

# Уменьшение фрагментации ресурсов при честном планировании на распределенных вычислительных кластерах

Тонких Андрей Александрович

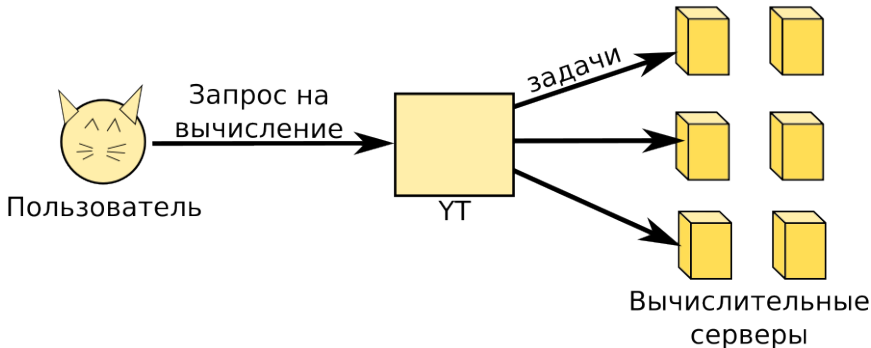
Руководитель: Игнатий Игоревич Колесниченко

НИУ ВШЭ СПб

7 июня 2019 г.

# MapReduce

Работа выполнялась в рамках MapReduce-системы YT<sup>1</sup>.



<sup>1</sup>YT: зачем Яндексу своя MapReduce-система и как она устроена — <https://habr.com/ru/company/yandex/blog/311104>

Планировщик выбирает, какие задачи на каких вычислительных серверах запускать.

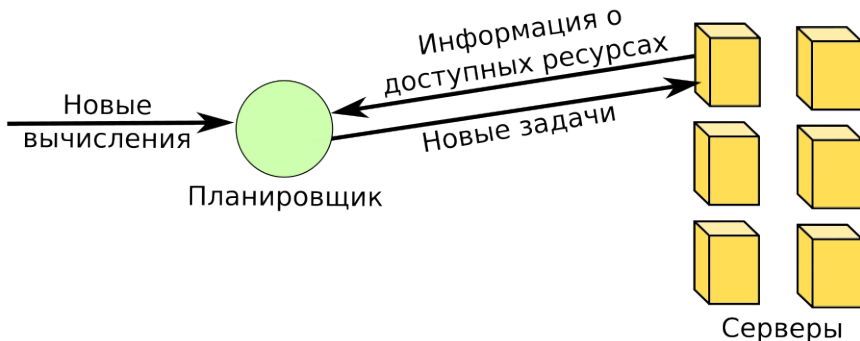


Рис.: Устройство планировщика УТ

- В основе алгоритма планирования YT лежит понятие Dominant Resource Fairness<sup>2</sup> (DRF), а точнее его обобщение для иерархий пользователей – HDRF<sup>3</sup>.
- Честный алгоритм обладает рядом важных свойств. Например, устойчивость к обману и стимул делиться.

---

<sup>2</sup>Ali Ghodsi, Matei Zaharia, Benjamin Hindman, Andy Konwinski, Scott Shenker, and Ion Stoica. Dominant Resource Fairness: Fair Allocation of Multiple Resource Types. In Nsdi, 2011.

<sup>3</sup>Arka A. Bhattacharya, David Culler, Eric Friedman, Ali Ghodsi, Scott Shenker, and Ion Stoica. Hierarchical Scheduling for Diverse Datacenter Workloads. In SoCC'13, 2013.

# Фрагментация

Фрагментация может приводить к низкой утилизации ресурсов кластера.

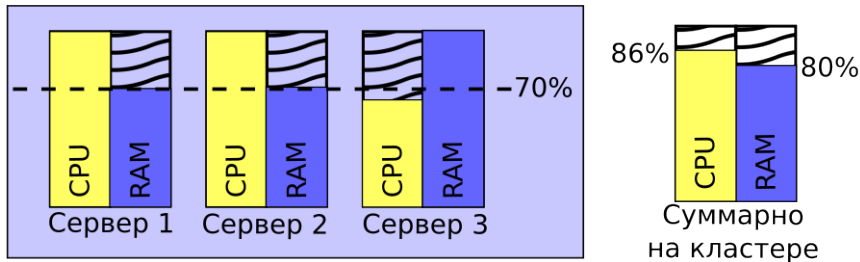


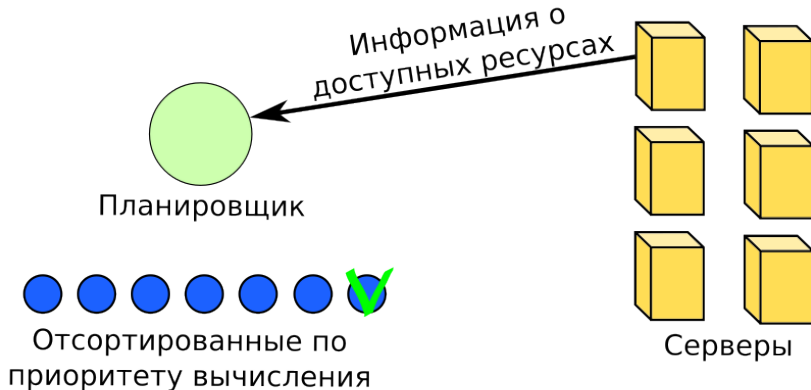
Рис.: Пример фрагментированного кластера из 3 серверов

Tetris<sup>5</sup> – планировщик, разработанный в Microsoft, одной из задач которого являлась эффективная упаковка задач.

---

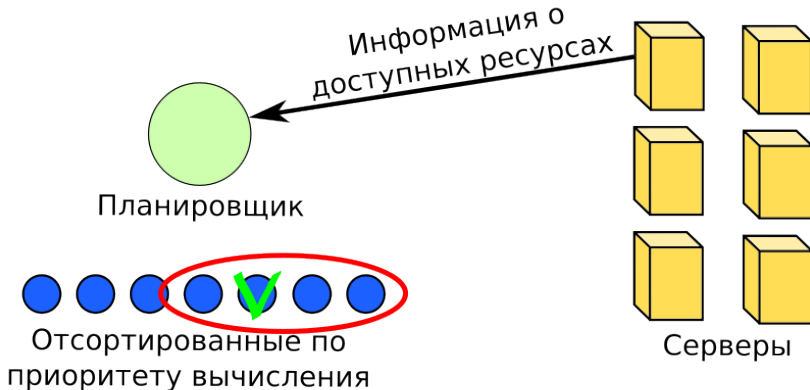
<sup>5</sup>Grandl, R., Ananthanarayanan, G., Kandula, S., Rao, S., & Akella, A. (2015). Multi-resource packing for cluster schedulers. ACM SIGCOMM Computer Communication Review, 44(4), 455-466.

Честный планировщик:



# Архитектура Tetris

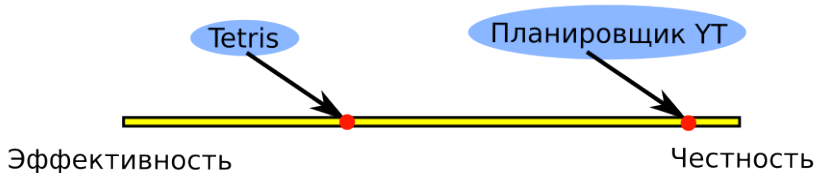
Tetris:



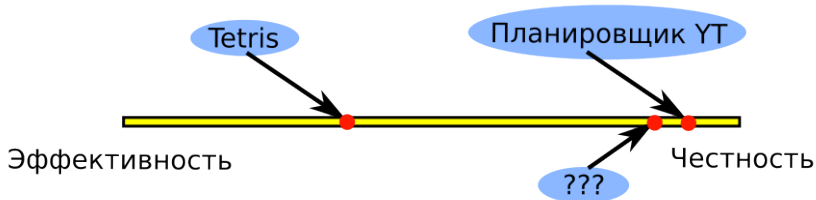


# Честность Tetris

- Авторы Tetris сознательно жертвуют большинством гарантий честного алгоритма.
- Существует trade-off между честностью и эффективностью планирования. Задачей Tetris было предоставить более эффективное и менее честное решение.



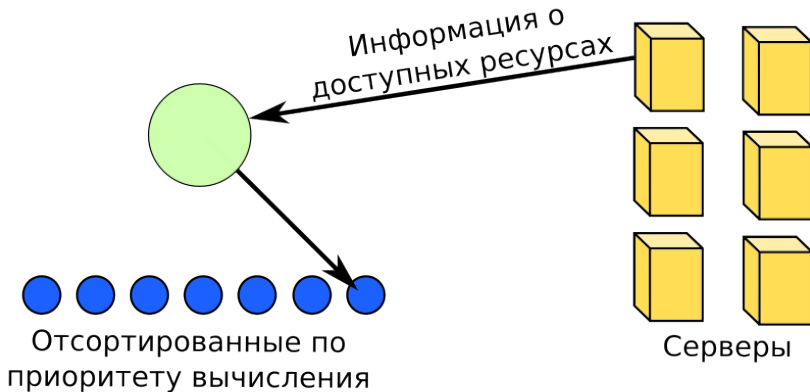
Разработать новый более гибкий подход к борьбе с фрагментацией ресурсов.



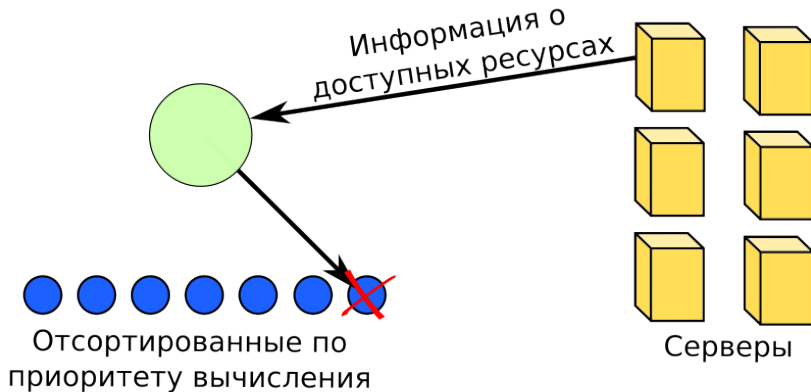
## Задачи:

- Разработать архитектуру.
- Разработать алгоритм в рамках выбранной архитектуры.
- Реализовать прототип.
- Провести эксперименты.

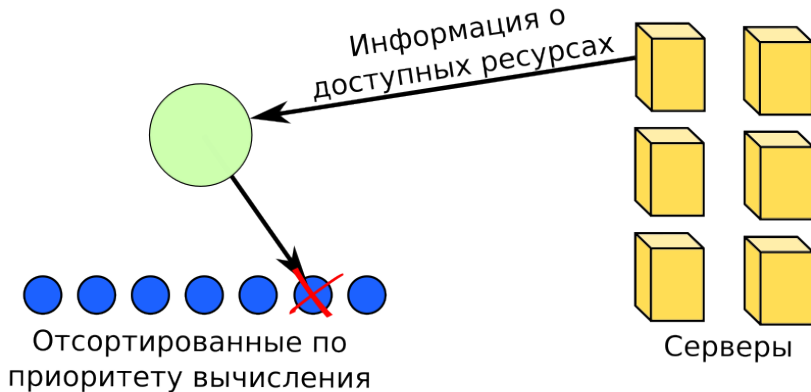
# Архитектура решения



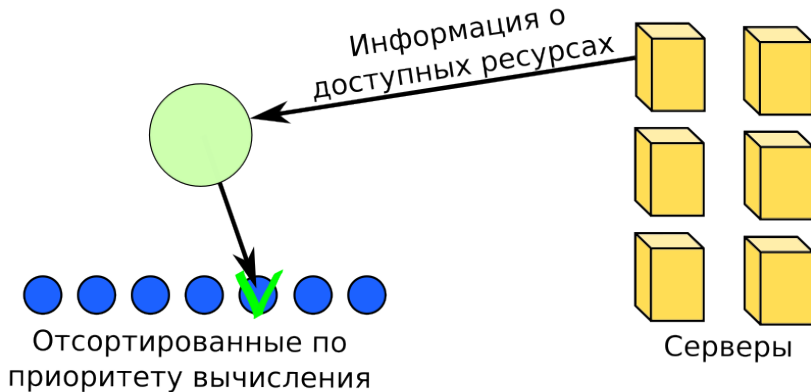
# Архитектура решения



# Архитектура решения



# Архитектура решения



Алгоритм принятия решений контроллером вычисления:

- Посмотрим на вектор предлагаемых ресурсов как на случайную величину.
- Предполагается, что распределение предлагаемых ресурсов имеет длинные промежутки стабильности.
- Контроллер смотрит на последние  $n = 15$  предложений как на выборку из случайной величины и отказывается от аллокации, если задачи данного вычисления относительно плохо упаковываются на данный сервер.
- Качество упаковки оценивается при помощи эвристической функции

$H : (server\_resources, task\_resources) \rightarrow double.$



- Каждый контроллер вычислений может иметь свою собственную стратегию принятия решений, что делает данный подход крайне гибким и хорошо настраиваемым.
- Аллокация сходится к честной при некоторых предположениях, которые выполняются на больших кластерах, разделяемых между многими пользователями.

Прототип алгоритма упаковки был реализован в планировщике YT и протестирован на кластере из 73 серверов.

	<b>Без упаковки</b>	<b>С упаковкой</b>
<b>Утилизация CPU</b>	92.2%	96.7% (+3.5%)
<b>Утилизация RAM</b>	91.5%	96.4% (+4.9%)

Результаты экспериментов.

- Был предложен новый гибкий подход для борьбы с фрагментацией при честном планировании задач.
- Анализ решения показал, что аллокация сходится к честной.
- Реализован прототип в рамках MapReduce-системы YТ.
- Эксперименты показали, что предложенный подход может значительно увеличивать утилизацию.

- Разработать и протестировать алгоритм для контроллера с лучшей или регулируемой сходимостью.
- Использовать машинное обучение для принятия решений на контроллере.