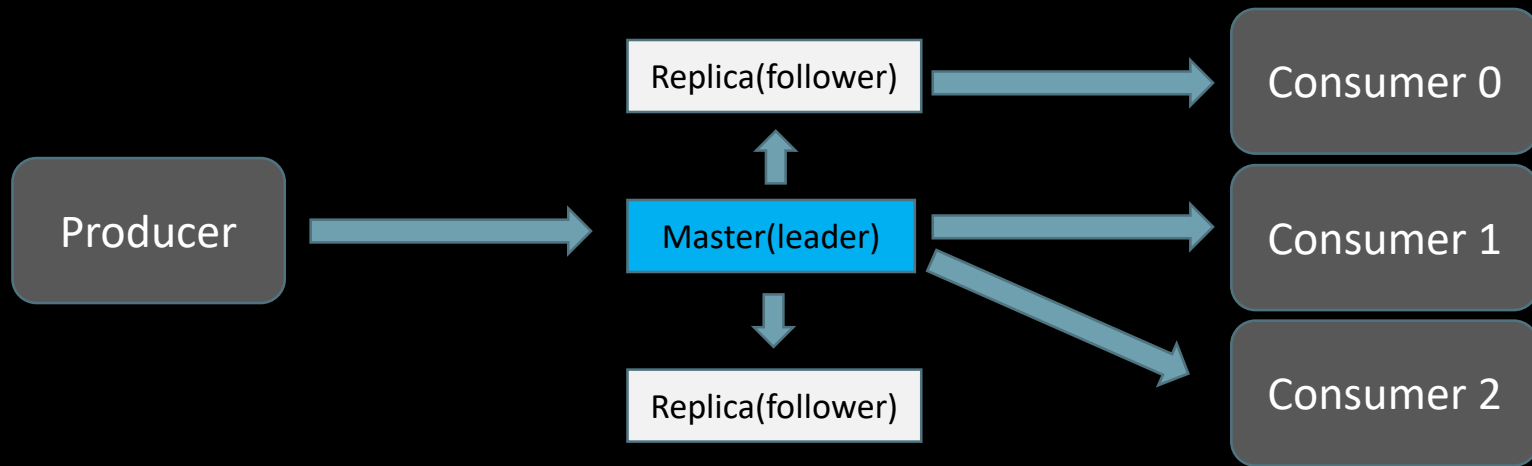
# Репликация

- Хранение копий данных на разных узлах кластера
- Обеспечение сохранности данных в случае сбоя системы
- Увеличение количества узлов для чтения данных
- Географическая локализация данных – ближе к читателям

# Репликация



# Репликация



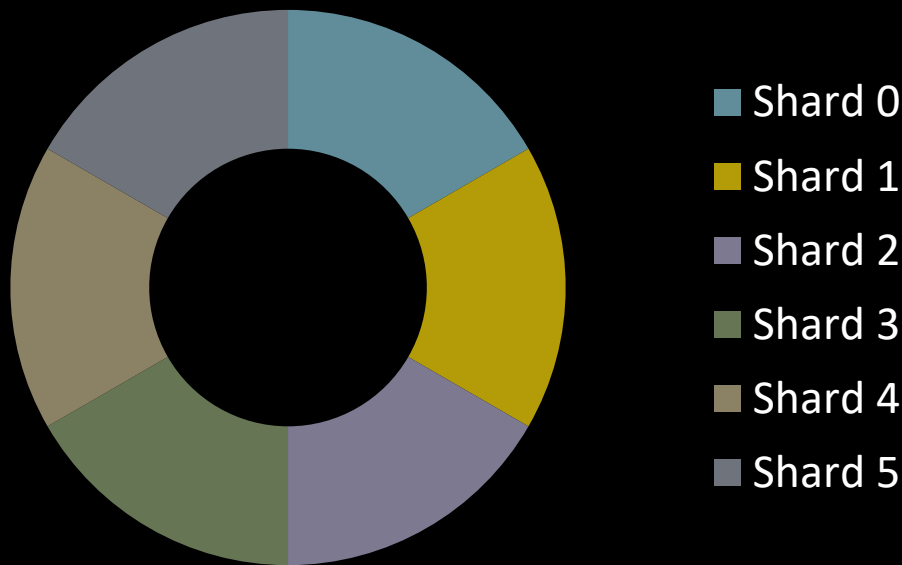
## Репликация. Что тестировать

- Время и затраты ресурсов на восстановление
- Локализация потребления данных, объем трафика
- Работу механизмов выбора лидерства

# Шардирование

- Шардирование
- Секционирование
- Партиционирование

Разделение данных на кусочки



# Шардирование

Year	Data
2021	...
2021	...
2021	...
2021	...
2020	...
2020	...

# Шардирование вертикальное

Year	Data
2021	...
2021	...
2021	...
2021	...
2020	...
2020	...

Year	Data	Shard
2021	...	1
2021	...	1
2021	...	1
2021	...	1
2020	...	0
2020	...	0

# Шардирование вертикальное и горизонтальное

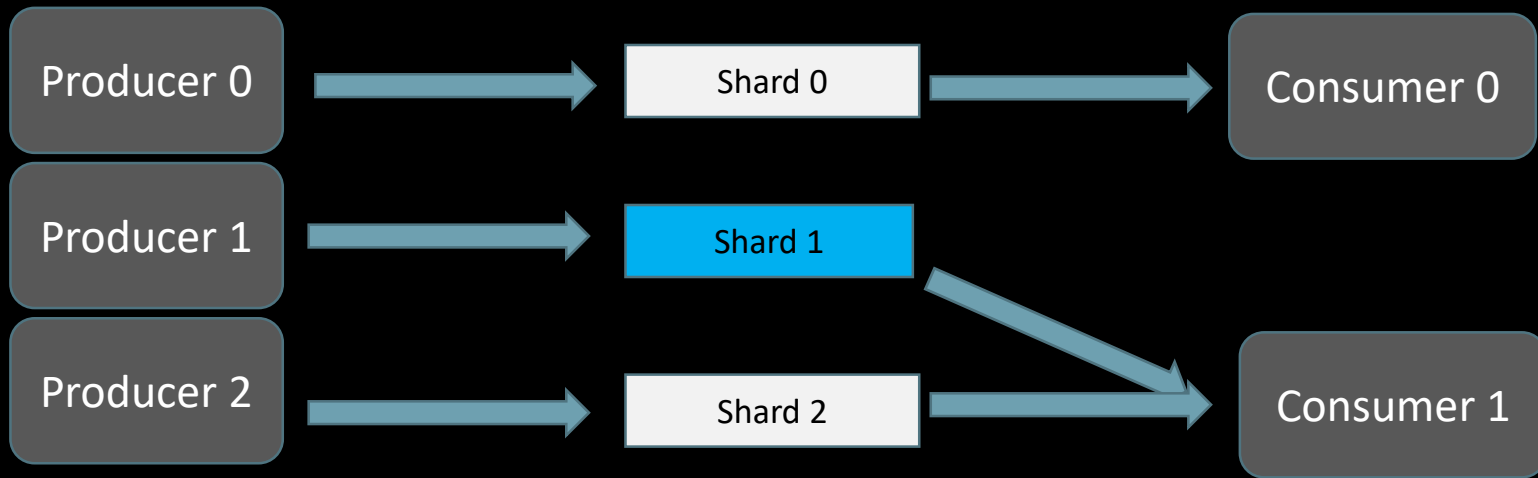
Year	Data
2021	...
2021	...
2021	...
2021	...
2020	...
2020	...

Year	Data	Shard
2021	...	1
2021	...	1
2021	...	1
2021	...	1
2020	...	0
2020	...	0

Year	Data	Shard
2021	...	0
2021	...	1
2021	...	0
2021	...	1
2020	...	0
2020	...	1

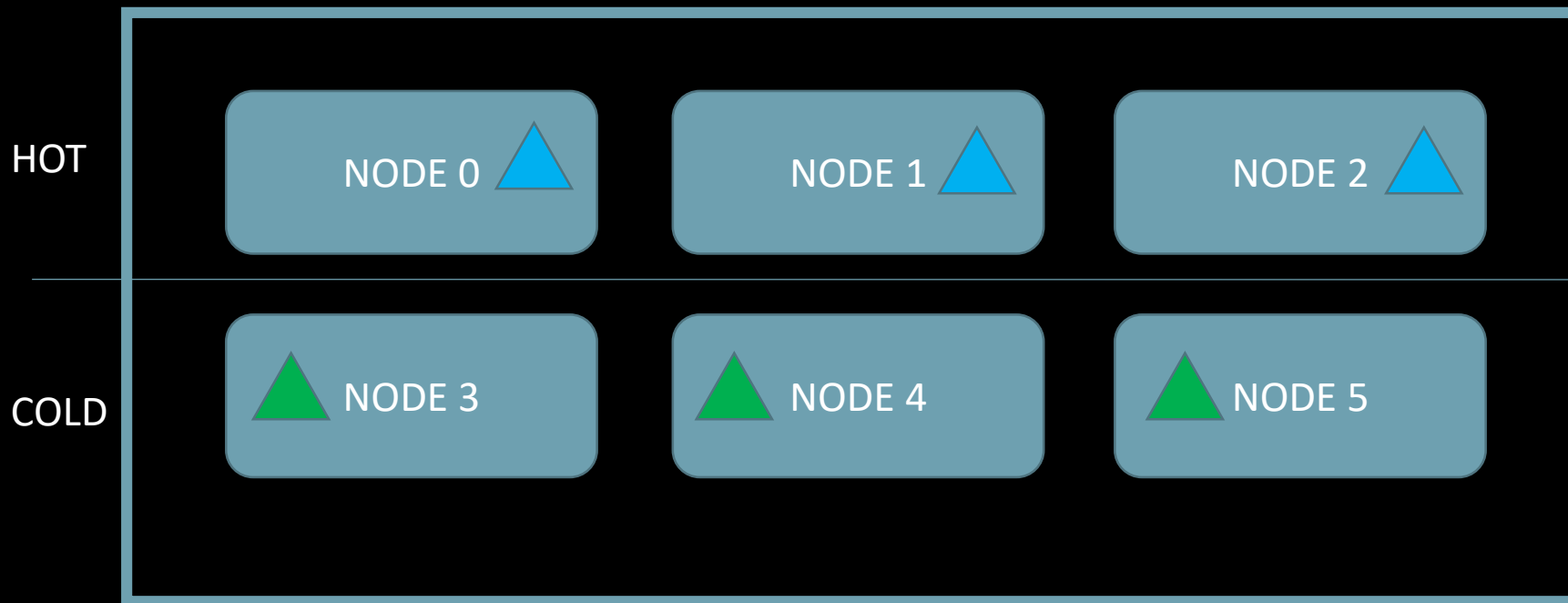


# Шардирование

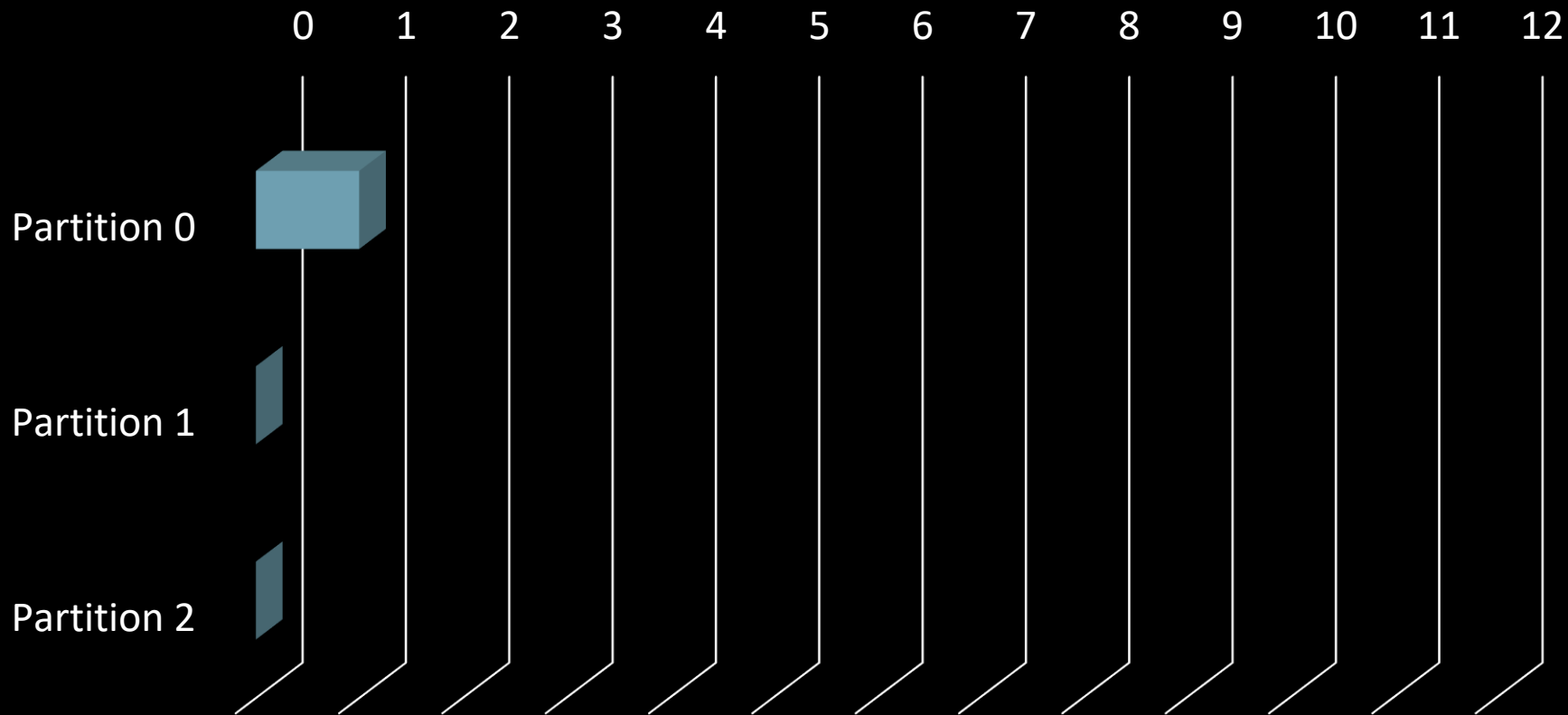


# Шардирование

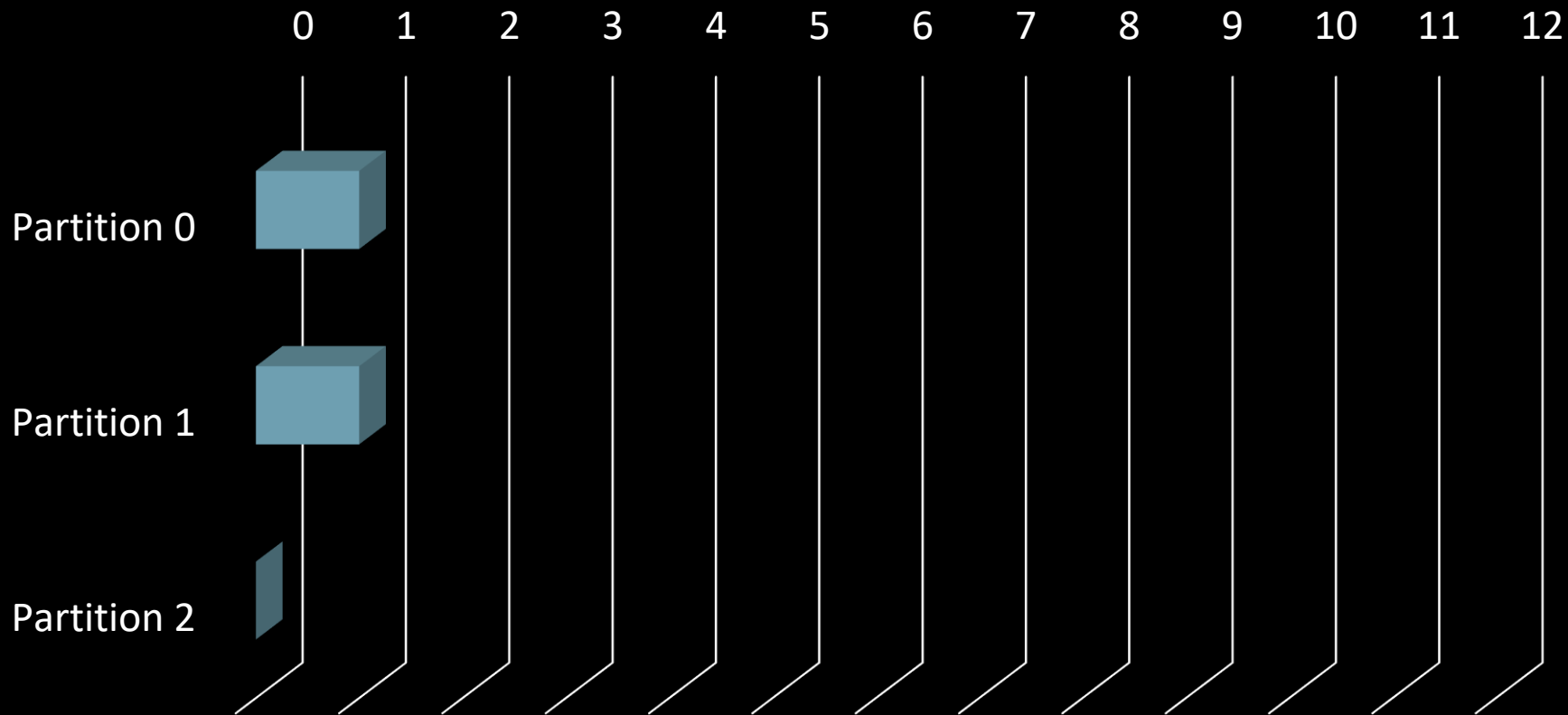
CLUSTER



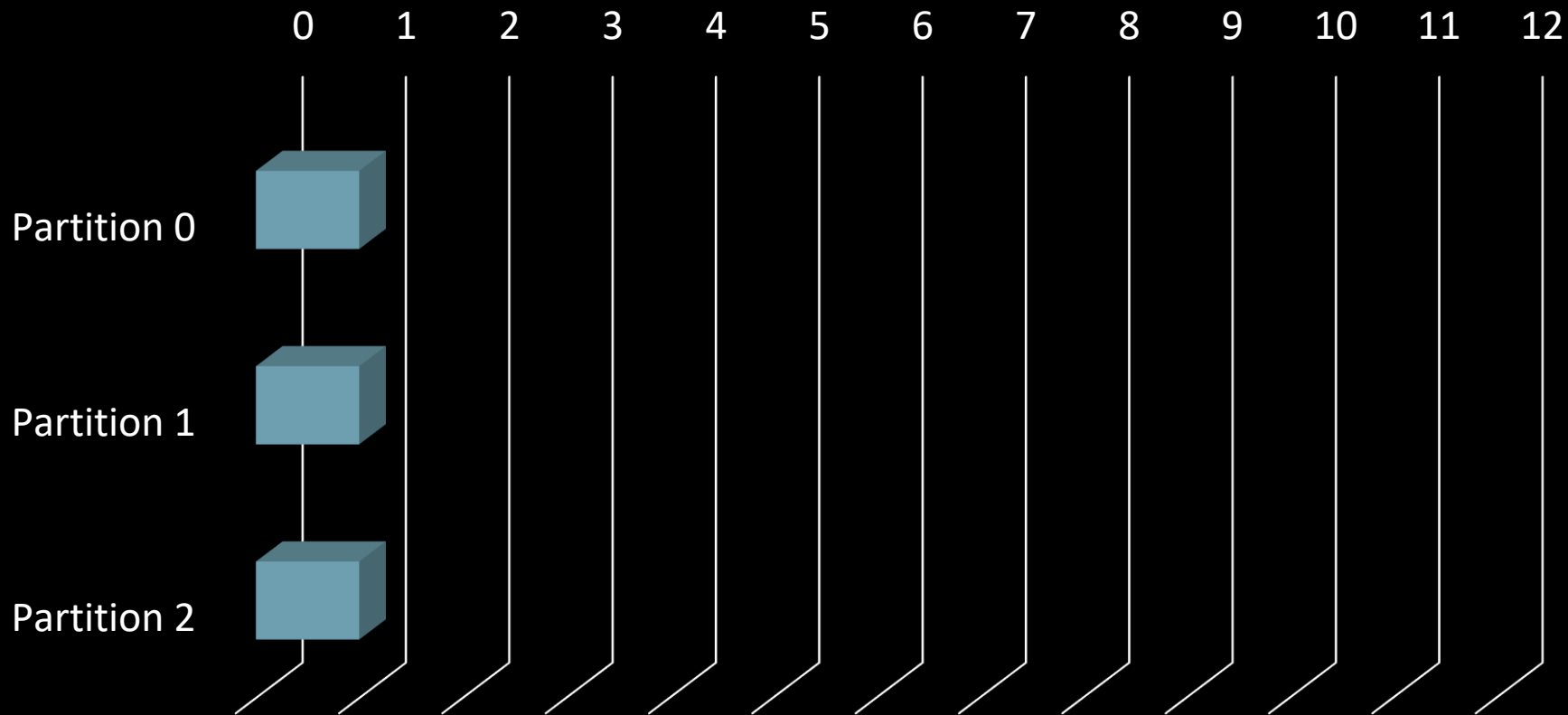
# Анатомия топика



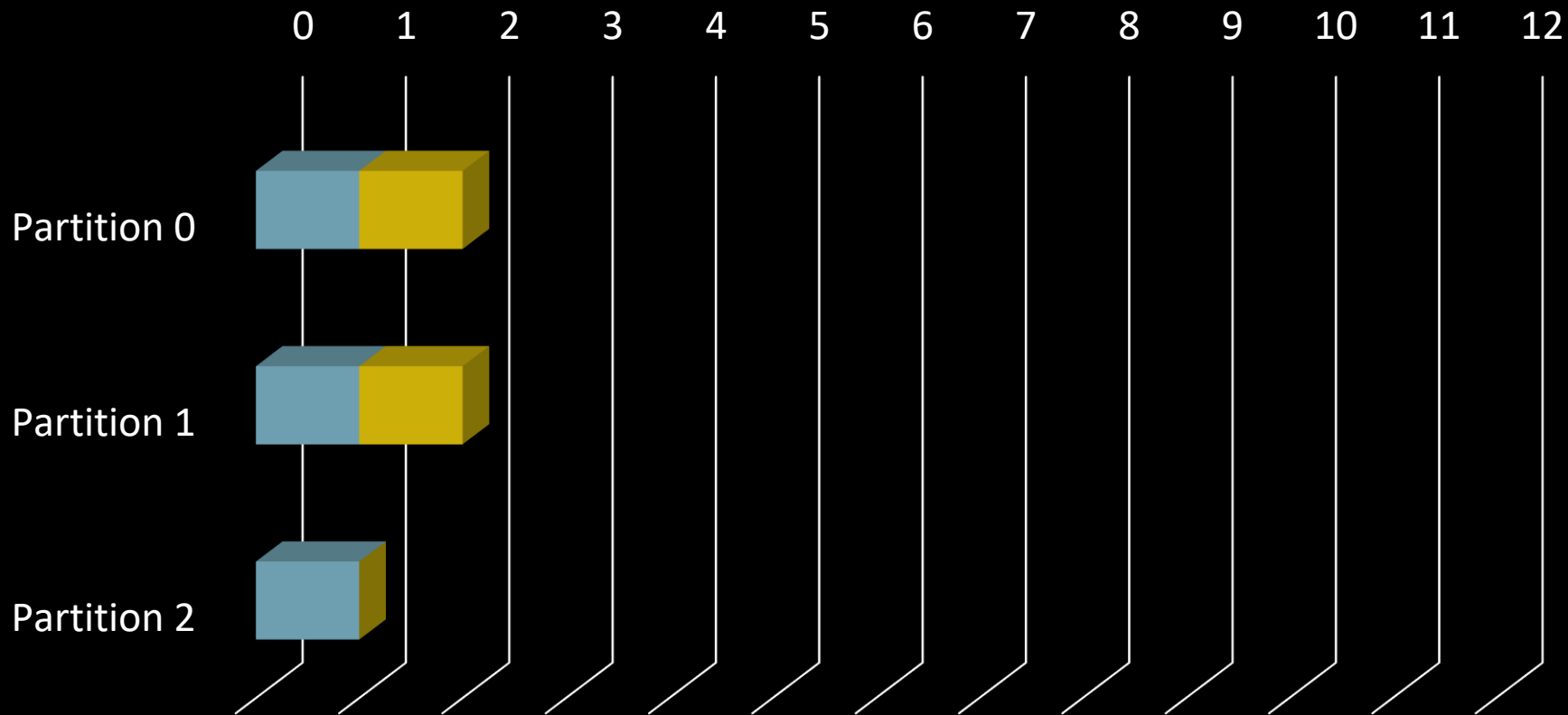
# Анатомия топика



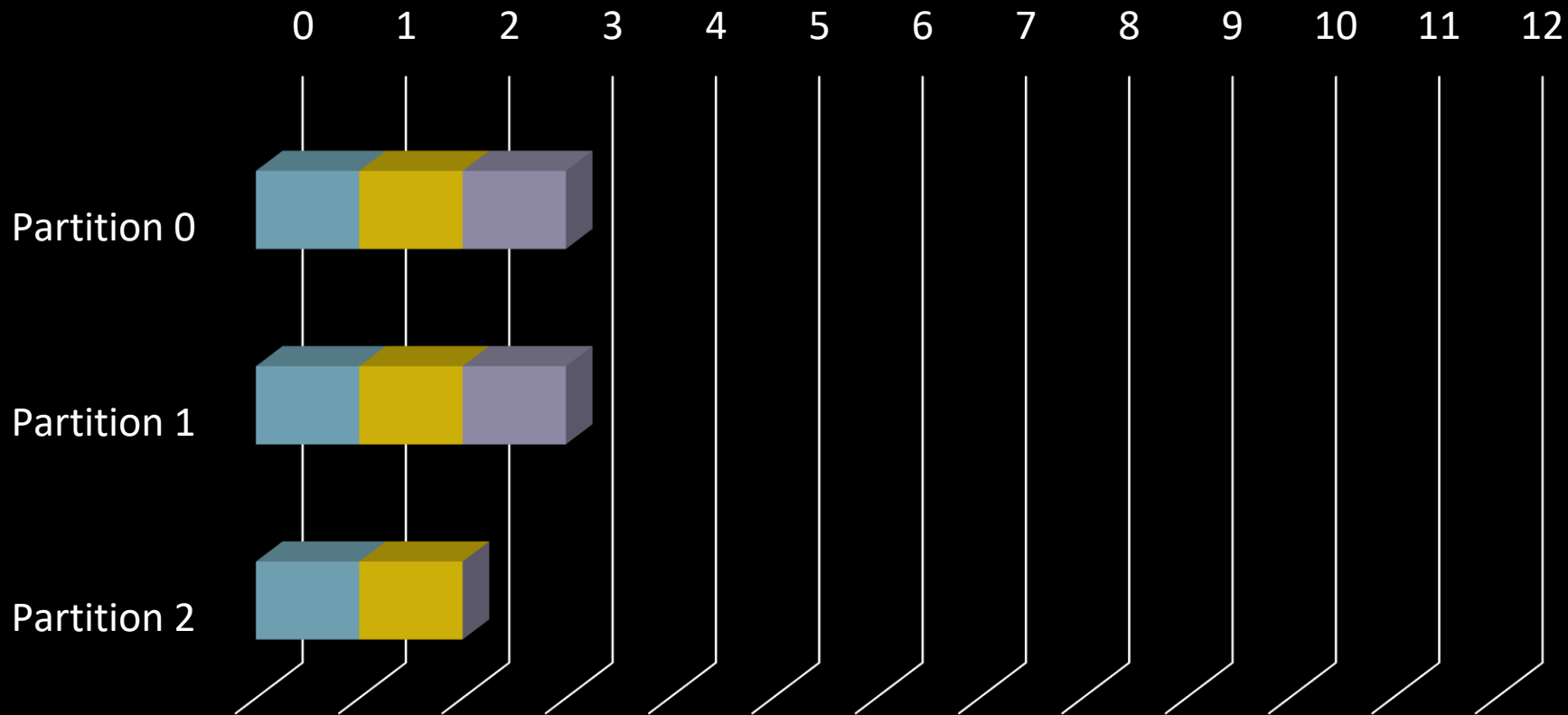
# Анатомия топика



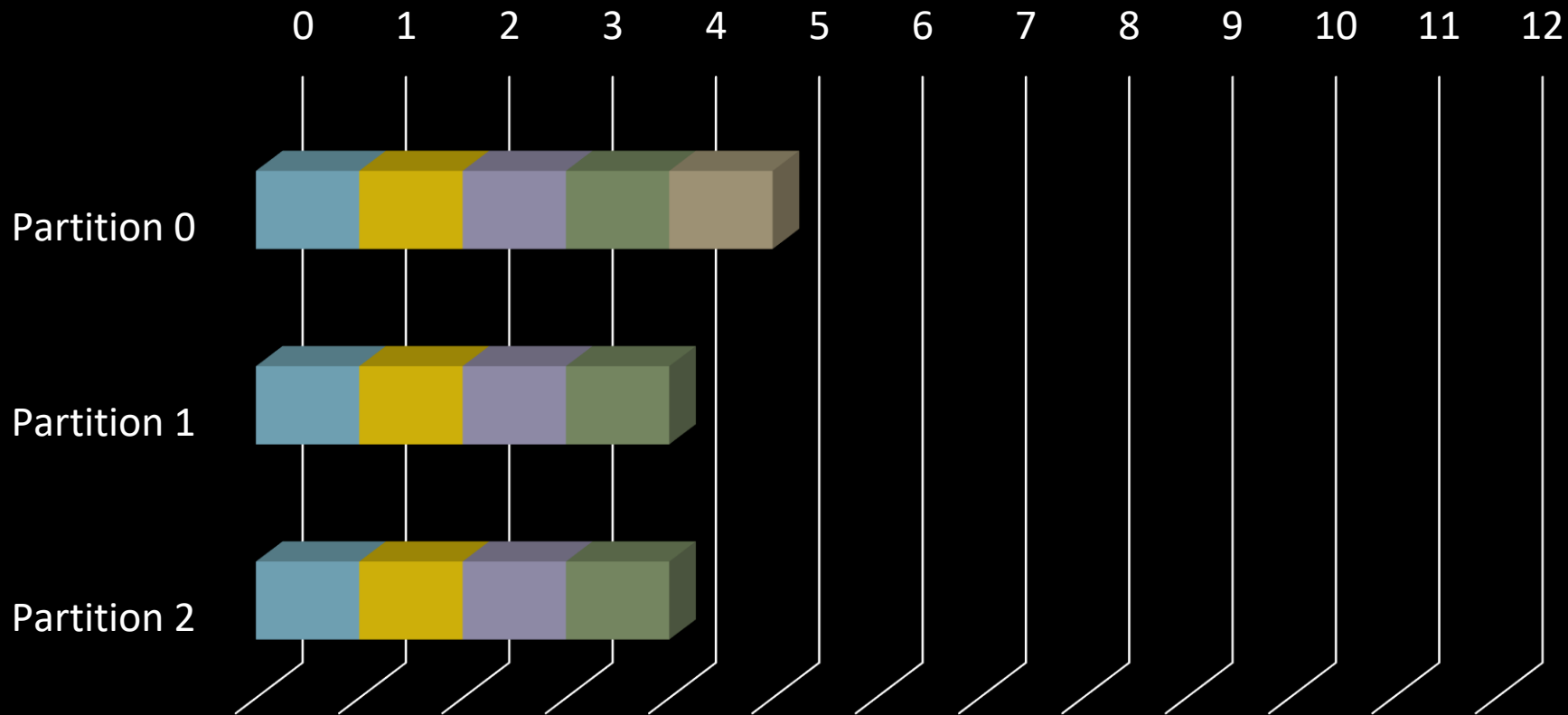
# Анатомия топика



# Анатомия топика



# Анатомия топика





# Шардирование



# Плюсы-минусы шардирования

+ увеличение  
производительности за счет  
параллелизации

+ более равномерное  
использование ресурсов  
кластера

- много сущностей – файловые  
дескрипторы, потоки и т.п.

- Затраты на обслуживание  
большого количества шард

# Тест разного количества партиций

Количество партиций	Скорость записи (MB/S)	Скорость чтения (MB/S)
1	39	86
3	86	199
5	97	201

# Что тестировать?

Производительность одного шарда(партиции)

Динамику роста производительности

Равномерность утилизации ресурсов

Предел производительности

Выполнение отложенных операций

**Про риски,  
требования и  
затраты**

# Риски

Искажение (порча) данных

Частичная или полная потеря данных

Временная недоступность сервиса на чтение и/или запись

Снижение скорости работы сервиса

# Требования

А действительно ли снижение характеристик работы системы недопустимо?

Можно ли снизить входящий поток или интенсивность чтения?

А точно данные нужно хранить продолжительное время?

# Баланс





## Второе дерево



## Второе дерево

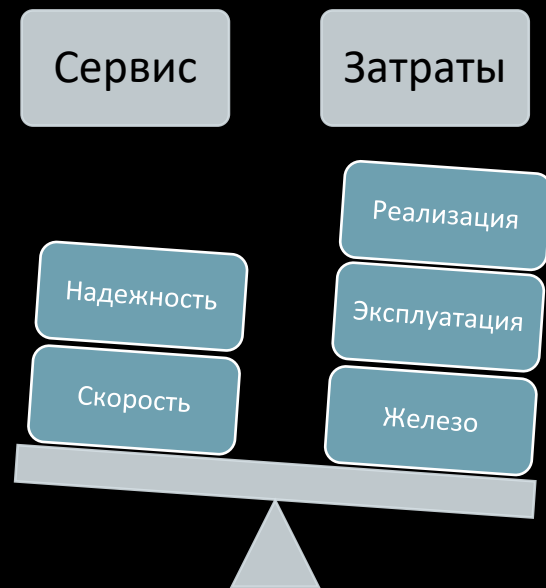


# Насколько дорого закидать ресурсами?

Закладывать железо при увеличении сложности оптимизаций или появления рисков ухудшения характеристик кластера

# Итого

- Оценка рисков и затрат, явных и неявных
- Не все оптимизации одинаково полезны
- Может вырастить еще одно дерево?



tech.kontur.ru

Сергей Махетов



**Profitfx@mail.ru**



**@Mahetovs**