



50 часов/час: масштабирование автотестов в iOS

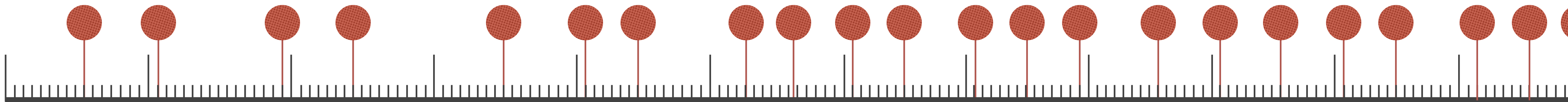
Владислав Алексеев, Авито

План

- Автотесты как возможность делать релизы часто
- Проблемы производительности при автотестировании
- Emsee — наша утилита для запуска тестов

У НАС МНОГО РЕЛИЗОВ

IOS РЕЛИЗЫ AVITO



October '17

November '17

December '17

January '18

February '18

March '18

April '18

May '18

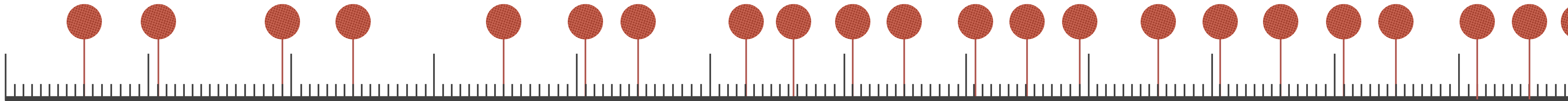
June '18

July '18

August '18

September '18

IOS РЕЛИЗЫ AVITO



October '17

November '17

December '17

January '18

February '18

March '18

April '18

May '18

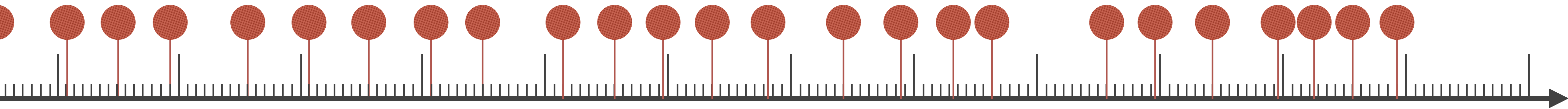
June '18

July '18

August '18

September '18

IOS РЕЛИЗЫ AVITO



May '18

June '18

July '18

August '18

September '18

October '18

November '18

December '18

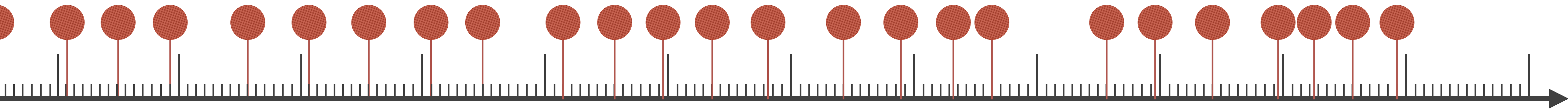
January '19

February '19

March '19

April '19

IOS РЕЛИЗЫ AVITO



May'18

June'18

July'18

August'18

September'18

October'18

November'18

December'18

January'19

February'19

March'19

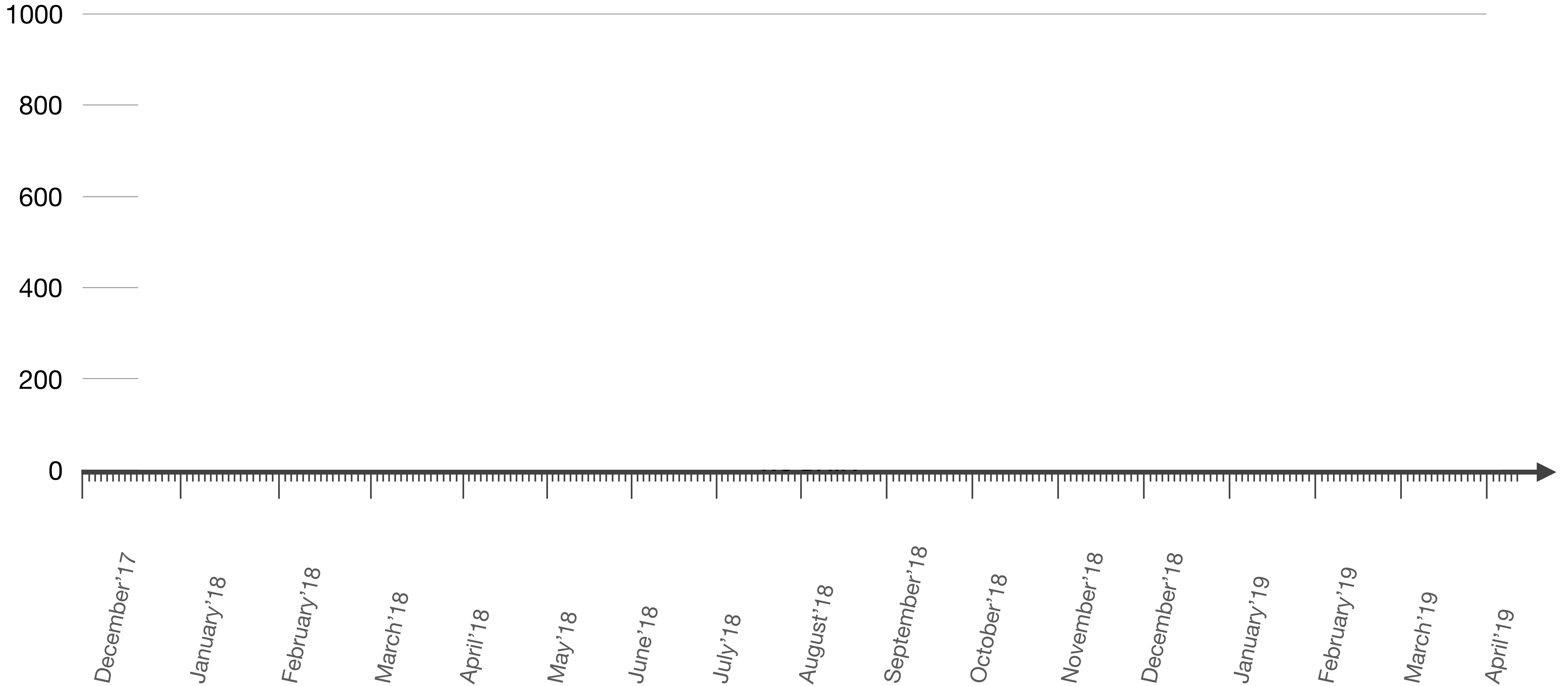
April'19

IOS РЕЛИЗЫ AVITO

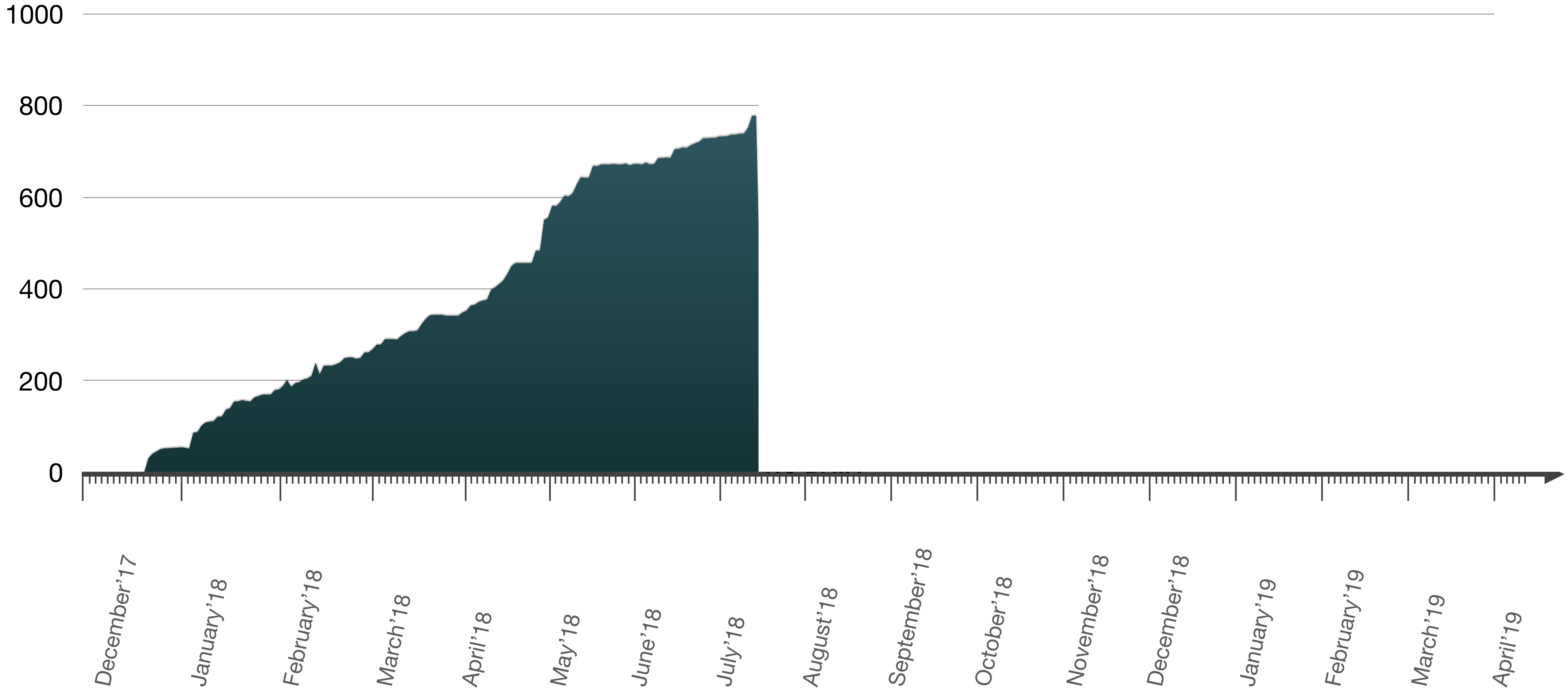


МЫ ГОНЯЕМ UI ТЕСТЫ

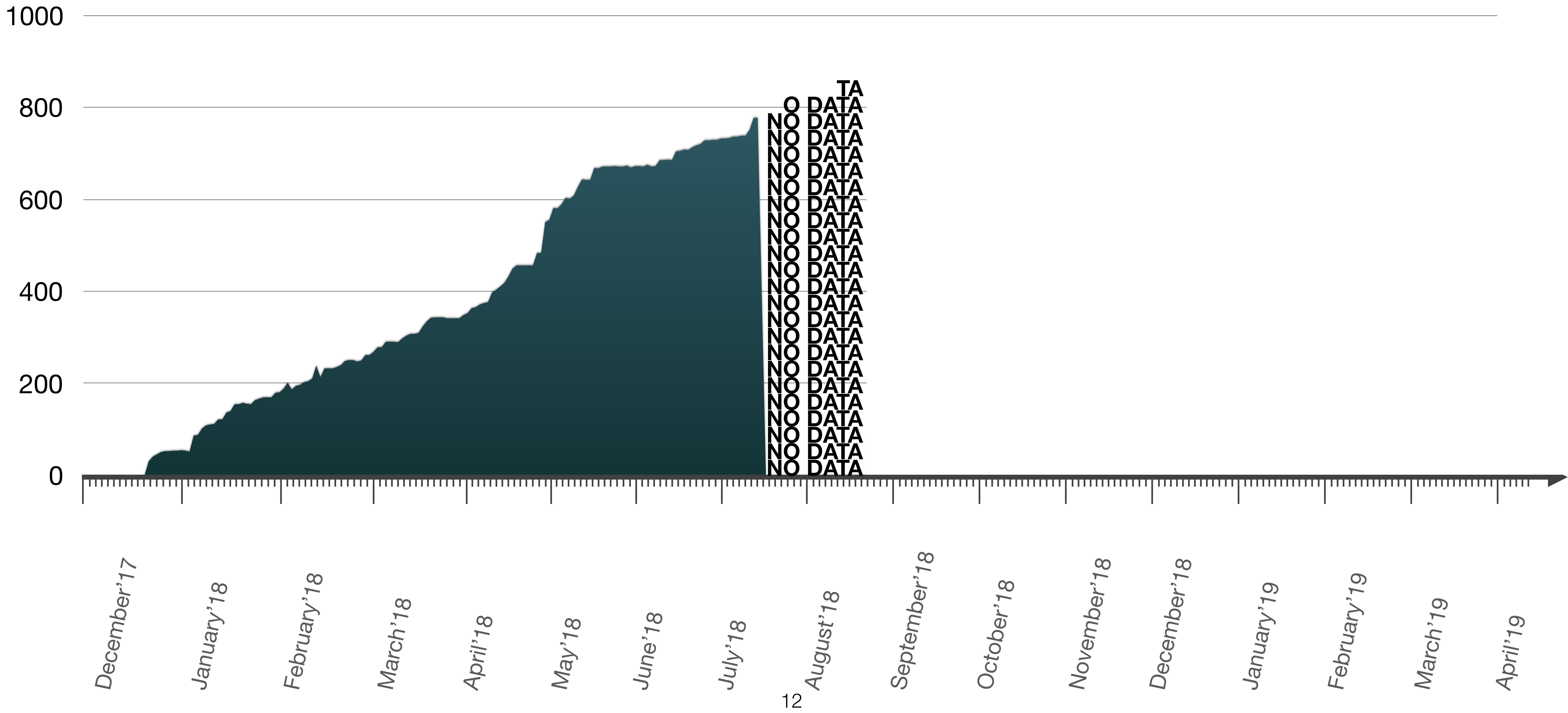
ПОСТ UI ТЕКТОВ (IOS)



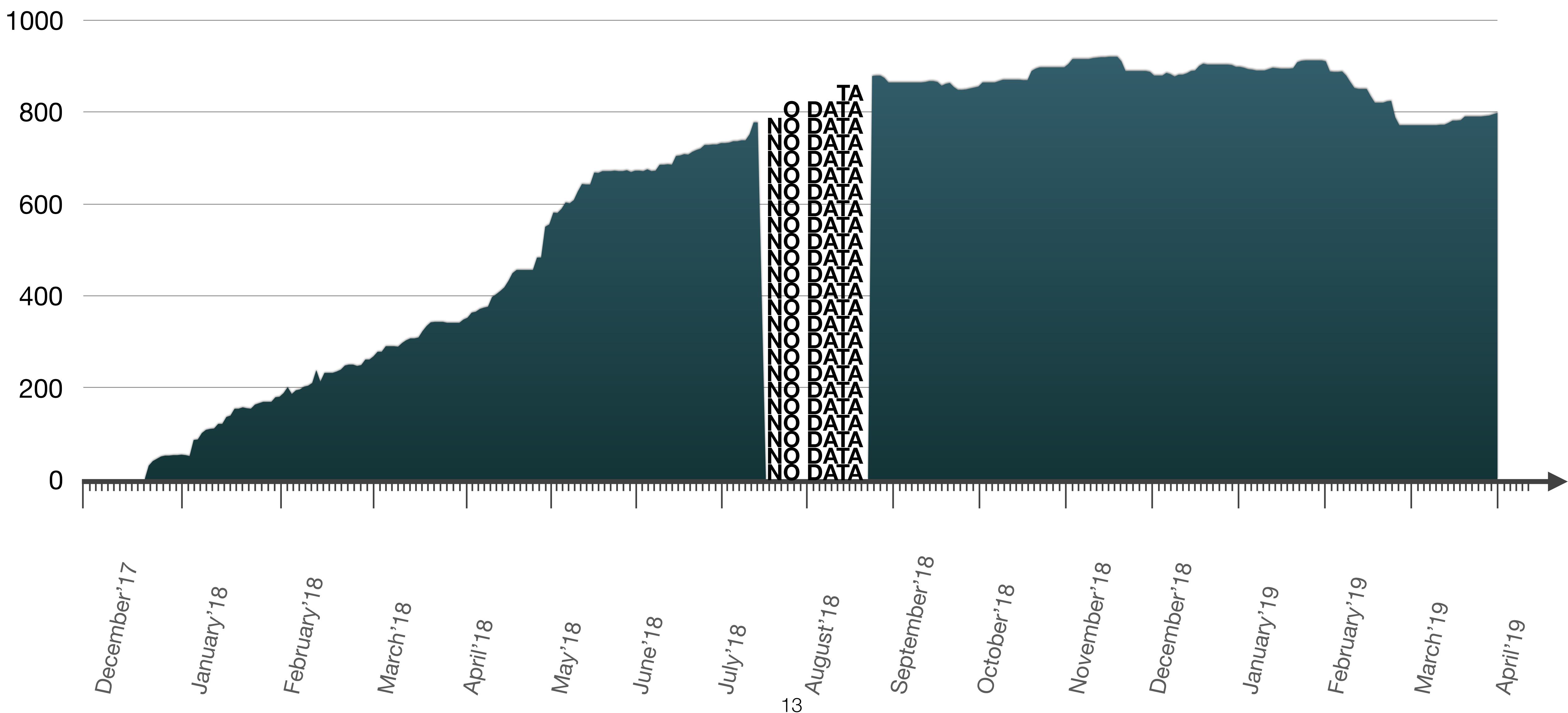
POCT UI TECTOB (IOS)



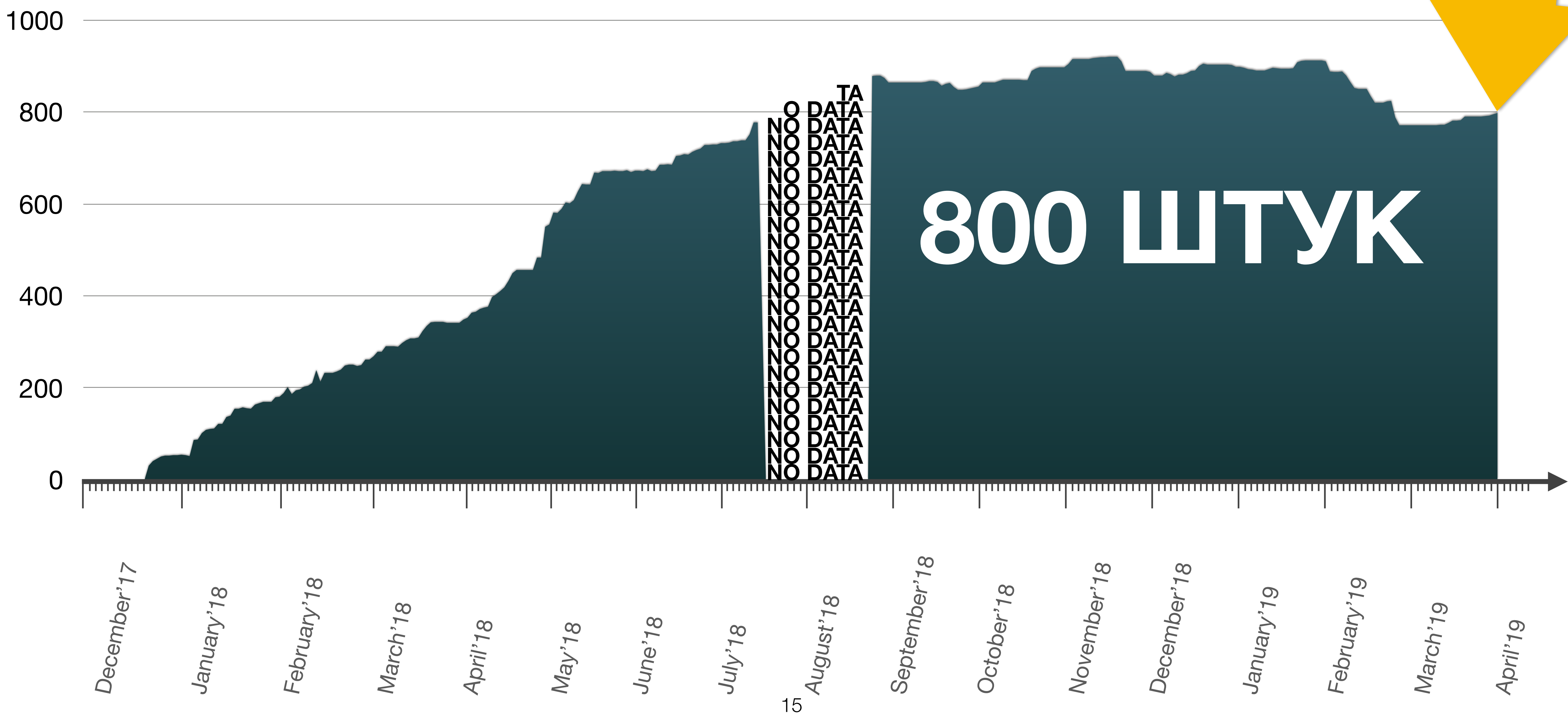
ПОСТ UI ТЕКТОБ (IOS)



ПОСТ UI ТЕКТОВ (IOS)



РОСТ UI ТЕКСТОВ (IOS)



МЫ ЗАПУСКАЕМ ТЕСТЫ

- НА ПУЛЛ РЕКВЕСТАХ
- В ТЕЧЕНИЕ ДНЯ

ВСЕ ТЕСТЫ В ТЕЧЕНИЕ ДНЯ

800 тестов

iOS 10, 11, 12

2400 тестов

НЕКОТОРЫЕ ТЕСТЫ НА ПУЛЛ РЕКВЕСТАХ

У нас много пулл реквестов

800 тестов запускать мы пока не можем

НЕКОТОРЫЕ ТЕСТЫ НА ПУЛЛ РЕКВЕСТАХ

У нас много пулл реквестов

800 тестов запускать мы пока не можем

Билд с тестами длился бы по 2 часа!

НЕКОТОРЫЕ ТЕСТЫ НА ПУЛЛ РЕКВЕСТАХ

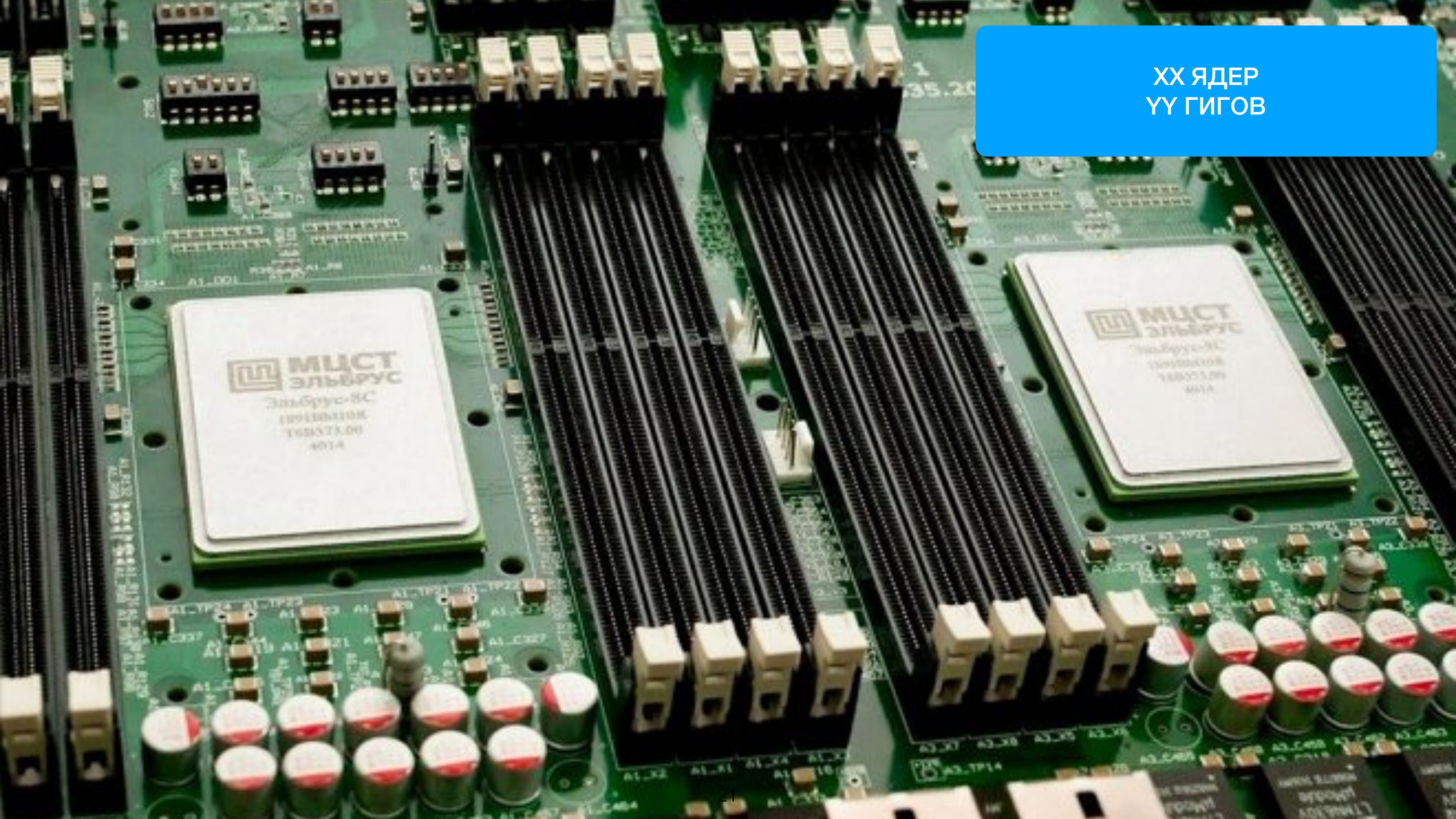
У нас много пулл реквестов

800 тестов запускать мы пока не можем

Билд с тестами длился бы по 2 часа!

Оборудование так себе

XX ЯДЕР
YY ГИГОВ





XX ЯДЕР
YY ГИГОВ



6 ЯДЕР
16 ГИГОВ

ИЗБРАННЫЕ UI ТЕСТЫ НА PR

Тесты, которым мы **доверяем**:

Заведомо **зеленые**

Тесты, КОТОРЫМ МЫ **доверяем**:

Заведомо **зеленые**

Крайне **стабильные**

Тесты, КОТОРЫМ МЫ **доверяем**:

Заведомо **зеленые**

Крайне **стабильные**

Высоко**качественные**

Тесты, которым мы **доверяем**:

- ✓ Заведомо **зеленые**
- ✓ Крайне **стабильные**
- ✓ Высоко**качественные**

Тесты, которым мы **доверяем**

МЫ ПРИДУМАЛИ **TRUSTED** ТЕСТЫ

ЗЕЛЕННЫЕ ТЕСТЫ НА ДИСТАНЦИИ

TRUSTED ТЕСТЫ

Test 1						
Test 2						
Test 3						
Test 4						
Test 5						
	Билд 771	Билд 772	Билд 773	Билд 774	Билд 775	Билд 776

TRUSTED ТЕСТЫ

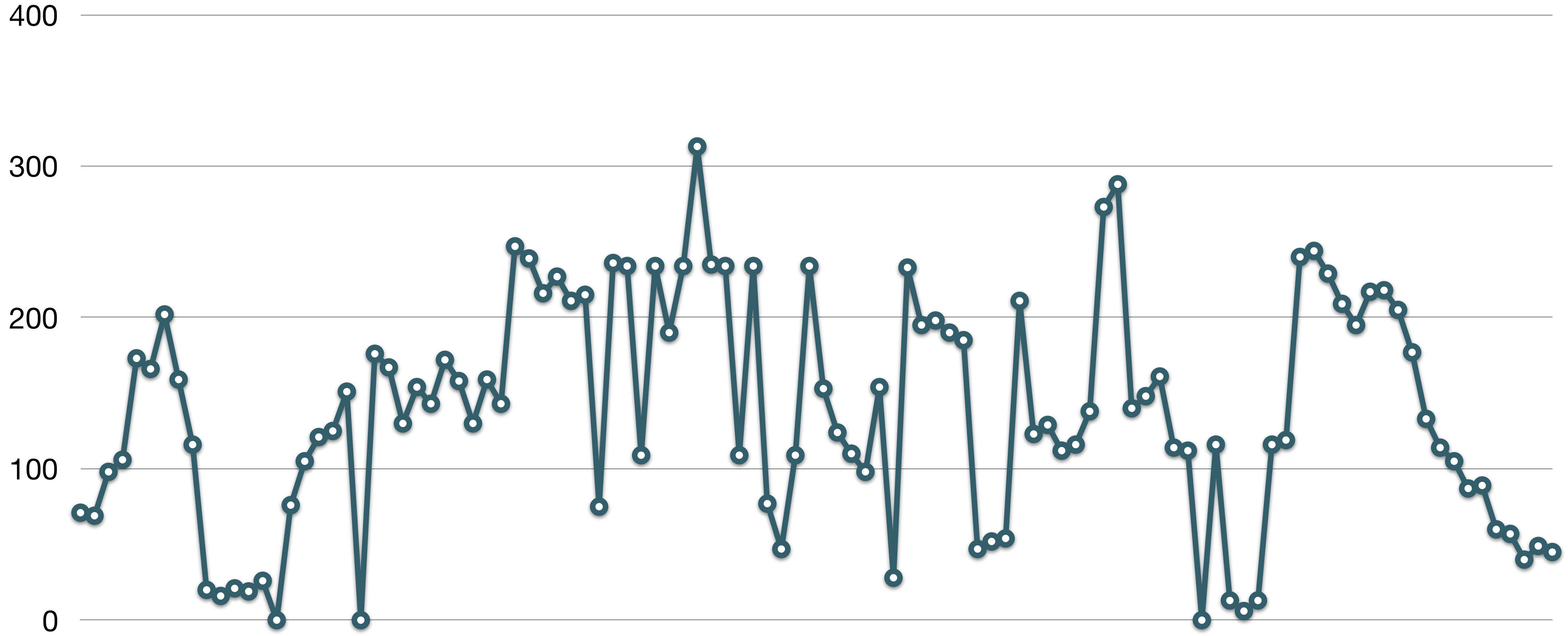
Дистанция: 4 Full Regress билда

Trusted?

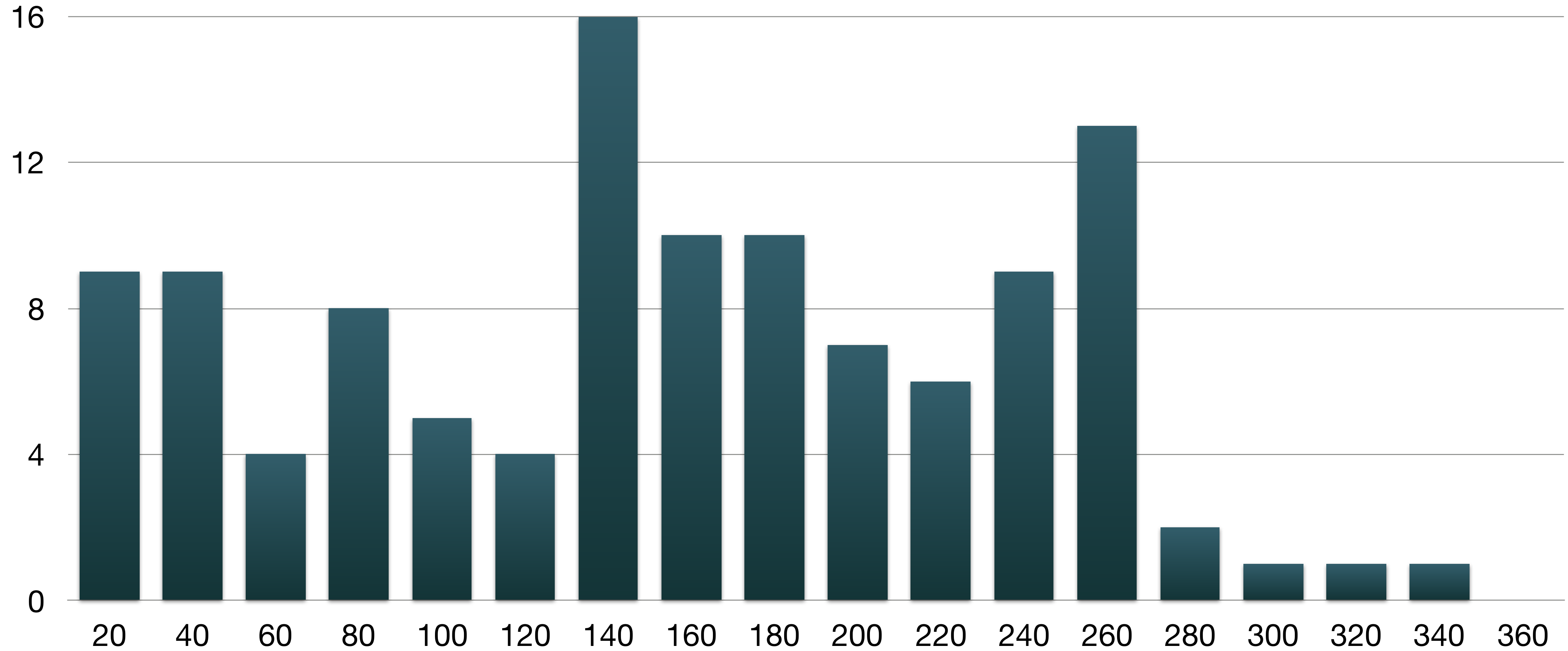
Test 1	✗	✗	✗	✗	✗	✗
Test 2	✓	✓	✓	✓	✓	✓
Test 3	✗	✓	✗	✓	✗	✗
Test 4	✓	✗	✓	✓	✓	✗
Test 5	✗	✗	✓	✓	✓	✓
	Билд 771	Билд 772	Билд 773	Билд 774	Билд 775	Билд 776

✗
✓
✗
✗
✓

TRUSTED ТЕСТЫ



TRUSTED ТЕСТЫ



TRUSTED ТЕСТЫ

200–300 trusted тестов — укладываемся в 20–30 минут



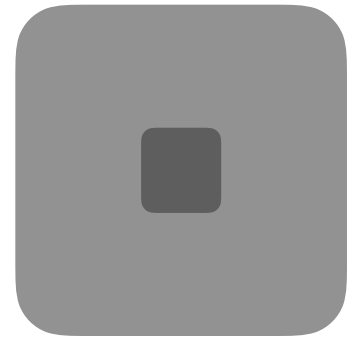
ТИПОВОЙ РАСЧЕТ ВРЕМЕНИ ПРОХОДА UI ТЕСТОВ

ТИПОВОЙ РАСЧЕТ ВРЕМЕНИ ПРОХОДА UI ТЕСТОВ

	Full Regress	На Pull Request
Средняя длительность одного теста	60 сек	60 сек
Число тестов	800 x 3 = 2400	120
Без перезапусков	39 часов	2 часа
С попытками перезапуска упавших тестов	55 часов	> 2 часов
По факту	1,5 – 2 часа	~25 минут

**Как прогонять тысячи UI тестов
в день наиболее эффективно?**

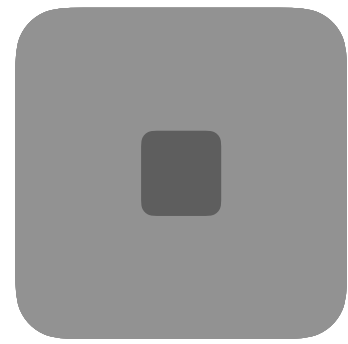
Симулятор



Симулятор

Симулятор

Симулятор



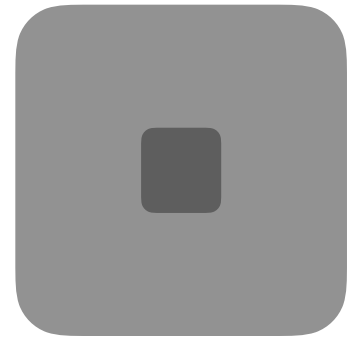
Симулятор

Симулятор

Симулятор

Масштабирование по симуляторам

Масштабирование по компьютерам



Симулятор

Симулятор

Симулятор



Симулятор

Симулятор

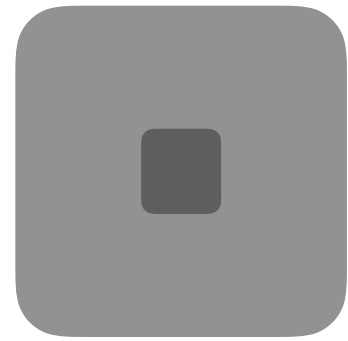
Симулятор



Симулятор

Симулятор

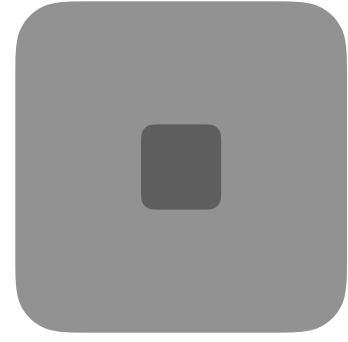
Симулятор



Симулятор

Симулятор

Симулятор



Симулятор

Симулятор

Симулятор

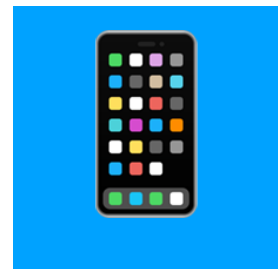
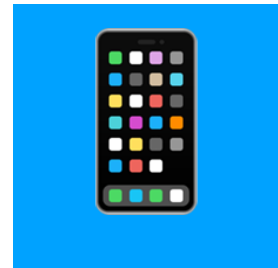
Масштабирование по симуляторам

Масштабирование по симуляторам

- Представим, что у нас есть 2 симулятора

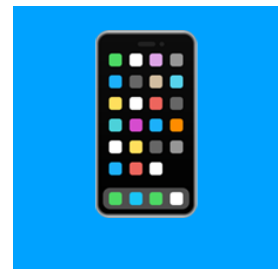
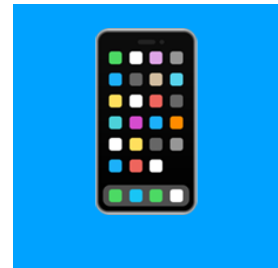
Масштабирование по симуляторам

- Представим, что у нас есть 2 симулятора



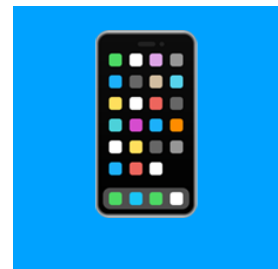
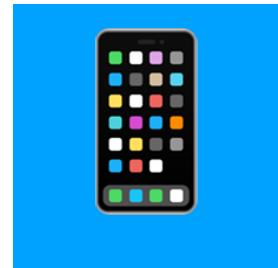
Масштабирование по симуляторам

- Представим, что у нас есть **2 симулятора** и **10 тестов**



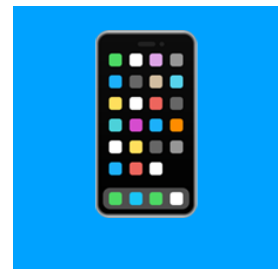
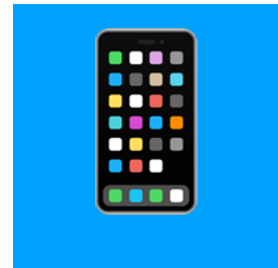
Масштабирование по симуляторам

- Представим, что у нас есть 2 симулятора и **10 тестов**



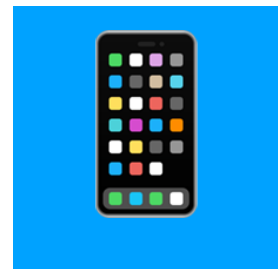
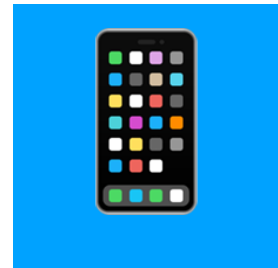
Масштабирование по симуляторам

- Представим, что у нас есть 2 симулятора и **10 тестов**



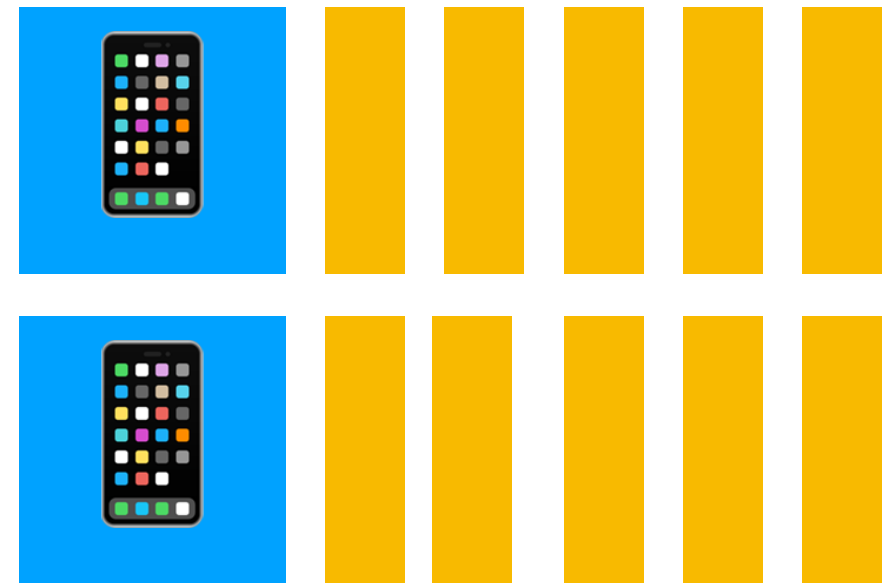
Масштабирование по симуляторам

Как раскидать по симуляторам 10 тестов?



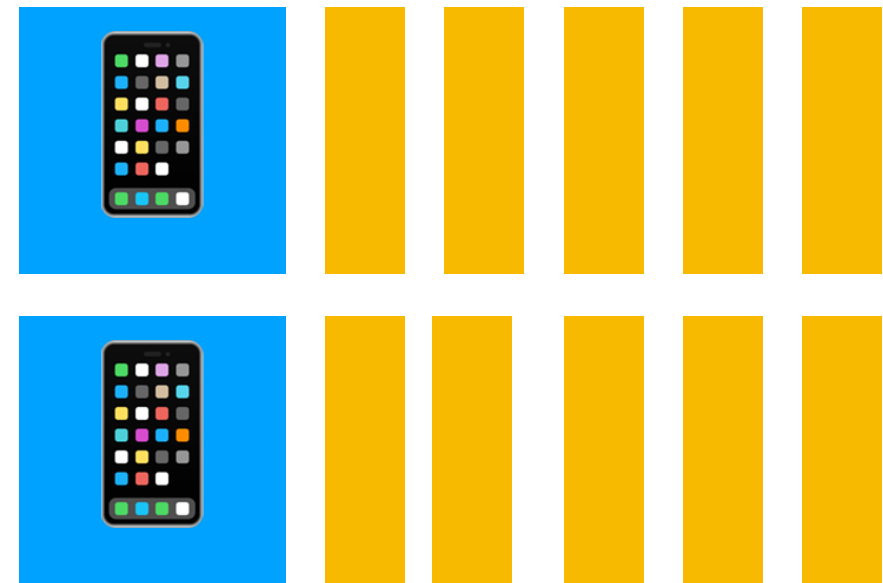
Масштабирование по симуляторам

Пополам!



Масштабирование по симуляторам

Длительность каждого теста неизвестна!



Масштабирование по симуляторам

Длительность каждого теста неизвестна!



Масштабирование по симуляторам

Длительность каждого теста неизвестна!



Масштабирование по симуляторам

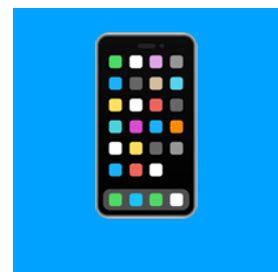
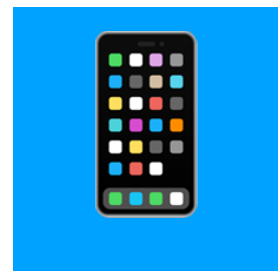
Длительность каждого теста неизвестна!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

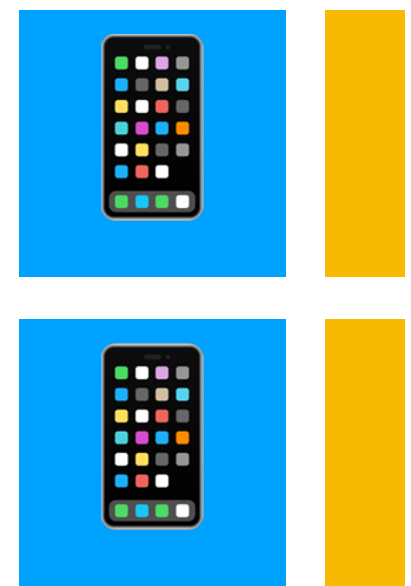
Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

Пусть забирают тесты сами!



Масштабирование по симуляторам

Раскидать по симуляторам 10 тестов **более** лучше?

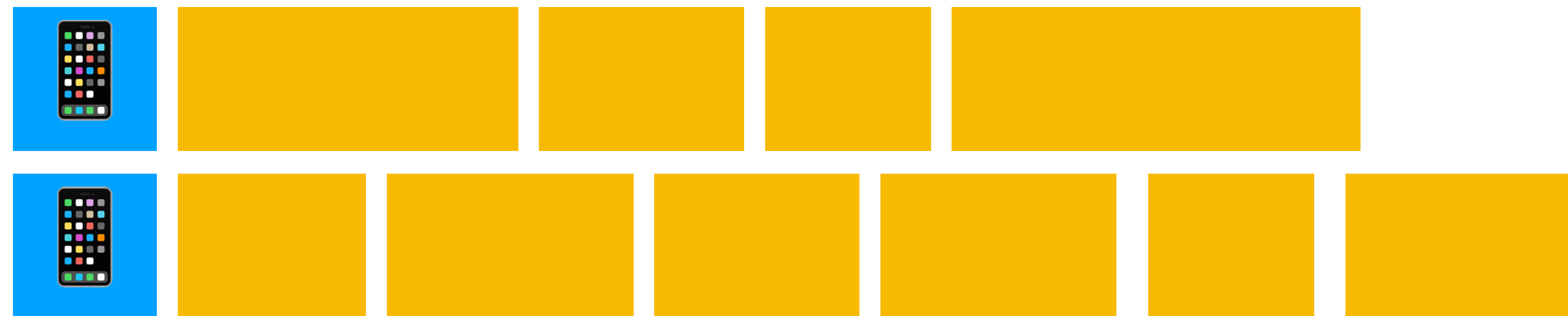
Пусть забирают тесты сами!



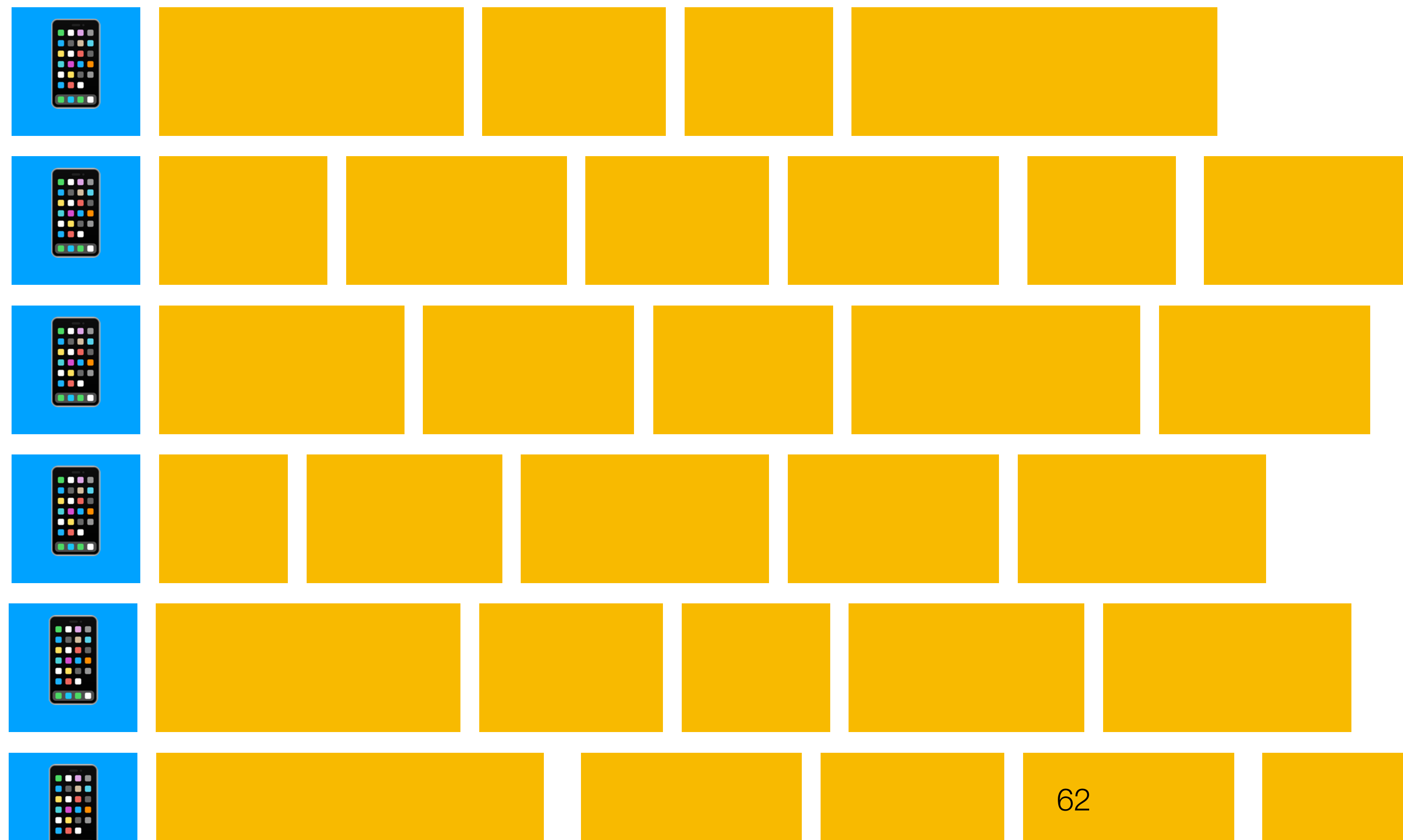
Масштабирование по компьютерам

Задача уже решена

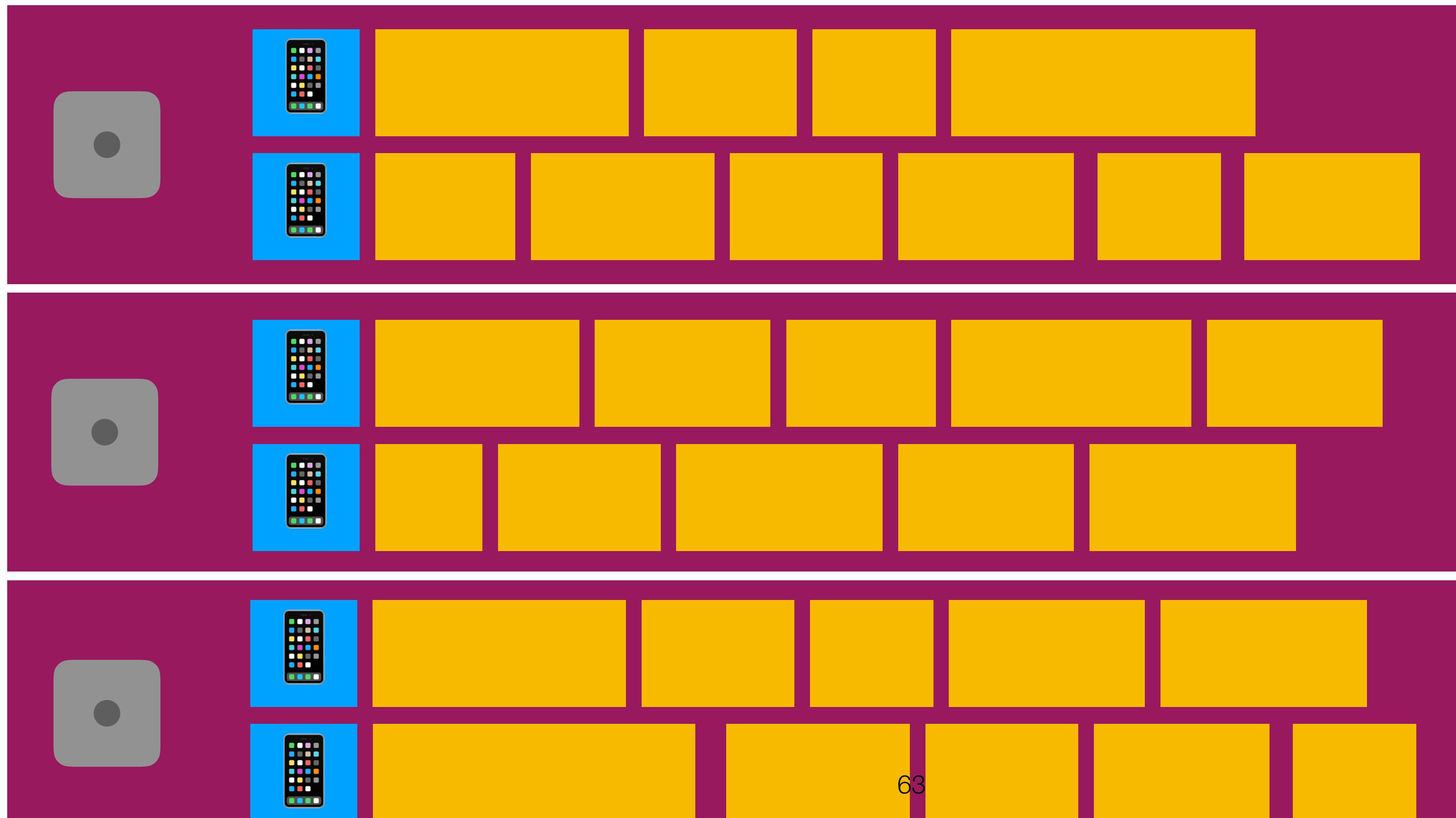
Масштабирование по компьютерам



Масштабирование по компьютерам



Масштабирование по компьютерам



УТИЛИЗАЦИЯ И КОМПЬЮТЕРОВ

B... SD B... MS S MS BIL SEL BIL SEL... RE T R... S SEL_9... SU A B M SE B... MSM S... S... BIL BIL B...

BI BX S... BIL SEL... BX M B... BIL BI BX S... BIL S... B M BIL B... MS BIL... BX_SEL MS B... BX_SEL MS B...

SE... M BIL MS M MS BX_BIL BX SEBIL B... BI BX B... BI M BIL B MS BI BX BIL SE MS BX MS BX MS MSG M...

SEL_91398_Publis... MSG... SE BIL B... BX... MSG_100622_Chan... MO SEL_91398_P... MSG... SE BIL M B... BX... MO B...

BILREC SE R... BX BX BX MSG TN BIL... BX SEL... B... SEL SU S... B... S... BX_SEL BIL... B... S... MO

MSG S... SE B BX B... MS SEL S MS SEL SE B BX B... MS SEL SEL S... RE M... SEL... B... M B... MO MO...

BIL BX... B... SD_ S... MS MMS BX S... B B... BX BIL SEL MS MS S SD_SEL MO MS MS S SD_SEL MO SD BI BIL

SEL... BIL SEL_91398_Publis... BIL SE BIL B... B... SEL_1... S... BI SEL_91398_P... BIL SEL... BIL B... B... SEL_1...

BI SEL_10... BIL SEL BX BIL... BI SEL... BI SE BX BIL BIL MS S... BI BX_ BIL MS S... BI BX_ M MON SUP MS SD_

SEL_10... B... MS MSG REC M SEL... BIL MS MSG REC M SE S... S S M MS B BIL BX MS BX BIL BX MS S...

MS BI SD_ BIL BI BIL BX B... BX... MS S... BI BI BI B BIL B... MS MSG Mai... MSBI BIL... MSG MSG Ma M BI BIL...

BX BIL... SD BIL... SE SE B MSG... B B... S B... S S B MSG... SEL SEL_9138... SE SEL... S SE S SE M... S M...

Wa TNMS B... B... BX_ BX_ TN SEL W T MS B... B... BX_ BX_ T S... M MSBX BIL SEL_9138... REC B... MS MS

B... BX_ SEL... B... MO BIL BX M B... BX S... BIL M BI B SEL... RE R S BIL B... S... BX... MS B... S... BX... M...

Воркер

Воркер

Воркер

Воркер

Воркер

Воркер

Воркер



EMCEE

A family of three—a woman, a man, and a young boy—are standing in the wooden frame of a house under construction. The woman is on the left, wearing a dark blue t-shirt and jeans, with her arms crossed. The man is in the center, wearing a light blue polo shirt and jeans, with his arm around the woman. The boy is on the right, wearing a blue and white plaid shirt and dark shorts, leaning against a wooden post. The background shows the skeletal structure of the house with vertical studs and horizontal beams under a cloudy sky.

EMCEE

ОЧЕРЕДИ

UI ТЕСТЫ

ДЖОБЫ

UNIT ТЕСТЫ

СИМУЛЯТОРЫ

ПРИОРИТЕТЫ

КОМПЬЮТЕРЫ

ДЕВАЙСЫ

БАЛАНСИРОВКА

ПЛАГИНЫ

EMCEE – ЗАПУСКАТОР IOS ТЕСТОВ

UI ТЕСТЫ UNIT ТЕСТЫ

СИМУЛЯТОРЫ КОМПЬЮТЕРЫ

ДЖОБЫ ОЧЕРЕДИ

ПРИОРИТЕТЫ

БАЛАНСИРОВКА

ПЛАГИНЫ

ДЕВАЙСЫ

EMCEE – ЗАПУСКАТОР IOS ТЕСТОВ

- ГОНЯЕТ UI ТЕСТЫ И UNIT ТЕСТЫ
- ОБСЛУЖИВАЕТ СИМУЛЯТОРЫ И КОМПЬЮТЕРЫ
- ЕДИНИЦЫ РАБОТЫ – ДЖОБЫ (ОЧЕРЕДИ ТЕСТОВ)
- ПОДДЕРЖИВАЕТ ПРИОРИТЕТЫ
- БАЛАНСИРОВКА ЗАГРУЗКИ ВОРКЕРОВ
- ПОДДЕРЖИВАЕТ ПЛАГИНЫ
- РЕАЛЬНЫЕ ДЕВАЙСЫ – TBD

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО



Инфраструктура UI-тестов в Авито
youtu.be/aBcJI80bSeU?t=427

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

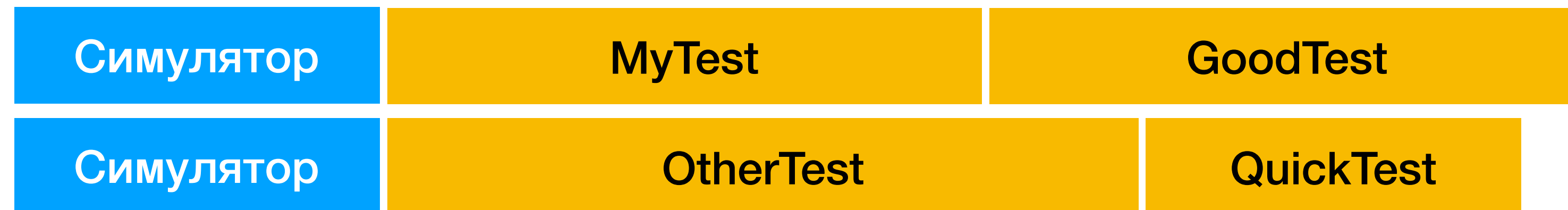
```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

Schedule Strategy

- Individual
- Equally_divided
- Progressive

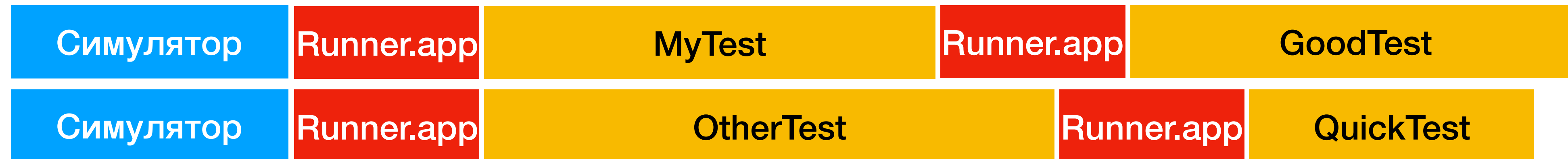
Schedule Strategy

- Individual



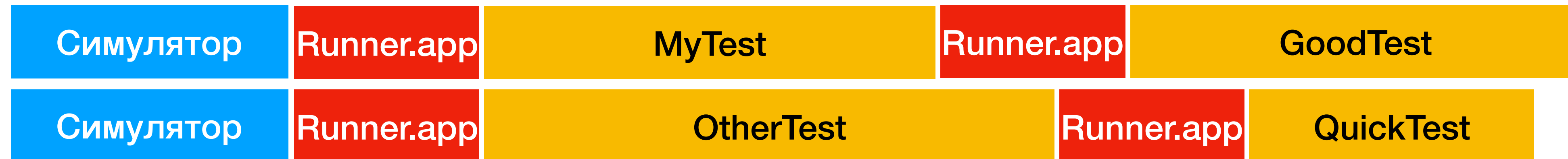
Schedule Strategy

- Individual



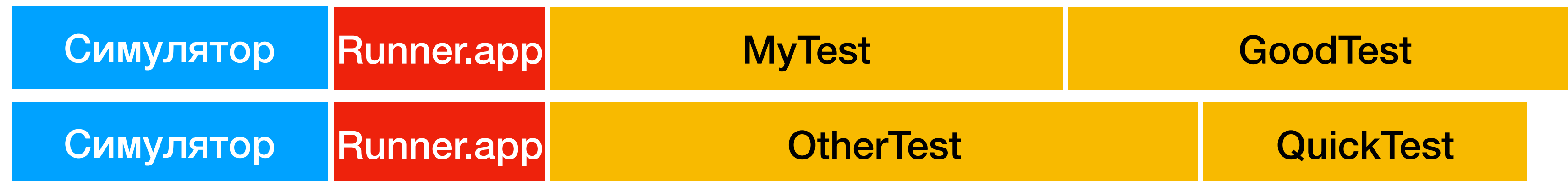
Schedule Strategy

- Equally_divided



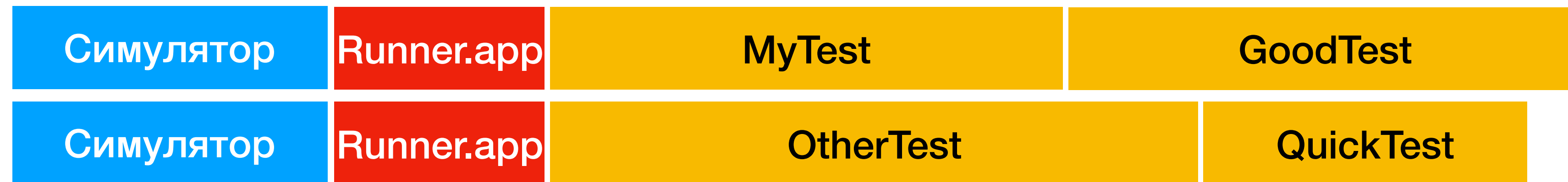
Schedule Strategy

- Equally_divided



Schedule Strategy

- Progressive



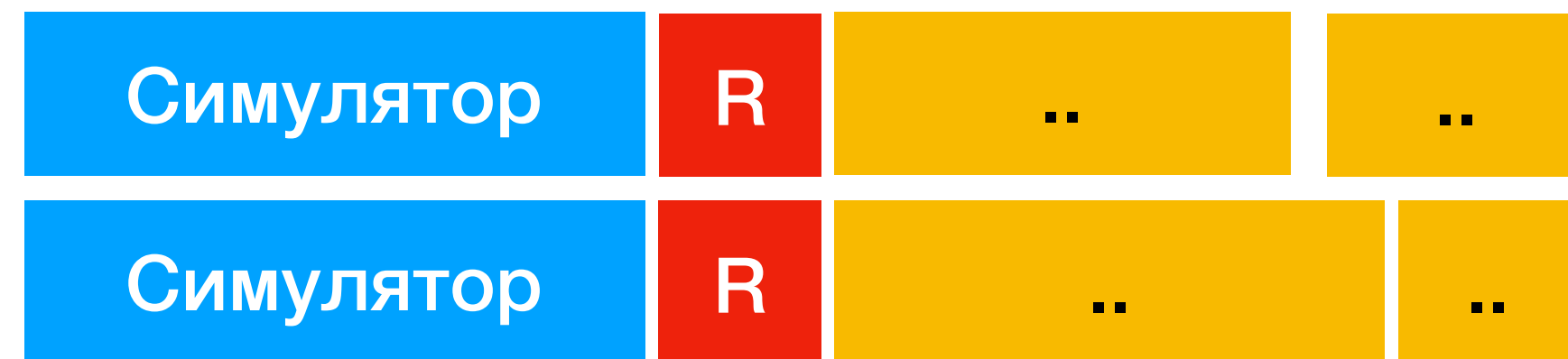
Schedule Strategy

- Progressive



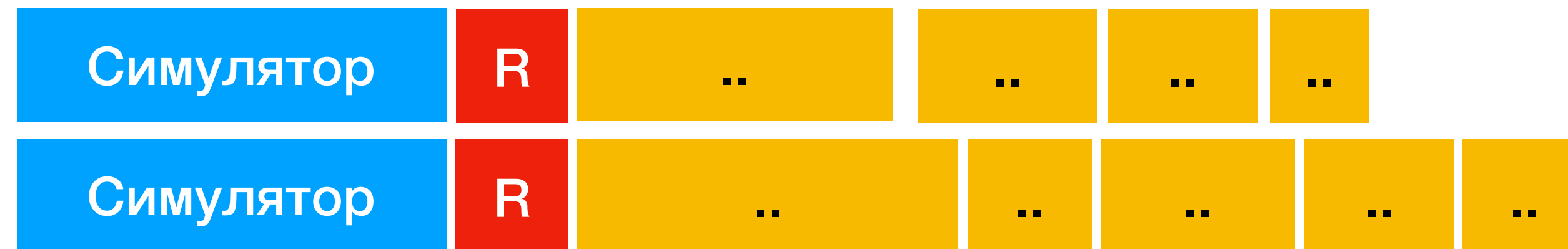
Schedule Strategy

- Progressive



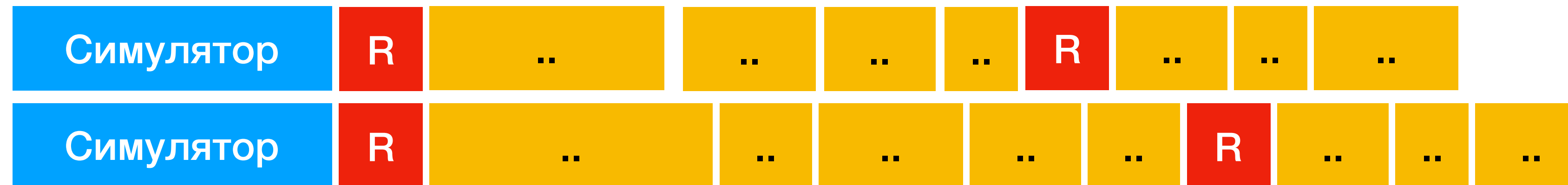
Schedule Strategy

- Progressive



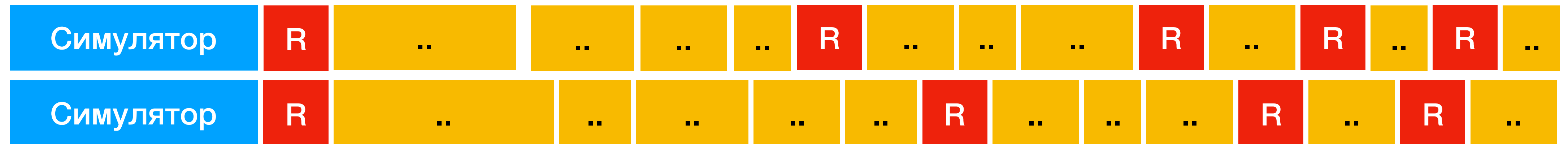
Schedule Strategy

- Progressive



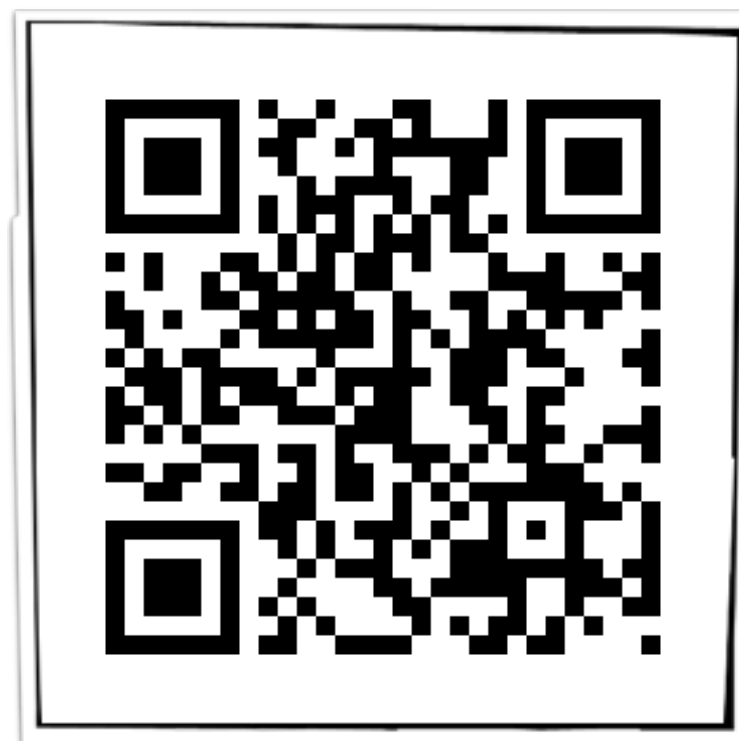
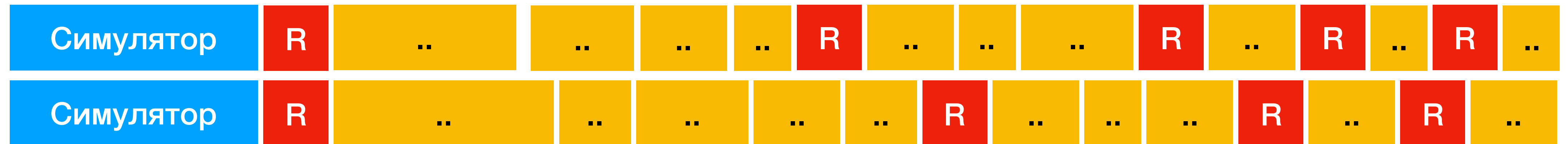
Schedule Strategy

- Progressive



Schedule Strategy

- Progressive



Инфраструктура UI-тестов в Авито
youtu.be/aBcJI80bSeU?t=427

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```


ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

Test Destination - На чем запускать тесты

```
[{
  "testDestination": {
    "deviceType": "iPhone SE",
    "runtime": "10.3"
  },
  "reportOutput": {
    "junit": "reports/iphone_se_ios_103.xml"
  }
}]
```

Test Destination - На чем запускать тесты

Xcode -> Preferences -> Components -> Simulators

```
[{  
  "testDestination": {  
    "deviceType": "iPhone SE",  
    "runtime": "10.3"  
  },  
  "reportOutput": {  
    "junit": "reports/iphone_se_ios_103.xml"  
  }  
}]
```

Test Destination - На чем запускать тесты

```
[{  
  "testDestination": {  
    "deviceType": "iPhone SE",  
    "runtime": "10.3"  
  },  
  "reportOutput": {  
    "junit": "reports/iphone_se_ios_103.xml"  
  }  
}]
```

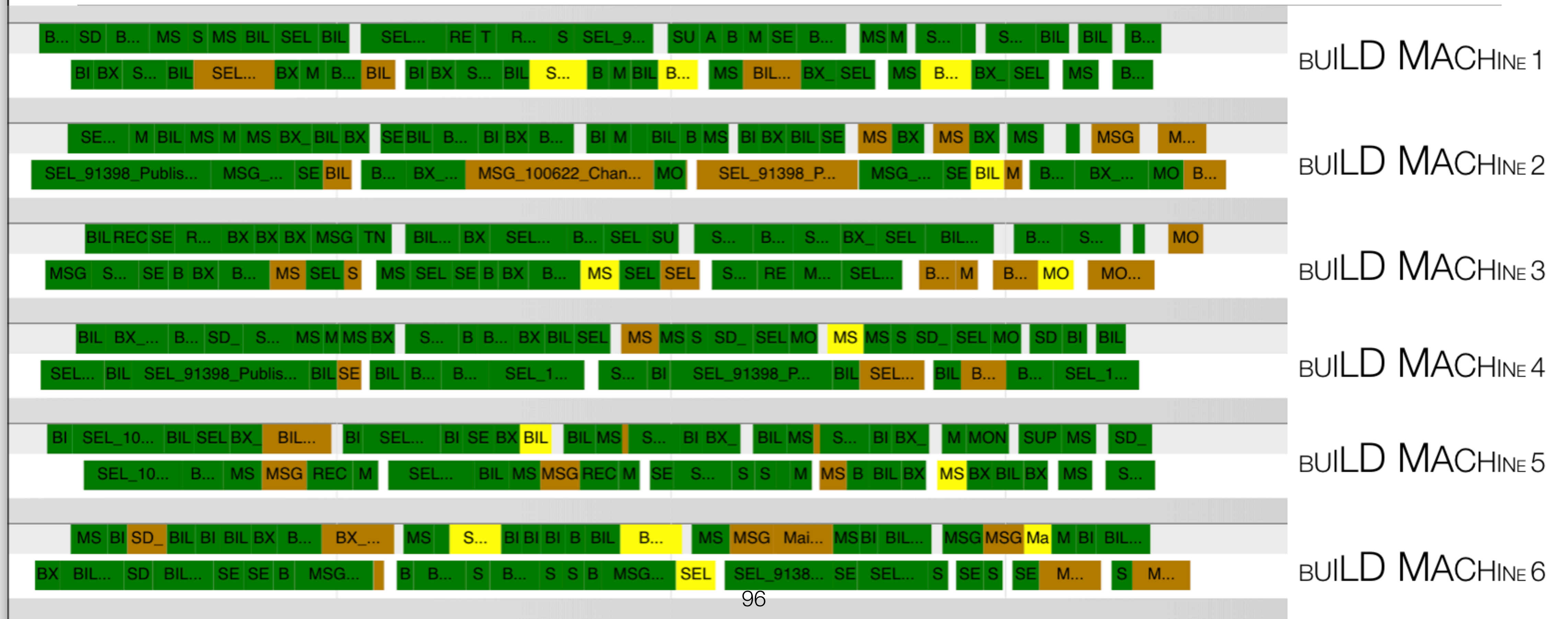
Test Destination - На чем запускать тесты

```
[{
  "testDestination": {
    "deviceType": "iPhone SE",
    "runtime": "10.3"
  },
  "reportOutput": {
    "junit": "reports/iphone_se_ios_103.xml",
    "trace": "reports/iphone_se_ios-103_trace.json"
  }
}]
```

Test Destination - На чем запускать тесты

Google Chrome: <chrome://tracing>

УТИЛИЗАЦИЯ N КОМПЬЮТЕРОВ



ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json"
```

ЗАПУСК ТЕСТОВ ЛОКАЛЬНО

```
AvitoRunner runTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--test-arg-file "tests.json"
```

--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```

--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```

--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```

--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```


--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```

--test-arg-file: Файл с тестовым планом

```
{
  "entries": [
    {
      "testToRun": "TestClass/testMethod",
      "testDestination": {"deviceType": "iPhone X", "runtime": "11.0"},
      "numberOfRetries": 2,
      "environment": {
        "TEST_SPECIFIC_ENVS": "if needed"
      }
    },
    {
      "testToRun": "AnotherTestClass/testSomethingImportant",
      "testDestination": {"deviceType": "iPhone SE", "runtime": "12.0"},
      "numberOfRetries": 0,
      "environment": {}
    }
  ]
}
```

Вытаскивание списка тестов

Список тестов в xctest банлде

```
AvitoRunner dump \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--output "dump.json"
```

Список тестов в xctest банлде

```
AvitoRunner dump \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--output "dump.json"
```

Список тестов в xctest банлде

```
AvitoRunner dump \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--output "dump.json"
```


Список тестов в xctest банлде

```
AvitoRunner dump \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--output "dump.json"
```

Список тестов в xctest банлде

```
[
  {
    "path" : ".../UITests/MONP/MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions"
  },
  {
    "path" : ".../UITests/Advertisement/BX_AdvertisementLogging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "BX_AdvertisementLogging"
  }
]
```

Список тестов в xcstest банлде

```
[
  {
    "path" : ".../UITests/MONP/MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions"
  },
  {
    "path" : ".../UITests/Advertisement/BX_AdvertisementLogging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "BX_AdvertisementLogging"
  }
]
```

Список тестов в xcctest банлде

```
[
  {
    "path" : ".../UITests/MONP/MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions"
  },
  {
    "path" : ".../UITests/Advertisement/BX_AdvertisementLogging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "BX_AdvertisementLogging"
  }
]
```

Список тестов в xctest банлде

Готовая реализация:

<https://github.com/avito-tech/Emcee/tree/master/Sources/RuntimeDump>

<https://github.com/avito-tech/Emcee/tree/master/TestApp/RuntimeDump>

ЗАПУСК ТЕСТОВ РАСПРЕДЕЛЕННО ПО КОМПЬЮТЕРАМ

Запуск тестов распределенно

```
AvitoRunner distRunTests \  
--number-of-simulators 2 \  
--app "MyApp.app" \  
--runner "MyAppUITests-Runner.app" \  
--xctest-bundle "MyAppUITests-Runner.app/PlugIns/MyAppUITests.xctest" \  
--schedule-strategy "individual" \  
--single-test-timeout 100 \  
--temp-folder "$(pwd)/tempfolder" \  
--test-destinations "destination_iphone_se_ios103.json" \  
--test-arg-file "tests.json"
```

Запуск тестов распределенно

```
AvitoRunner distRunTests \  
...
```

Запуск тестов распределенно

```
AvitoRunner distRunTests \  
... \  
--run-id "$(uuidgen)" \  
--destinations "remote_destinations.json"
```

Запуск тестов распределенно

```
AvitoRunner distRunTests \  
  ... \  
  --run-id "$(uuidgen)" \  
  --destinations "remote_destinations.json"
```

Запуск тестов распределенно

```
AvitoRunner distRunTests \  
  ... \  
  --run-id "$(uuidgen)" \  
  --destinations "remote_destinations.json"
```

--destinations: СПИСОК КОМПЬЮТЕРОВ

```
[  
  {  
    "host": "build-agent-macmini-01",  
    "port": 22,  
    "username": "remote_worker",  
    "password": "awesomeword",  
    "remote_deployment_path": "/Users/emcee/remote_ui_tests"  
  },  
  ...  
]
```

--destinations: СПИСОК КОМПЬЮТЕРОВ

```
[  
  {  
    "host": "build-agent-macmini-01",  
    "port": 22,  
    "username": "remote_worker",  
    "password": "awesomeword",  
    "remote_deployment_path": "/Users/emcee/remote_ui_tests"  
  },  
  ...  
]
```

SSH

--destinations: СПИСОК КОМПЬЮТЕРОВ

```
[  
  {  
    "host": "build-agent-macmini-01",  
    "port": 22,  
    "username": "remote_worker",  
    "password": "awesomeword",  
    "remote_deployment_path": "/Users/emcee/remote_ui_tests"  
  },  
  ...  
]
```

Запуск тестов распределенно

КАК ЭТО РАБОТАЕТ?!



Запуск тестов распределенно

КАК ЭТО РАБОТАЕТ?!



Запуск тестов распределенно

КАК ЭТО РАБОТАЕТ?!



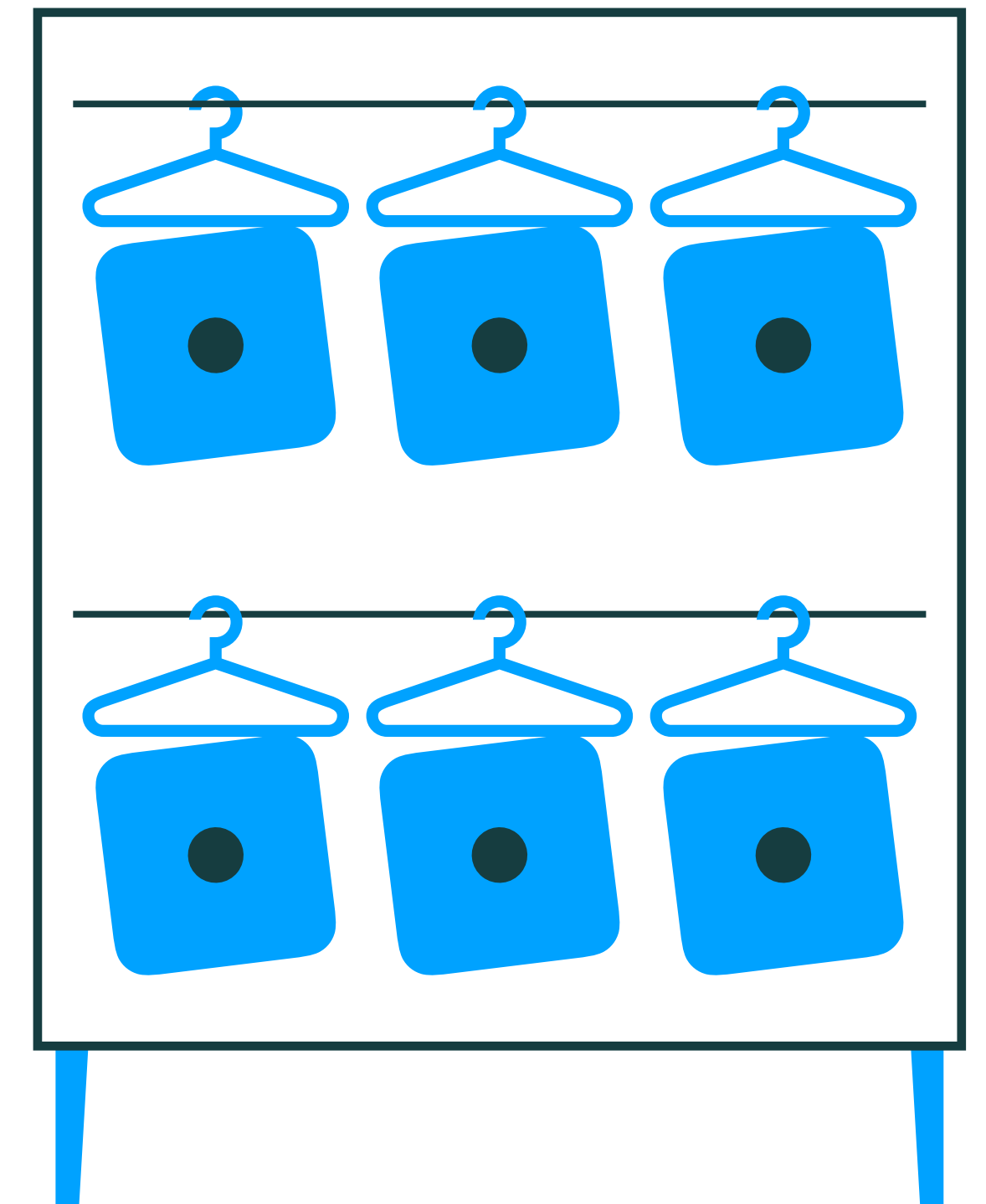
Запуск тестов распределенно

КАК ЭТО РАБОТАЕТ?!

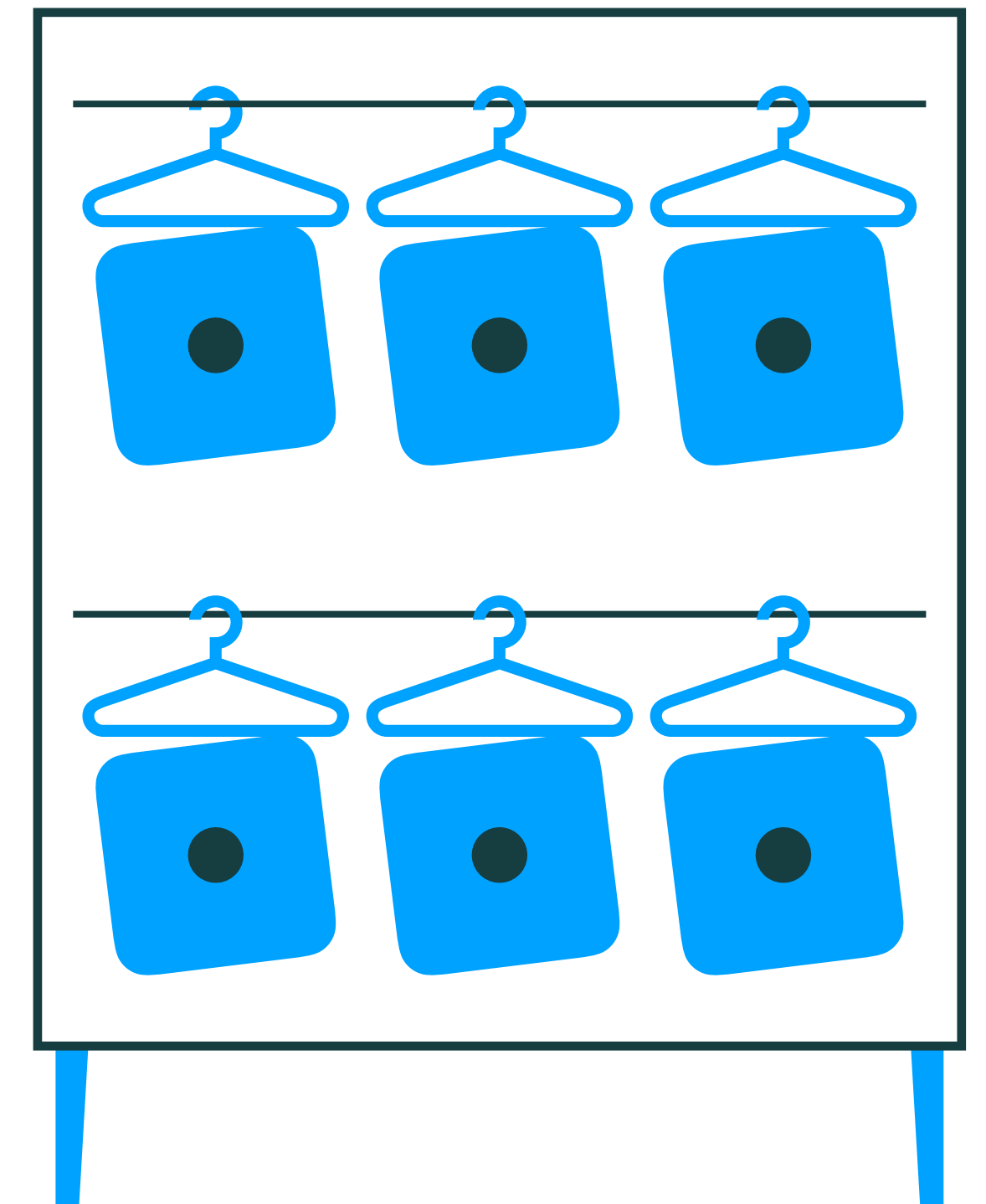
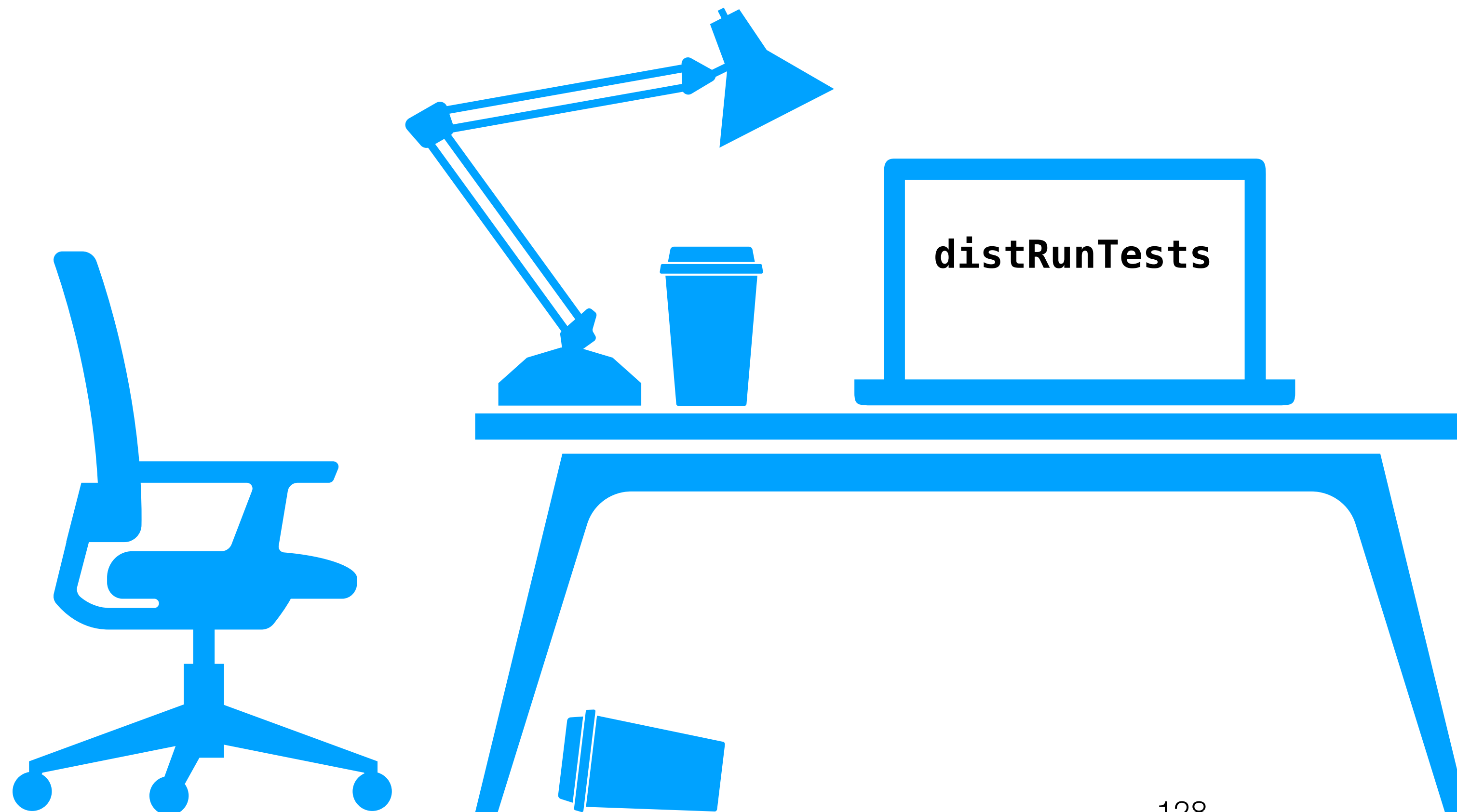
А я сейчас объясню



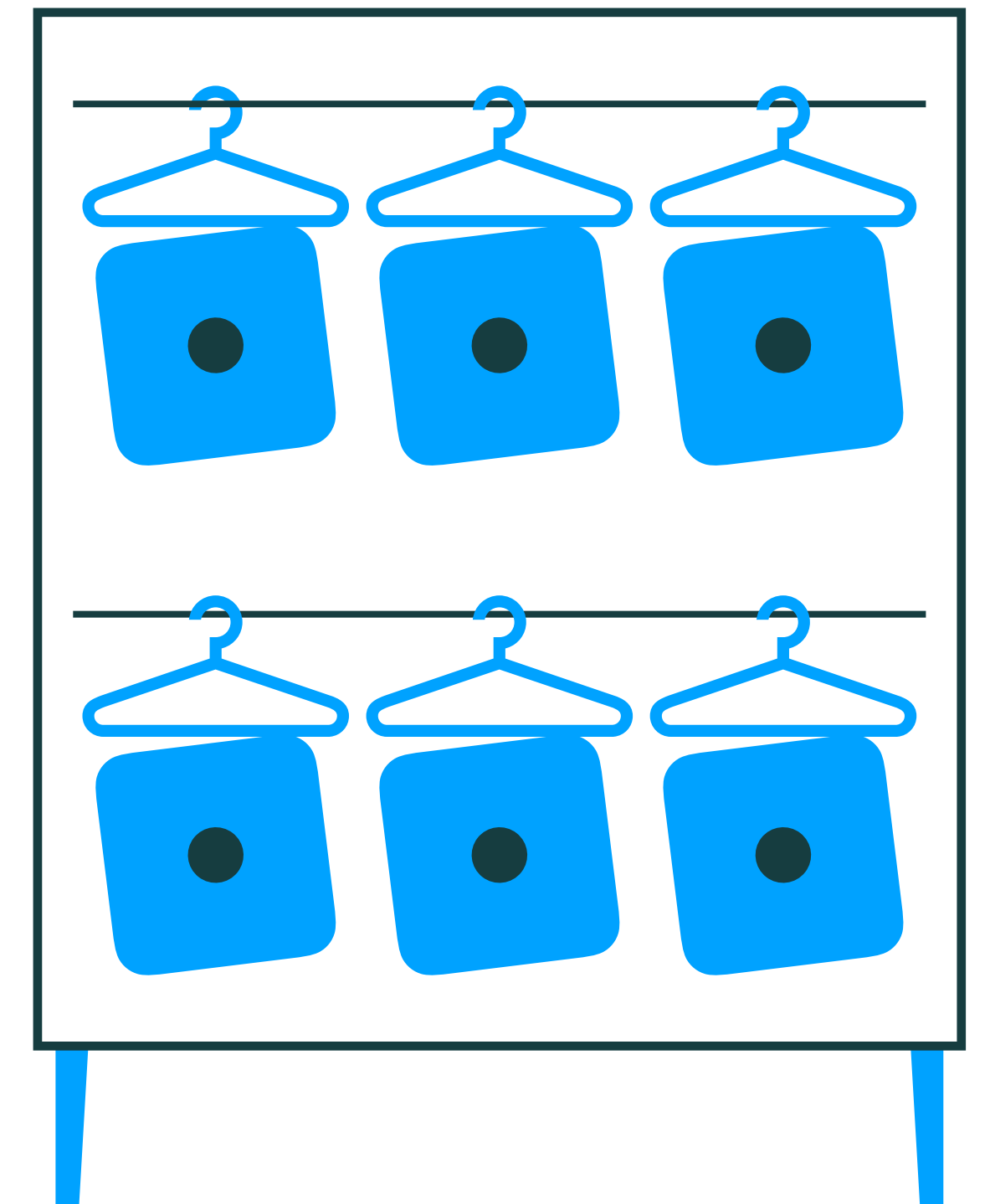
Запуск тестов распределенно



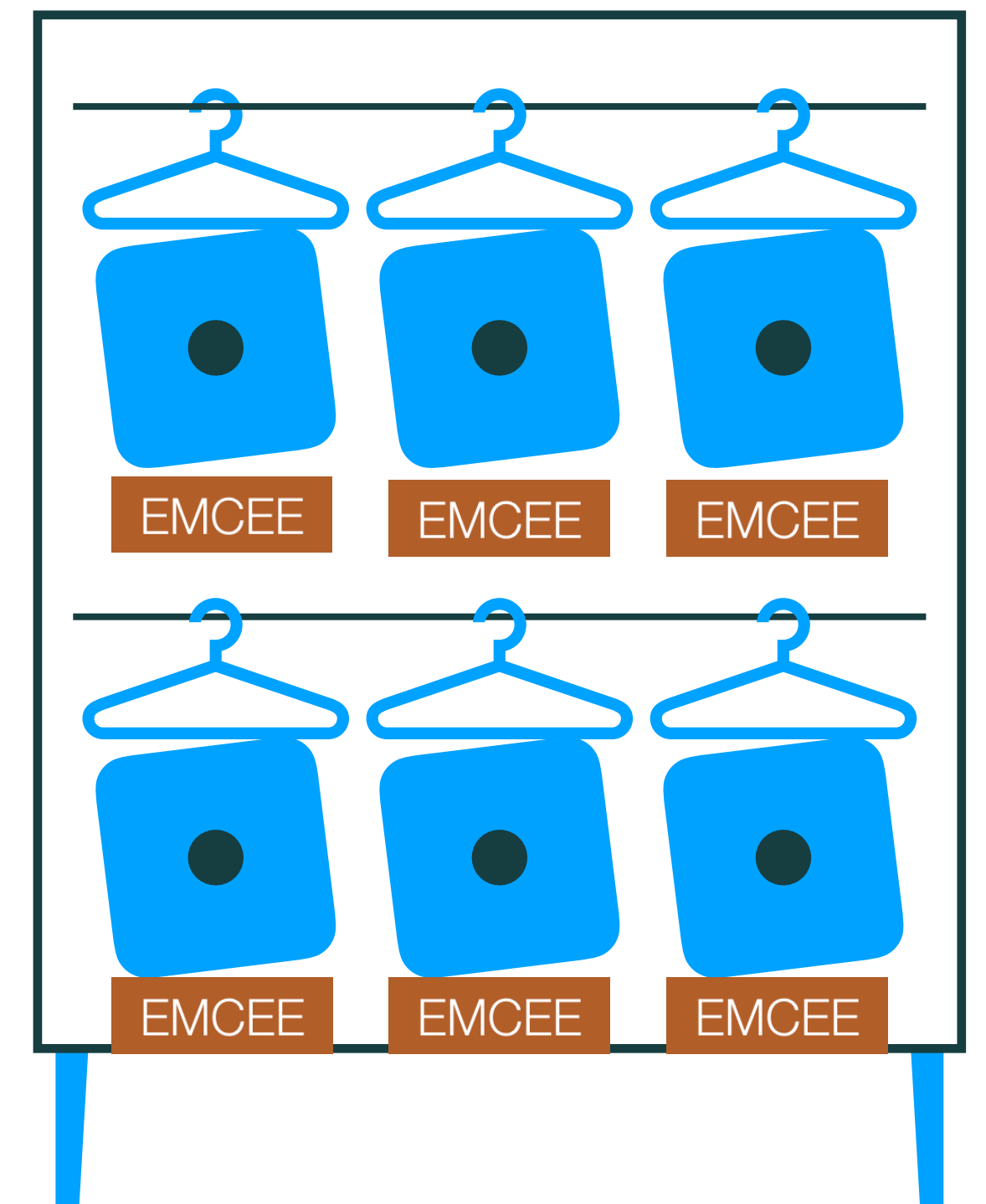
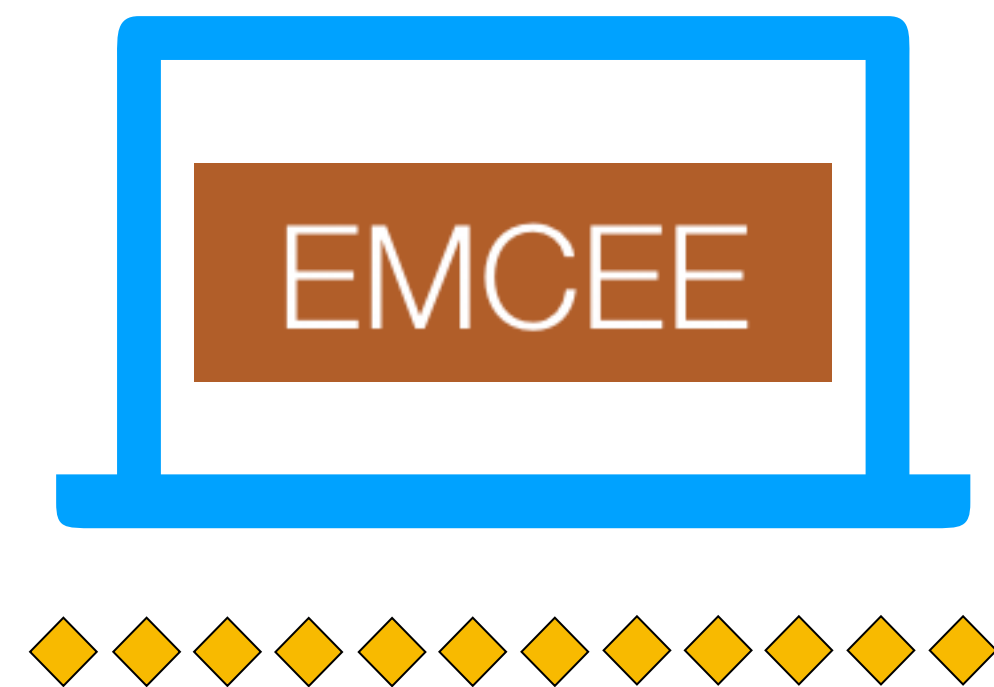
Запуск тестов распределенно



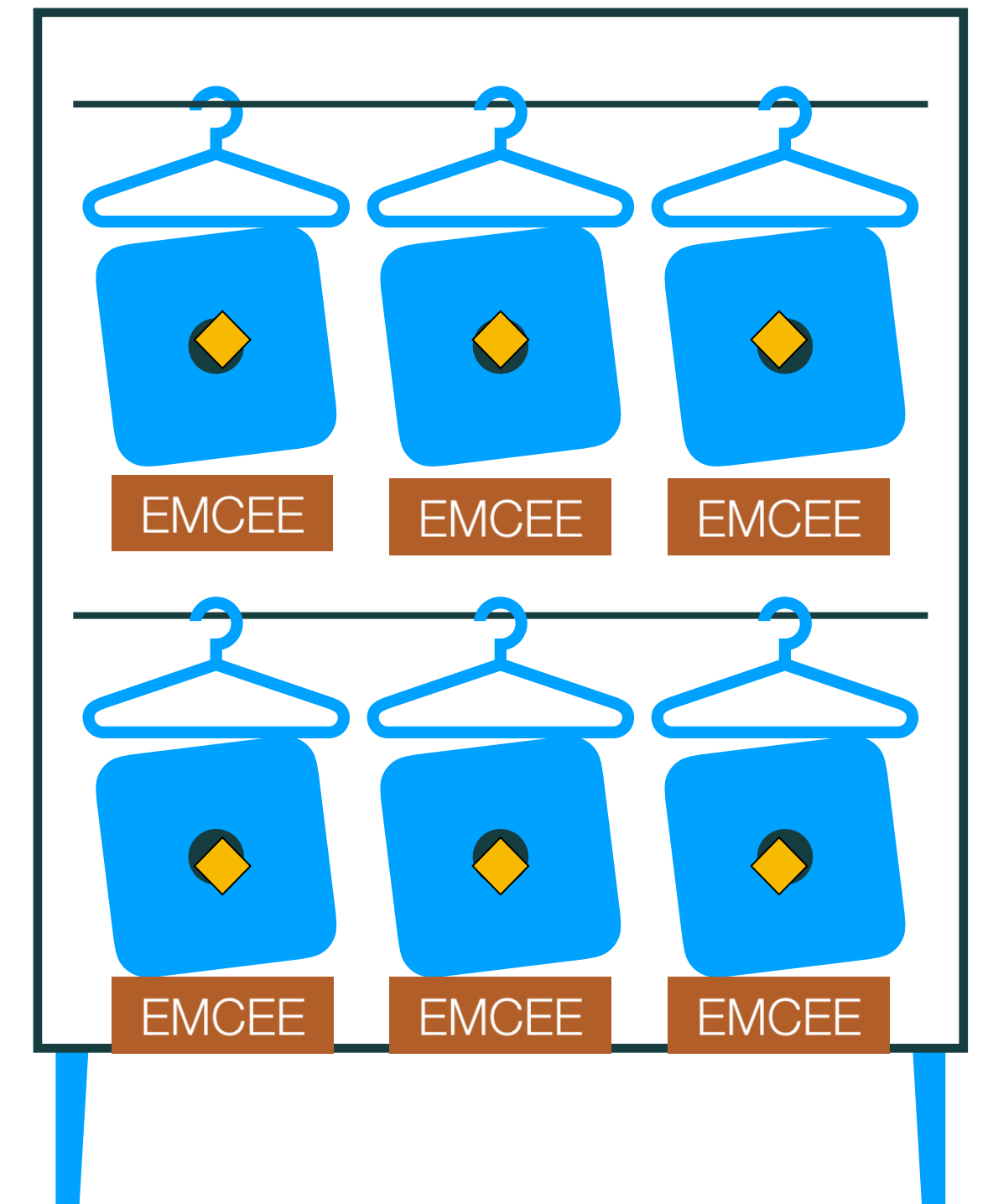
Запуск тестов распределенно



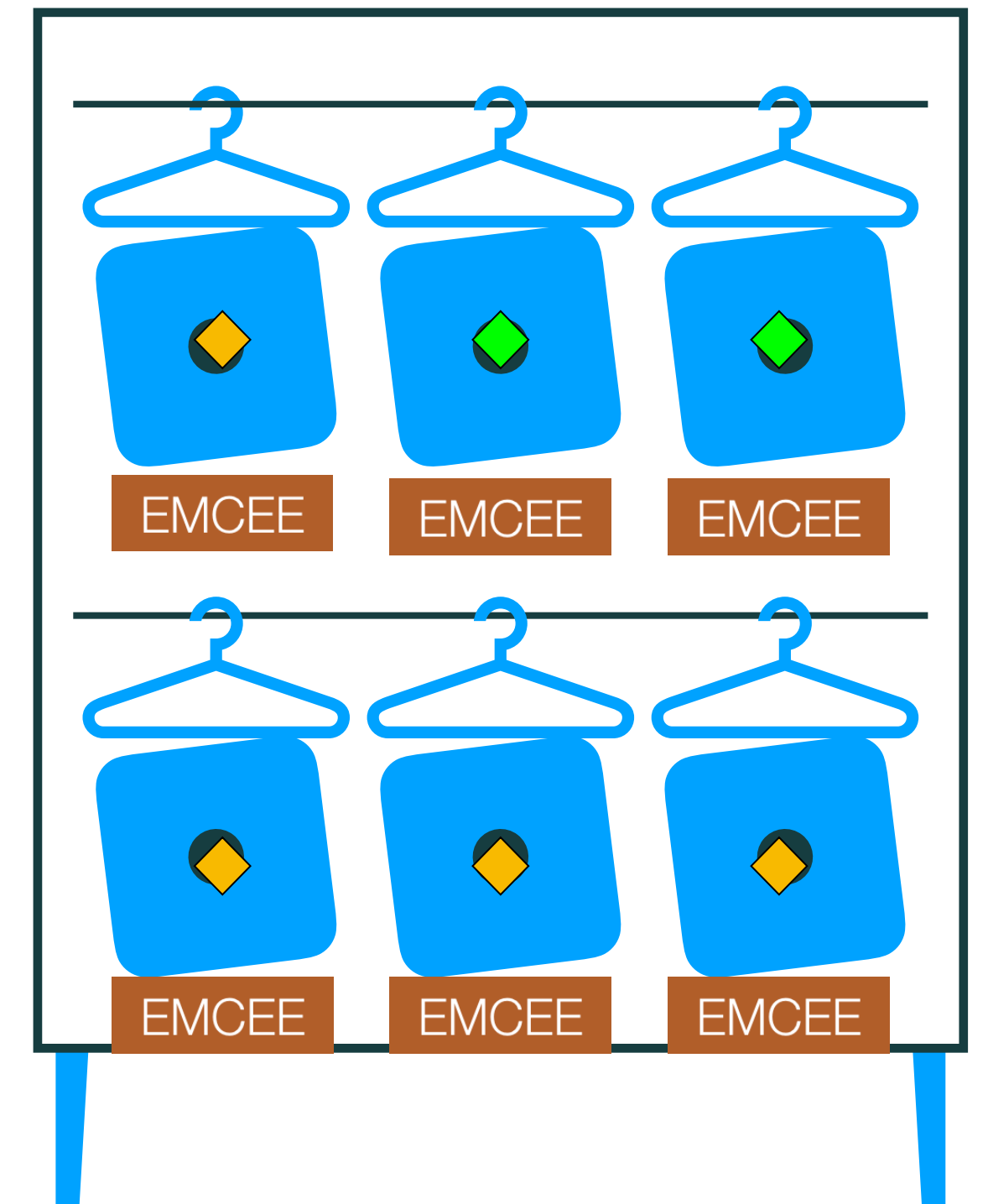
Запуск тестов распределенно



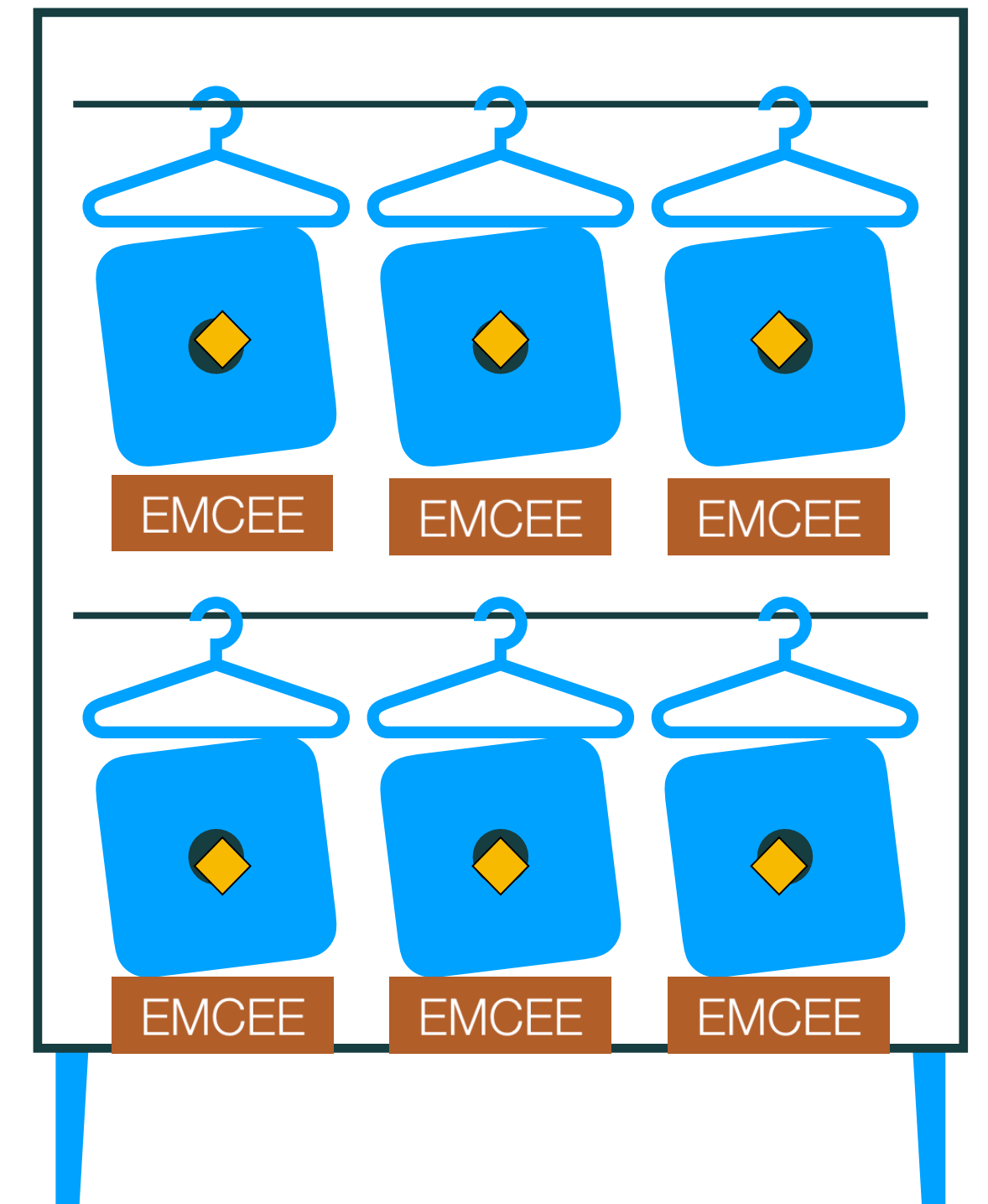
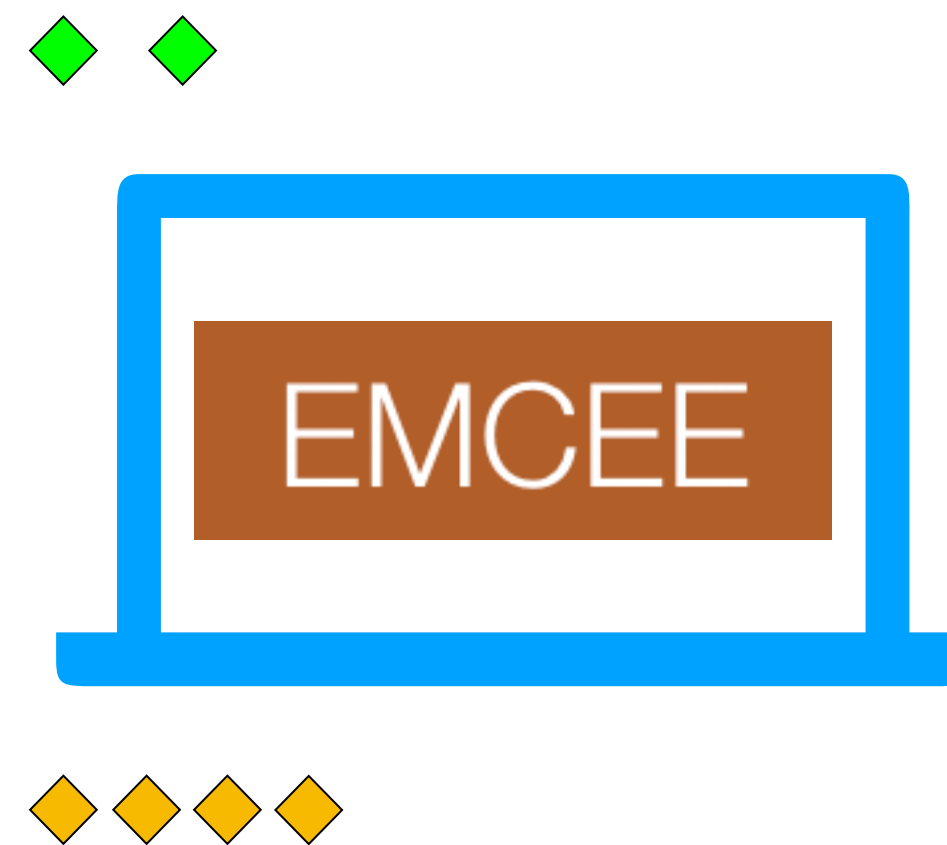
Запуск тестов распределенно



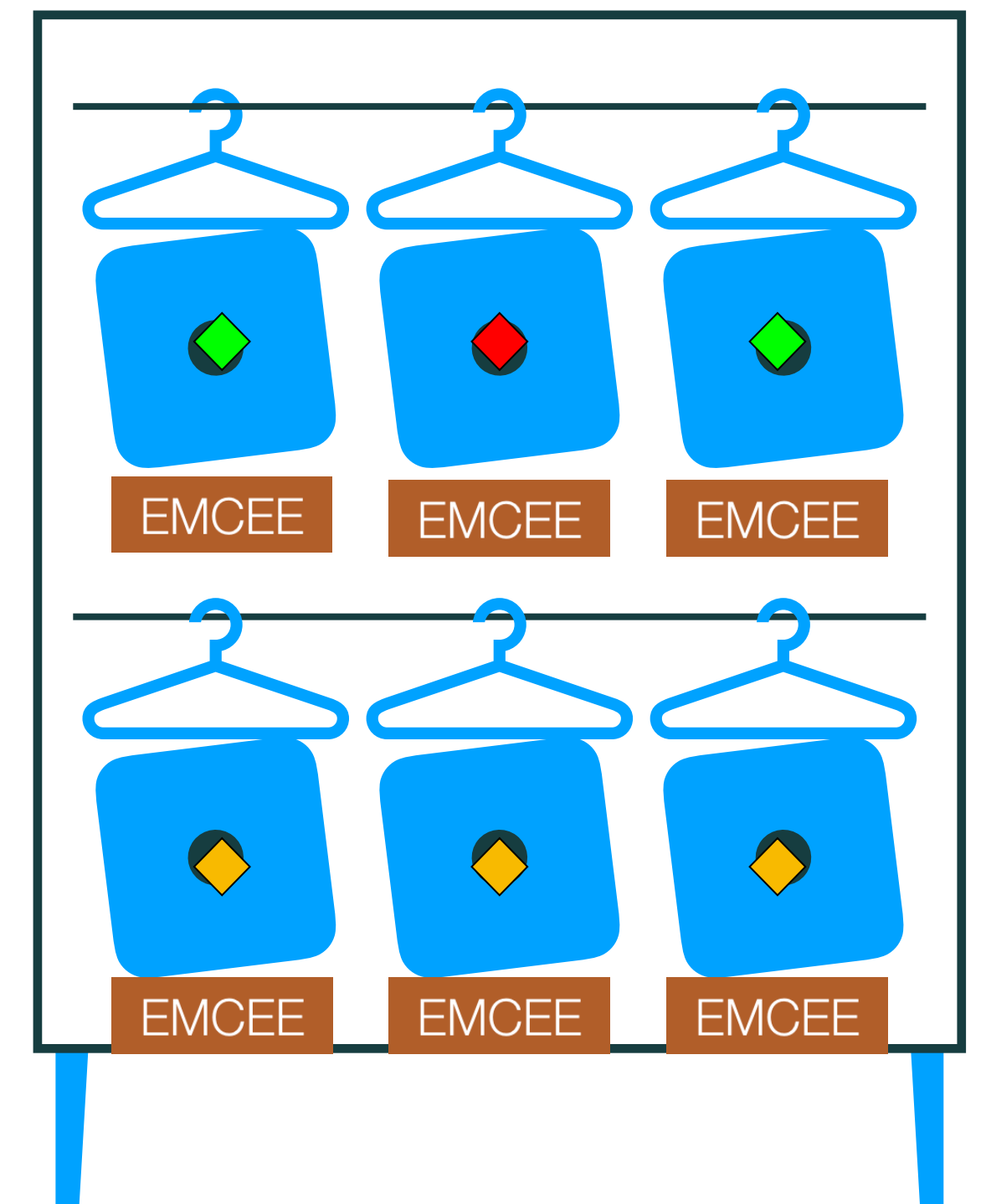
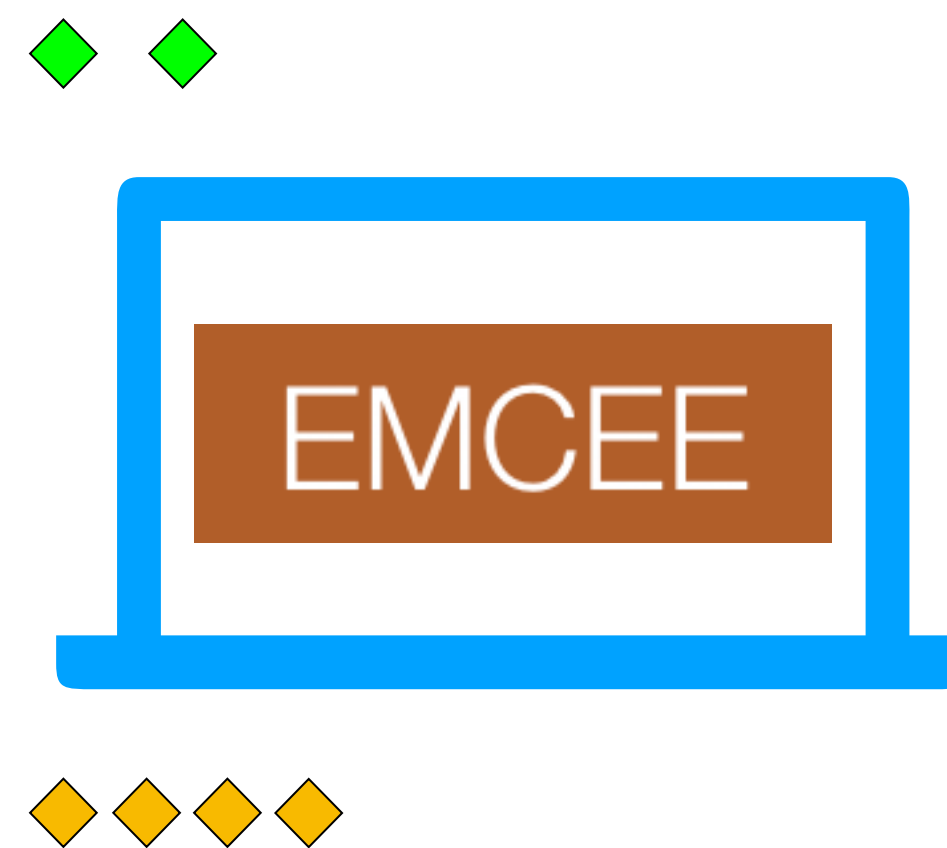
Запуск тестов распределенно



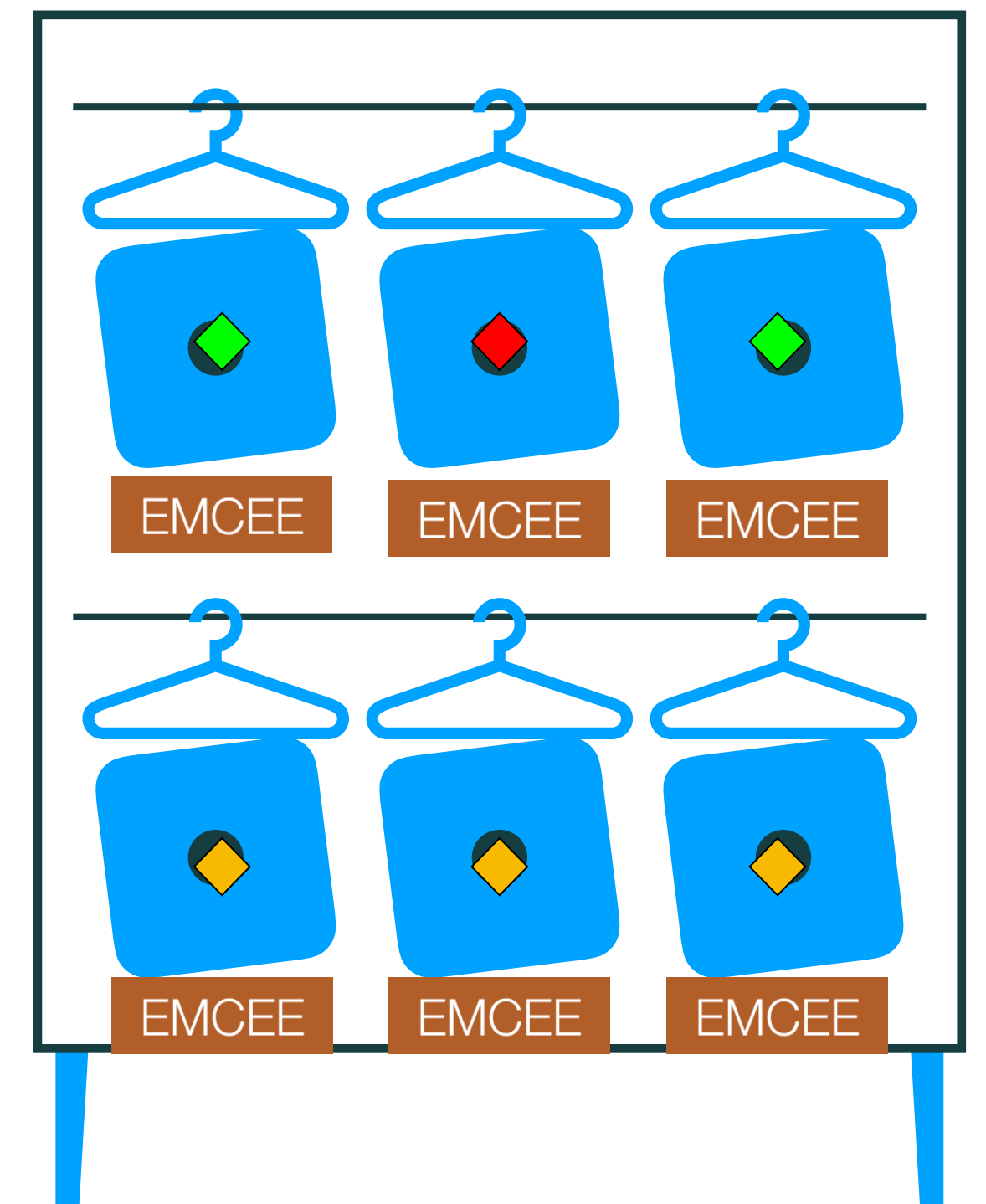
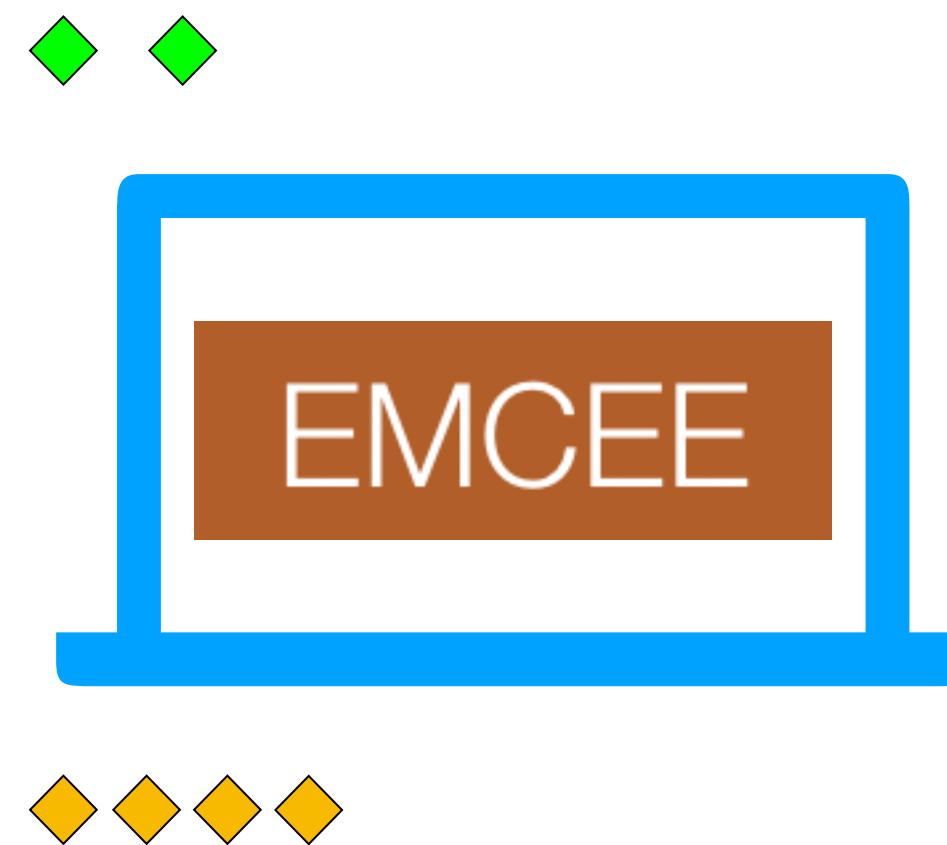
Запуск тестов распределенно



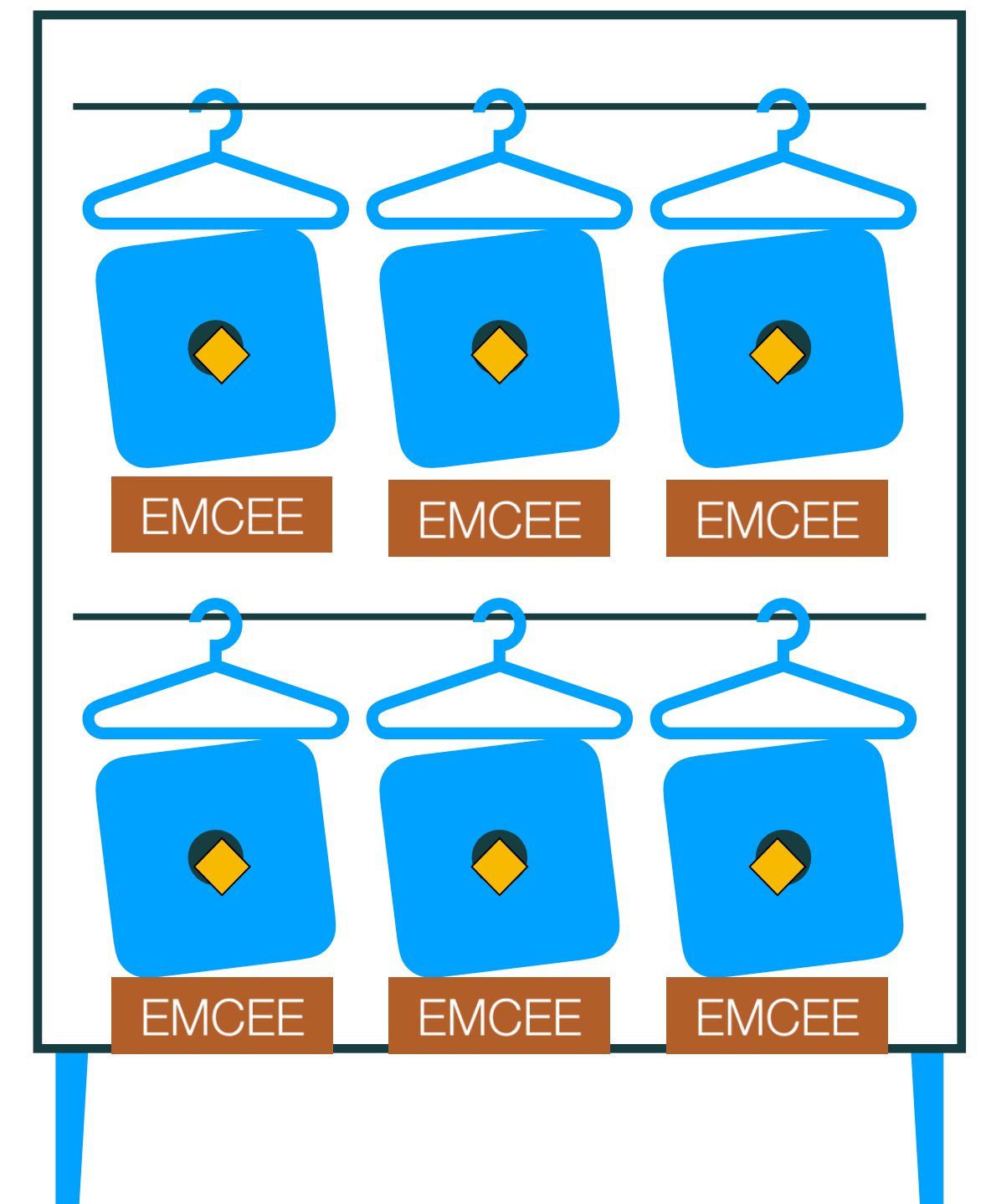
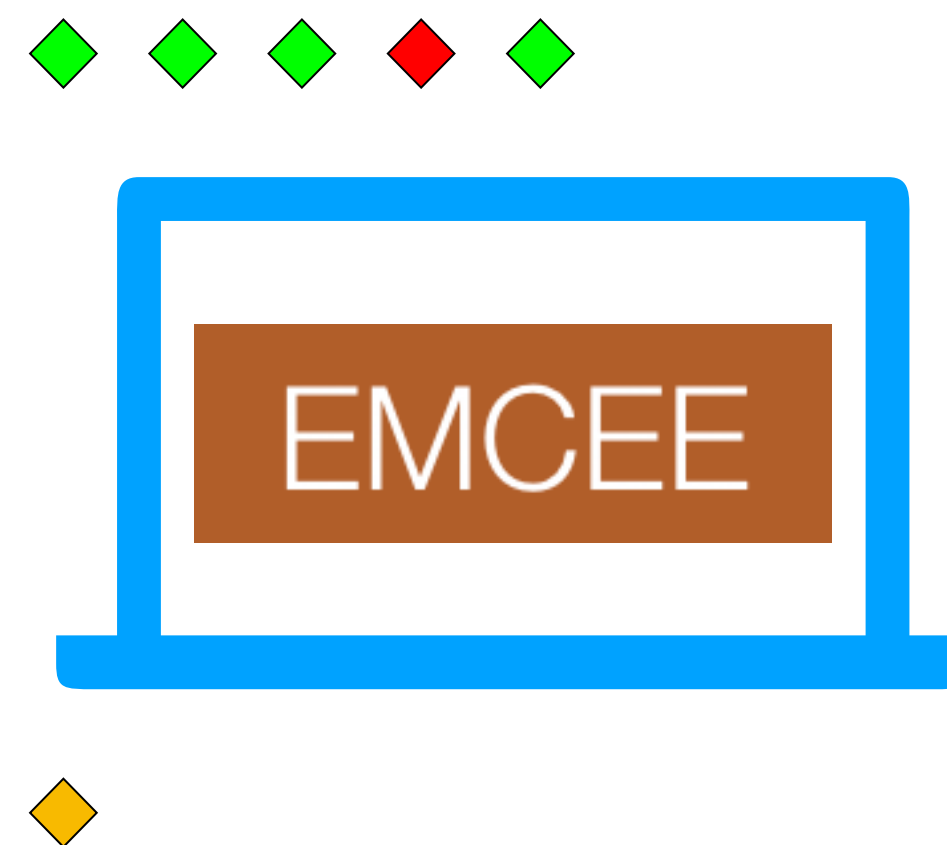
Запуск тестов распределенно



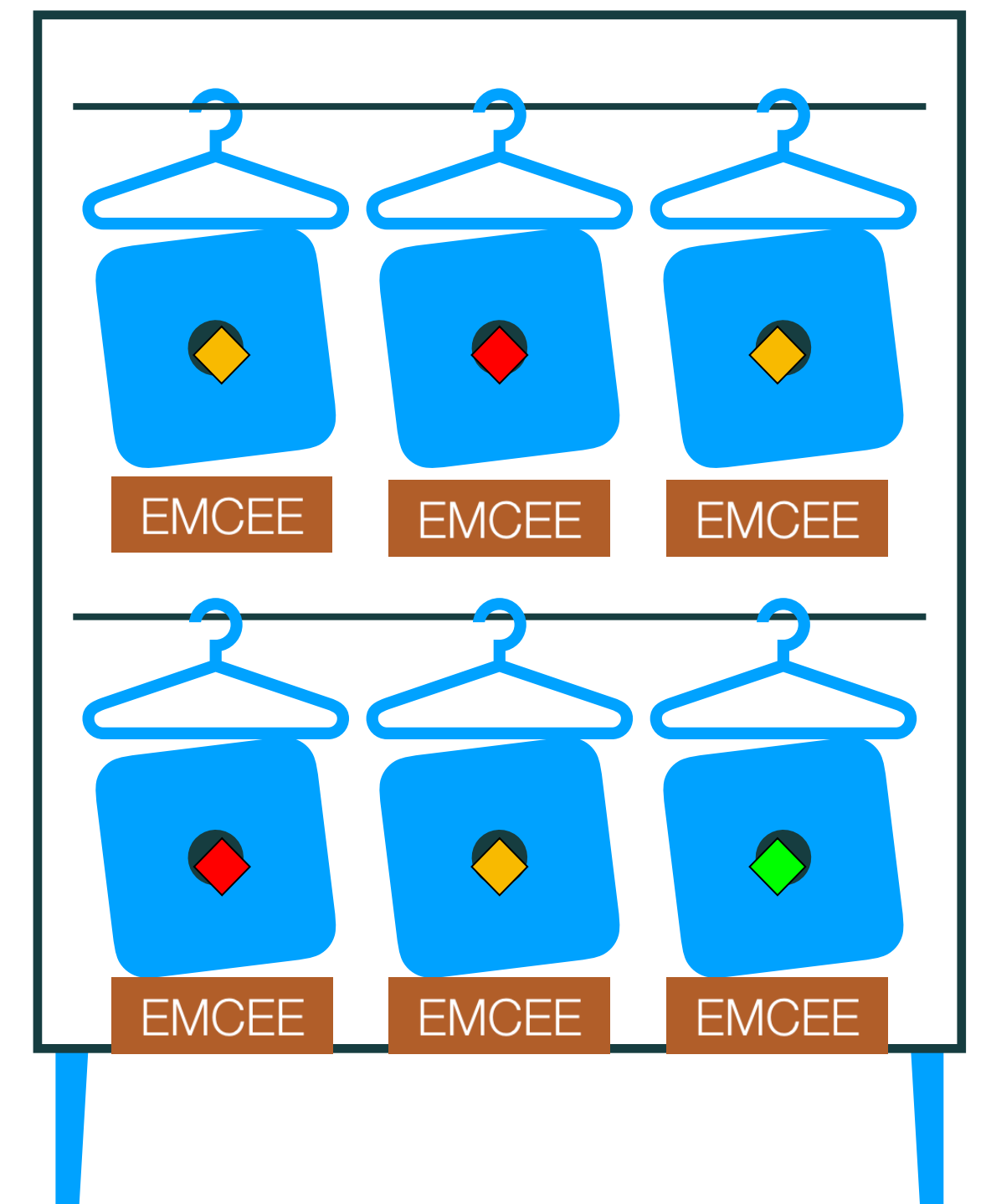
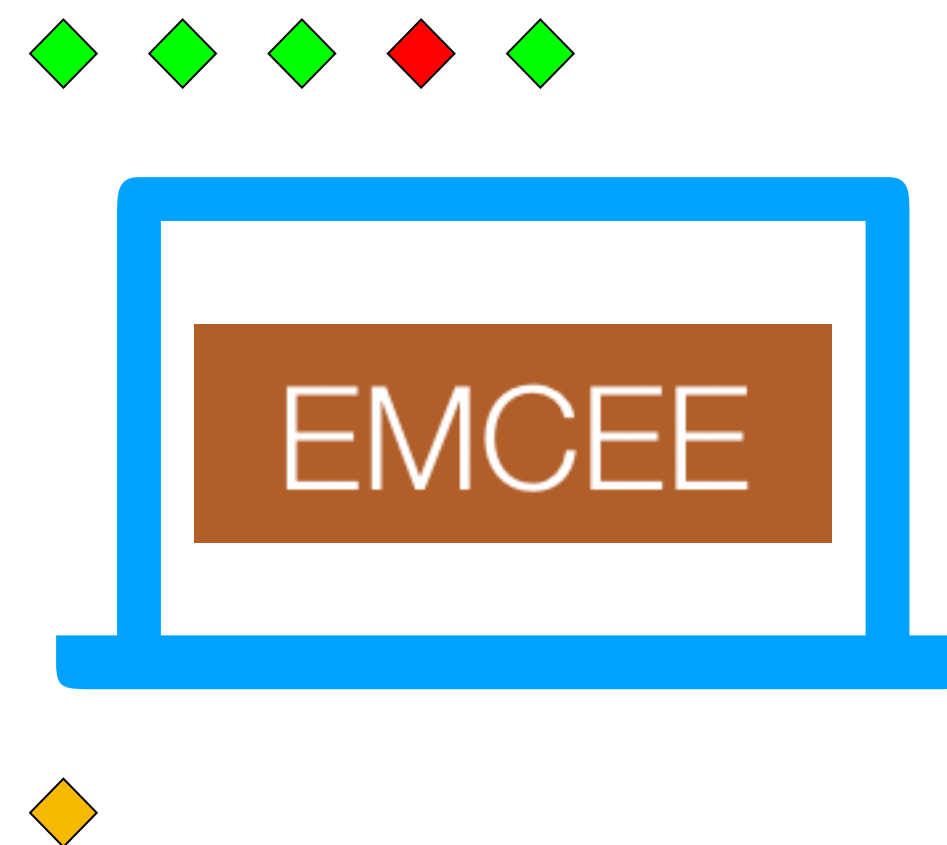
Запуск тестов распределенно



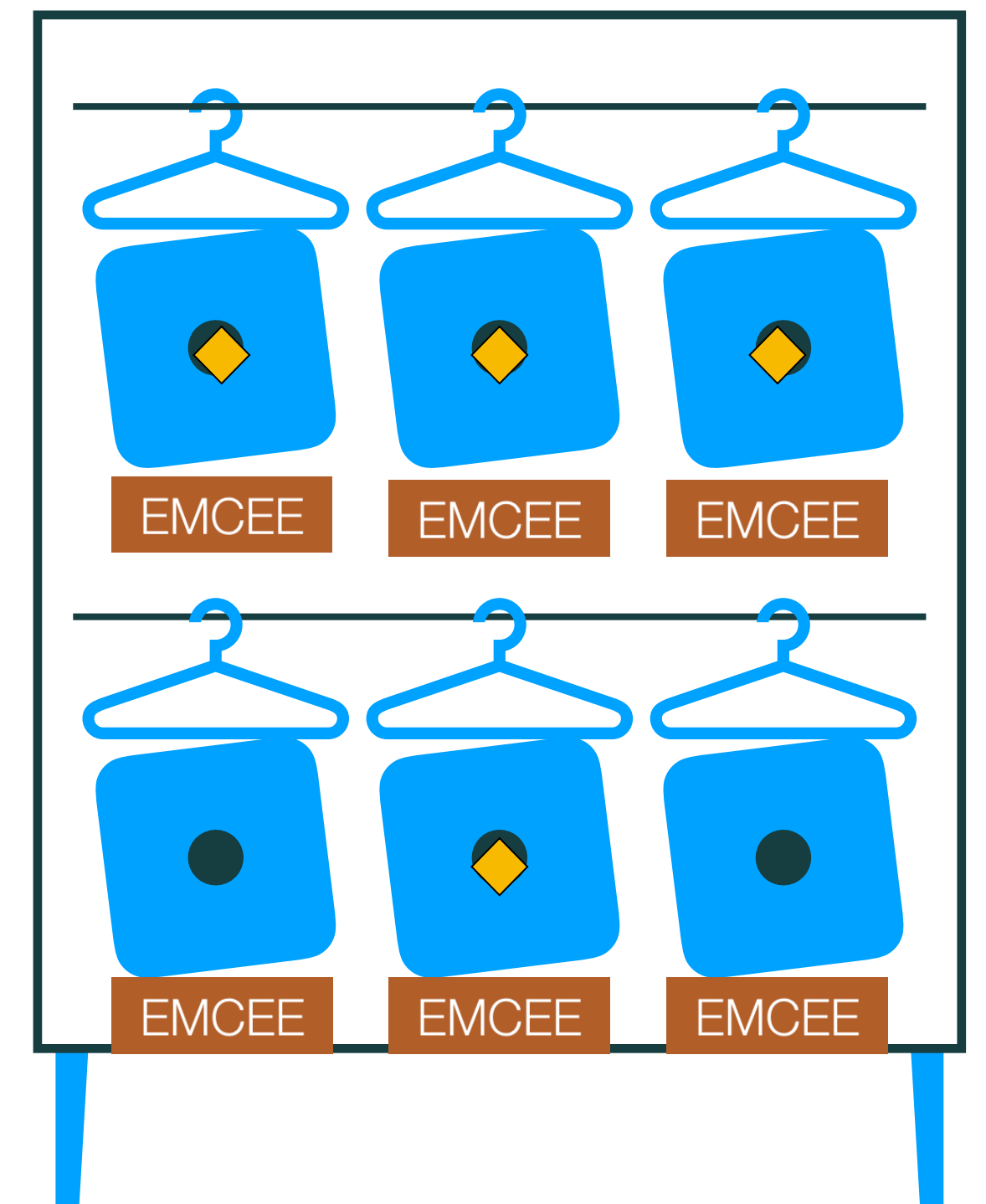
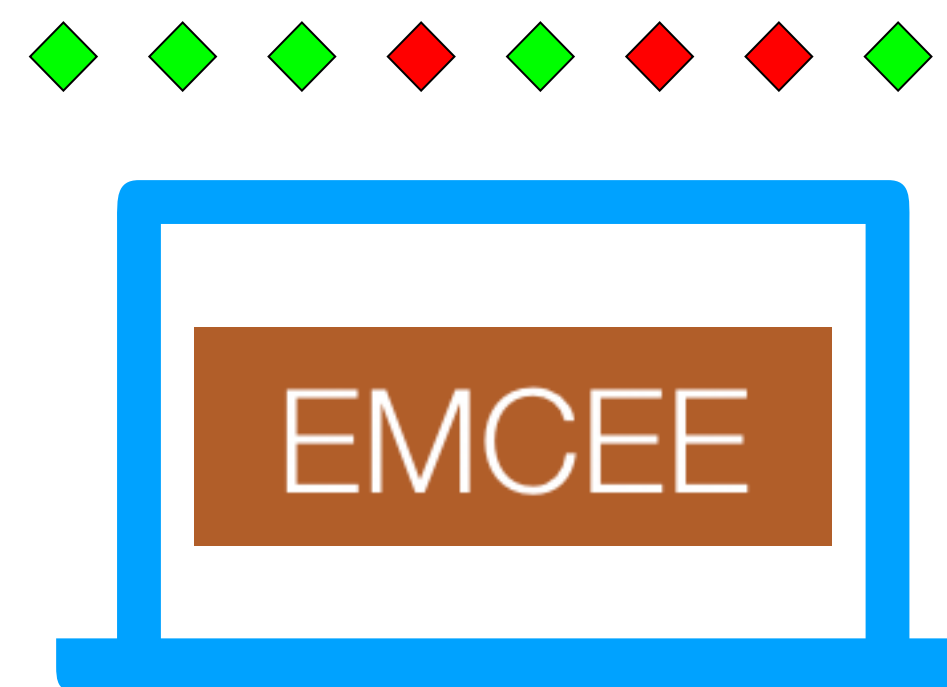
Запуск тестов распределенно



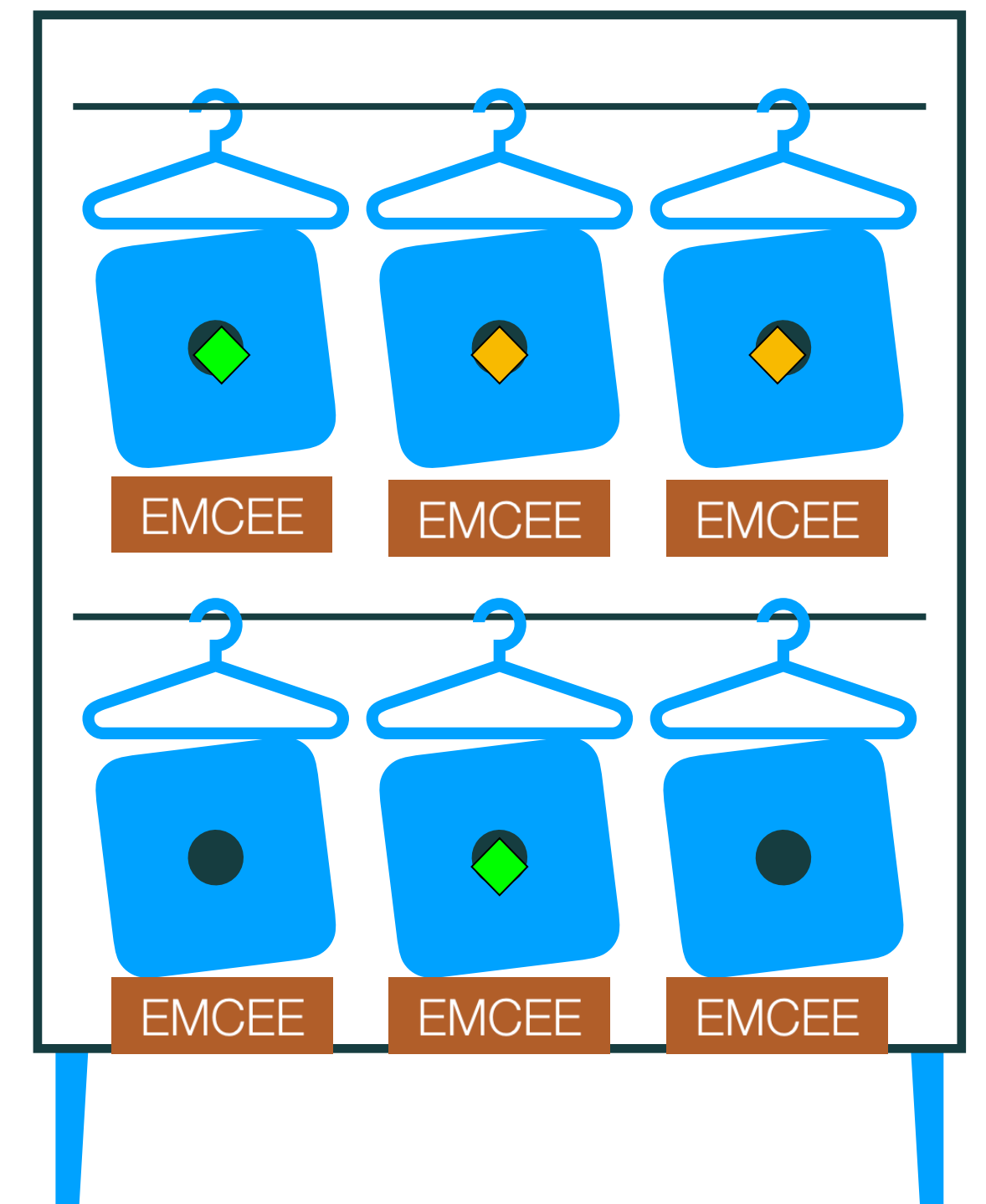
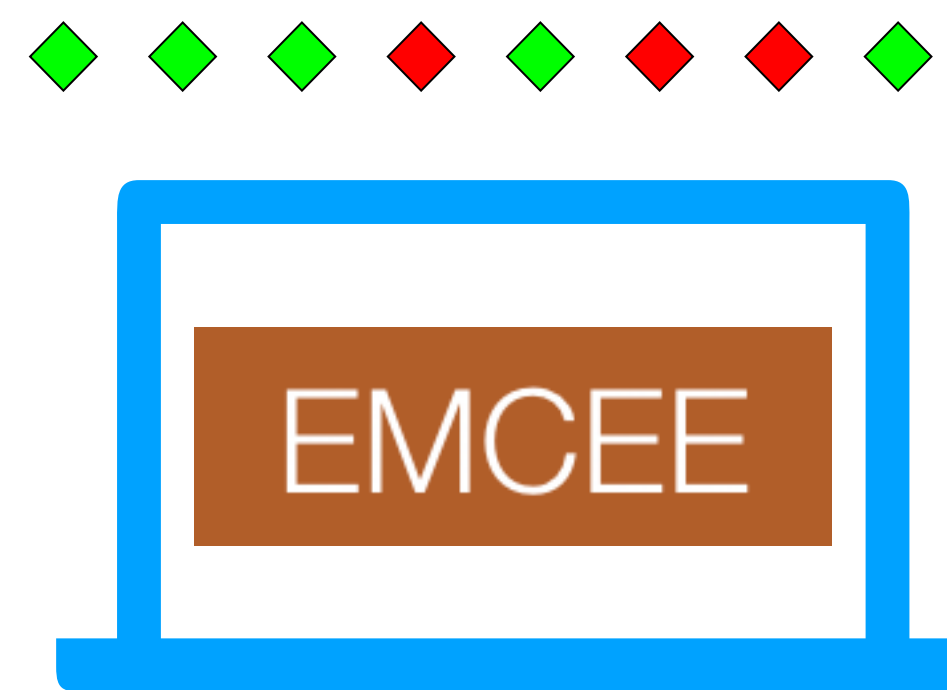
Запуск тестов распределенно



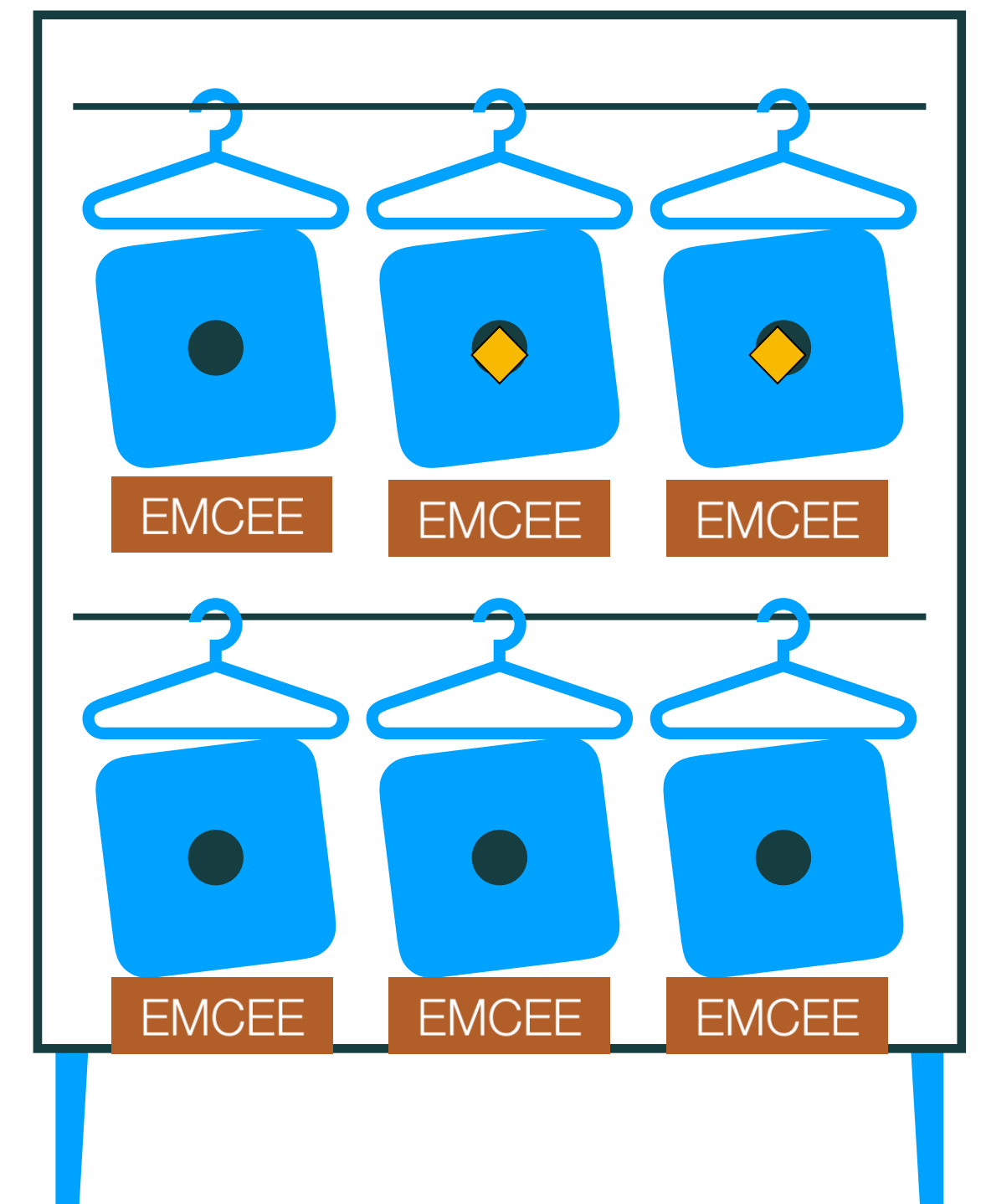
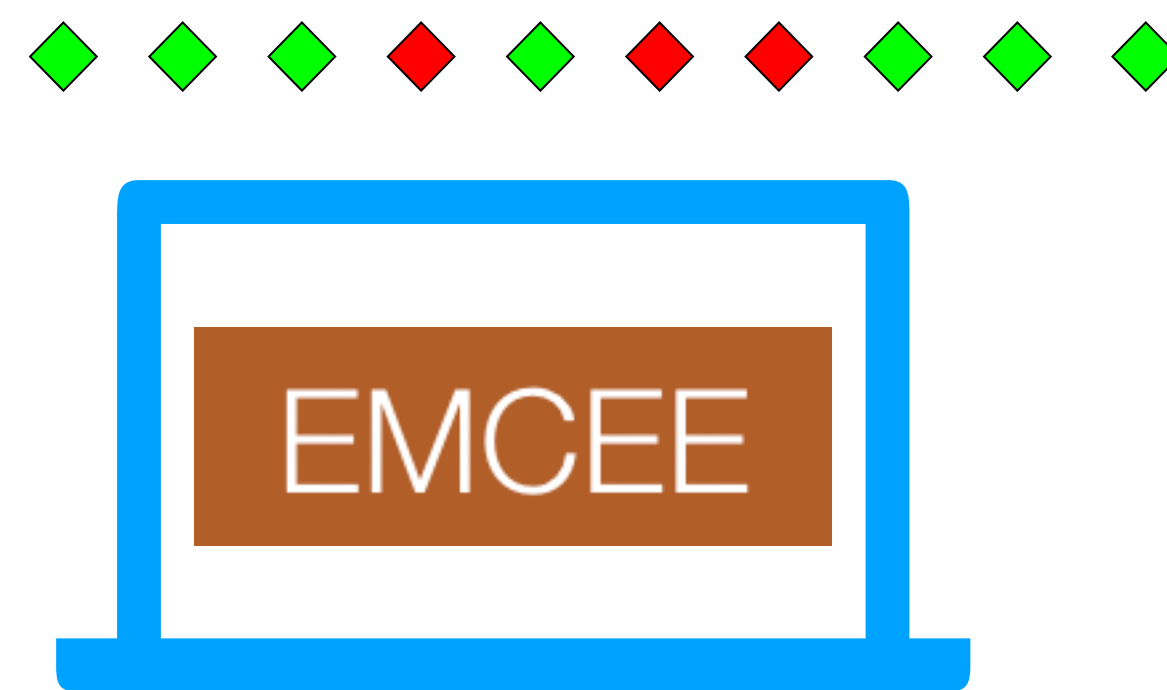
Запуск тестов распределенно



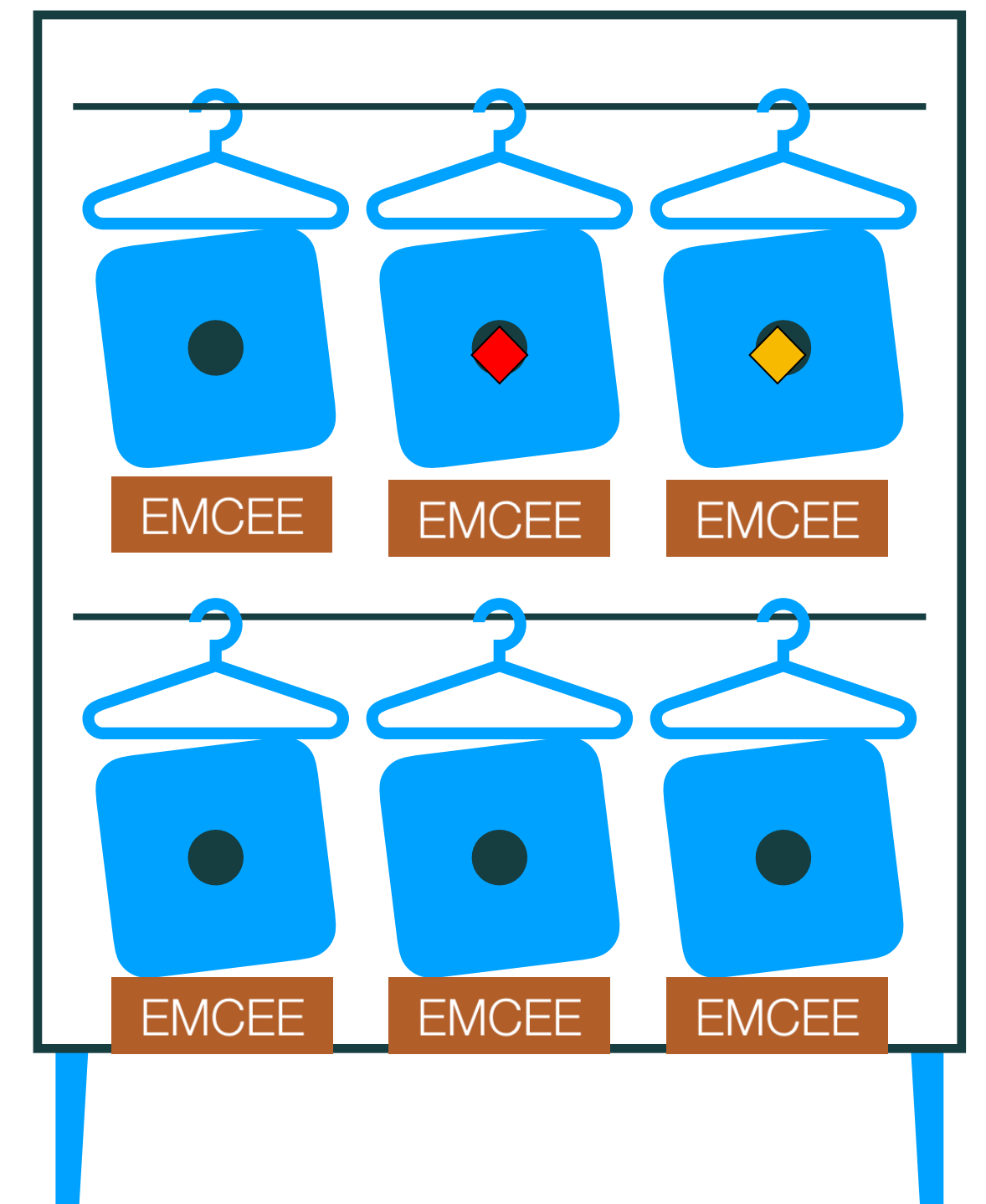
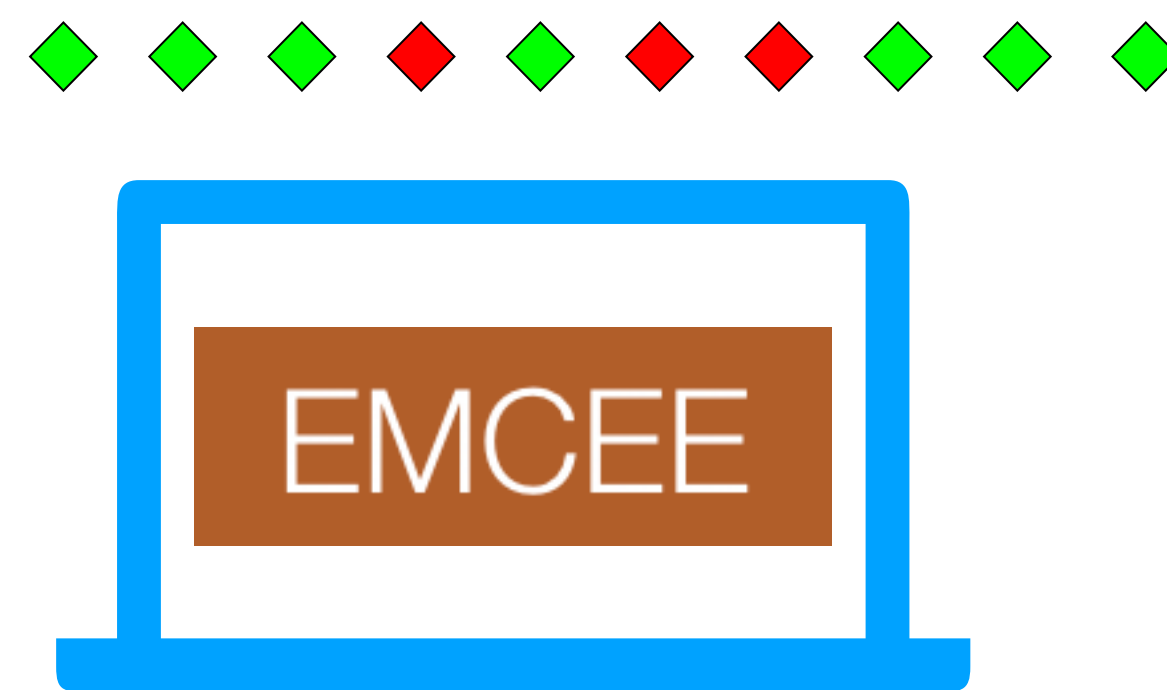
Запуск тестов распределенно



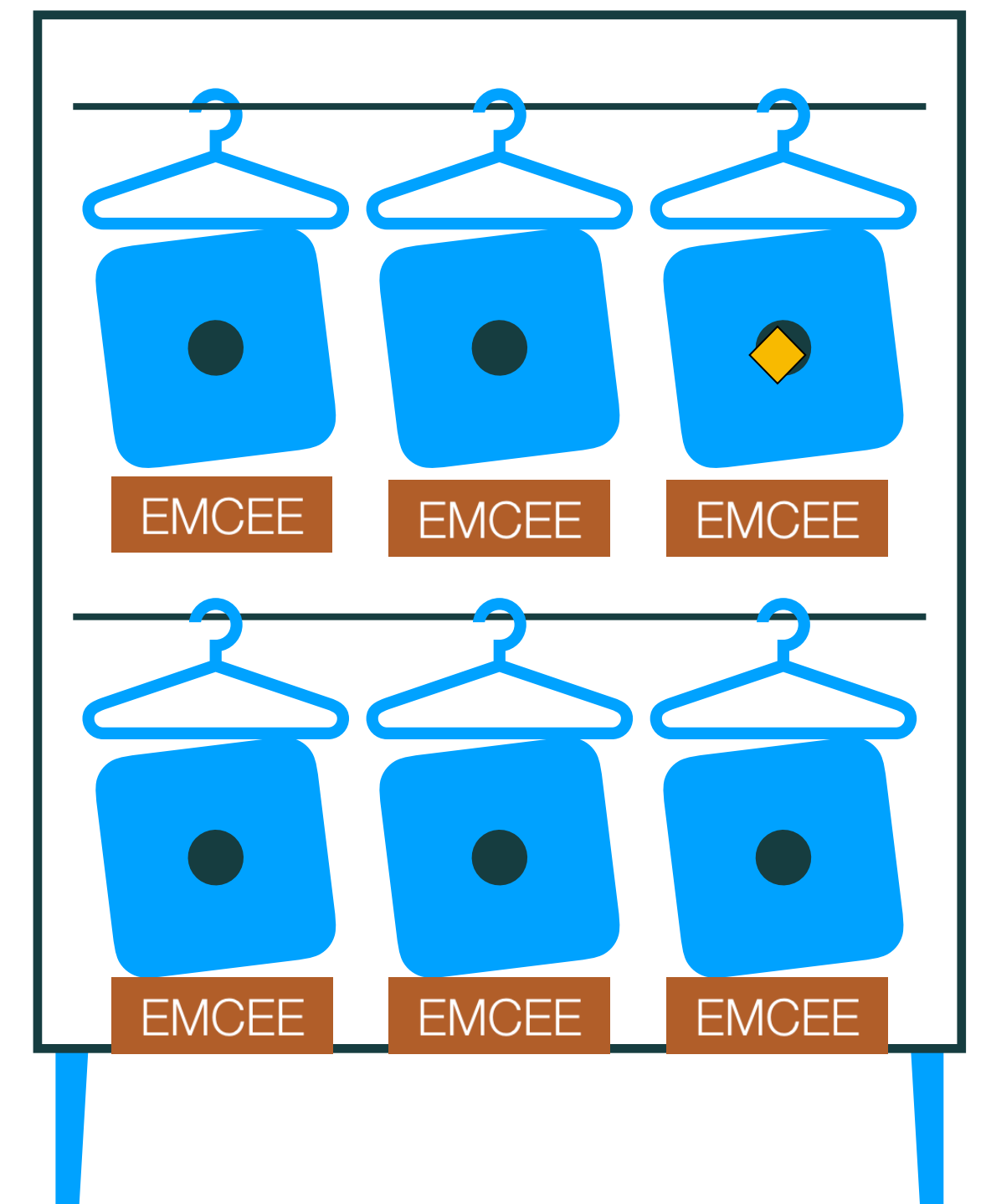
Запуск тестов распределенно



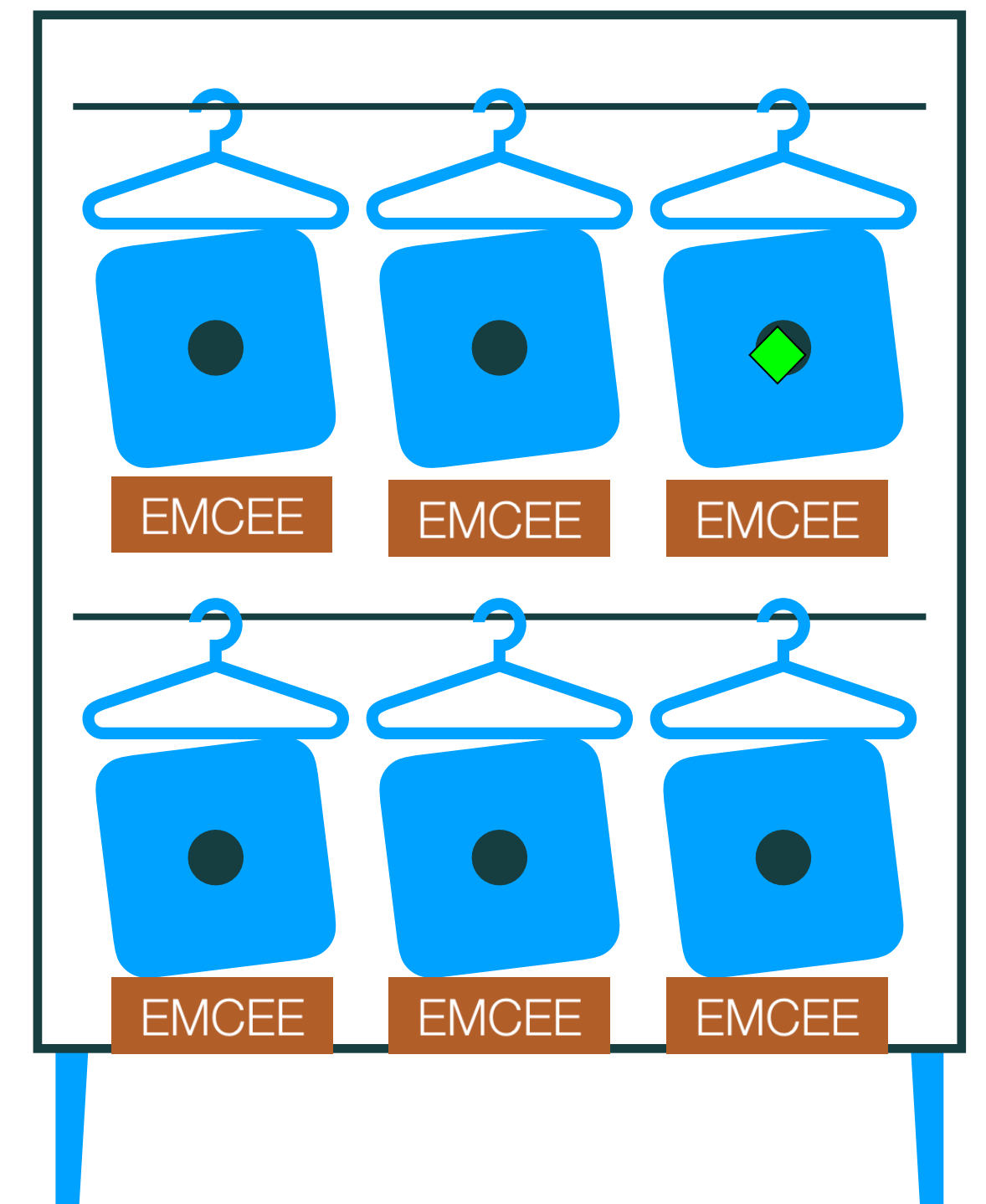
Запуск тестов распределенно



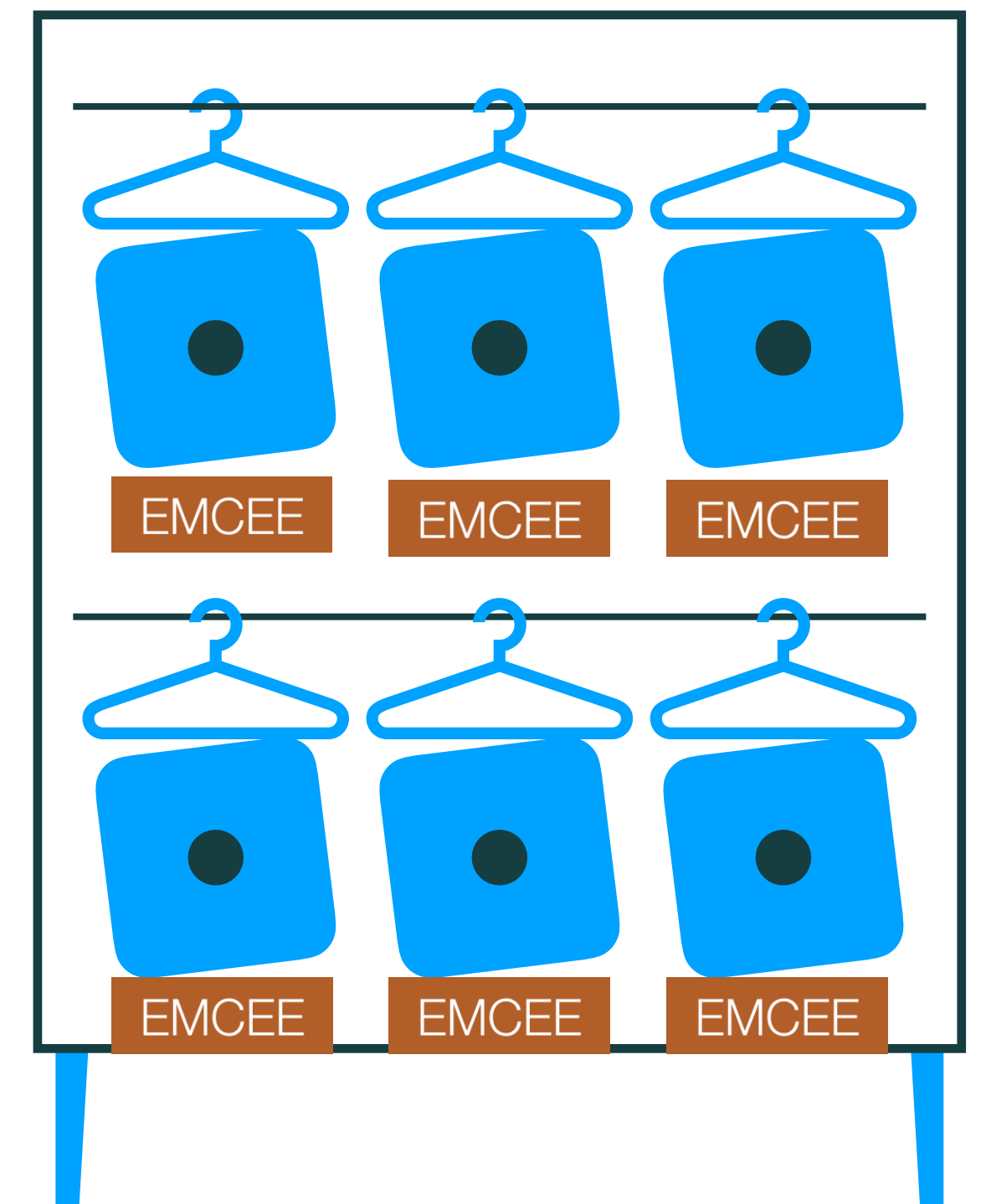
Запуск тестов распределенно



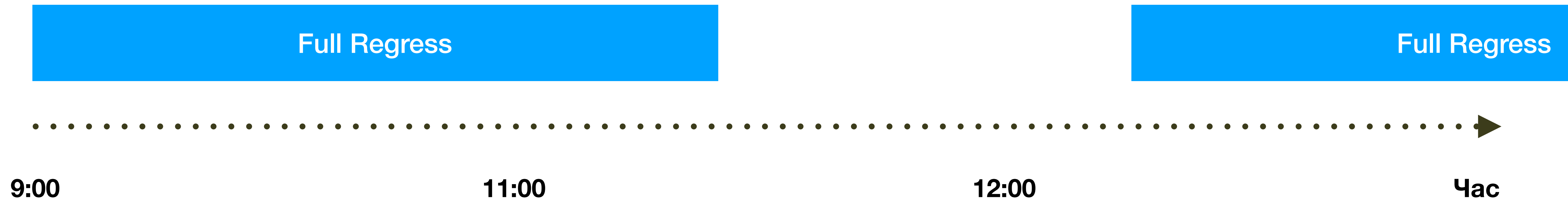
Запуск тестов распределенно



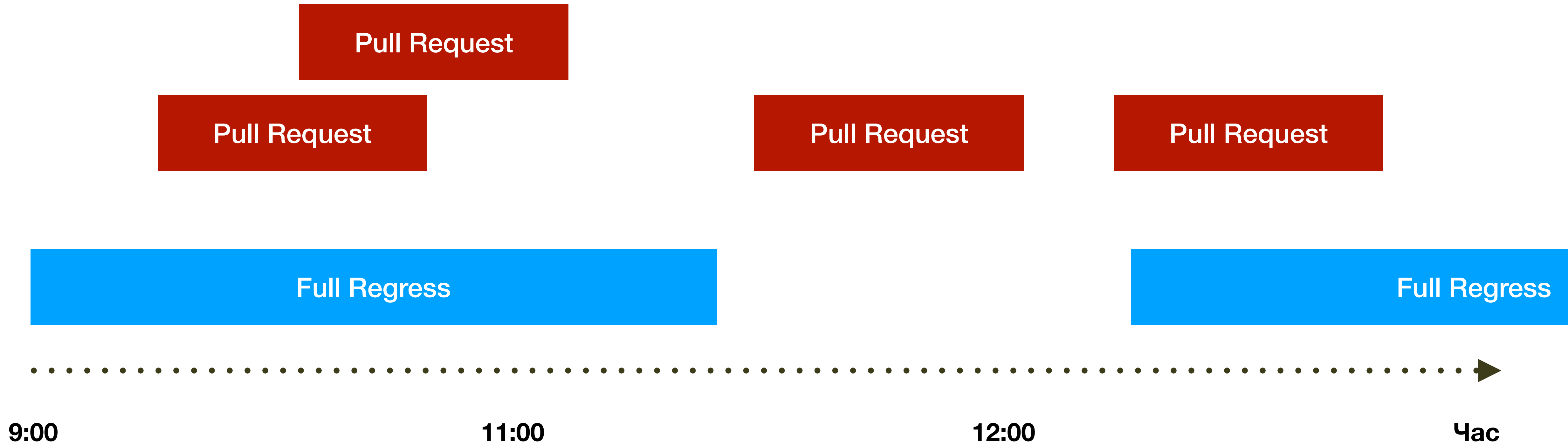
Запуск тестов распределенно



Джоббизация



Джобизация

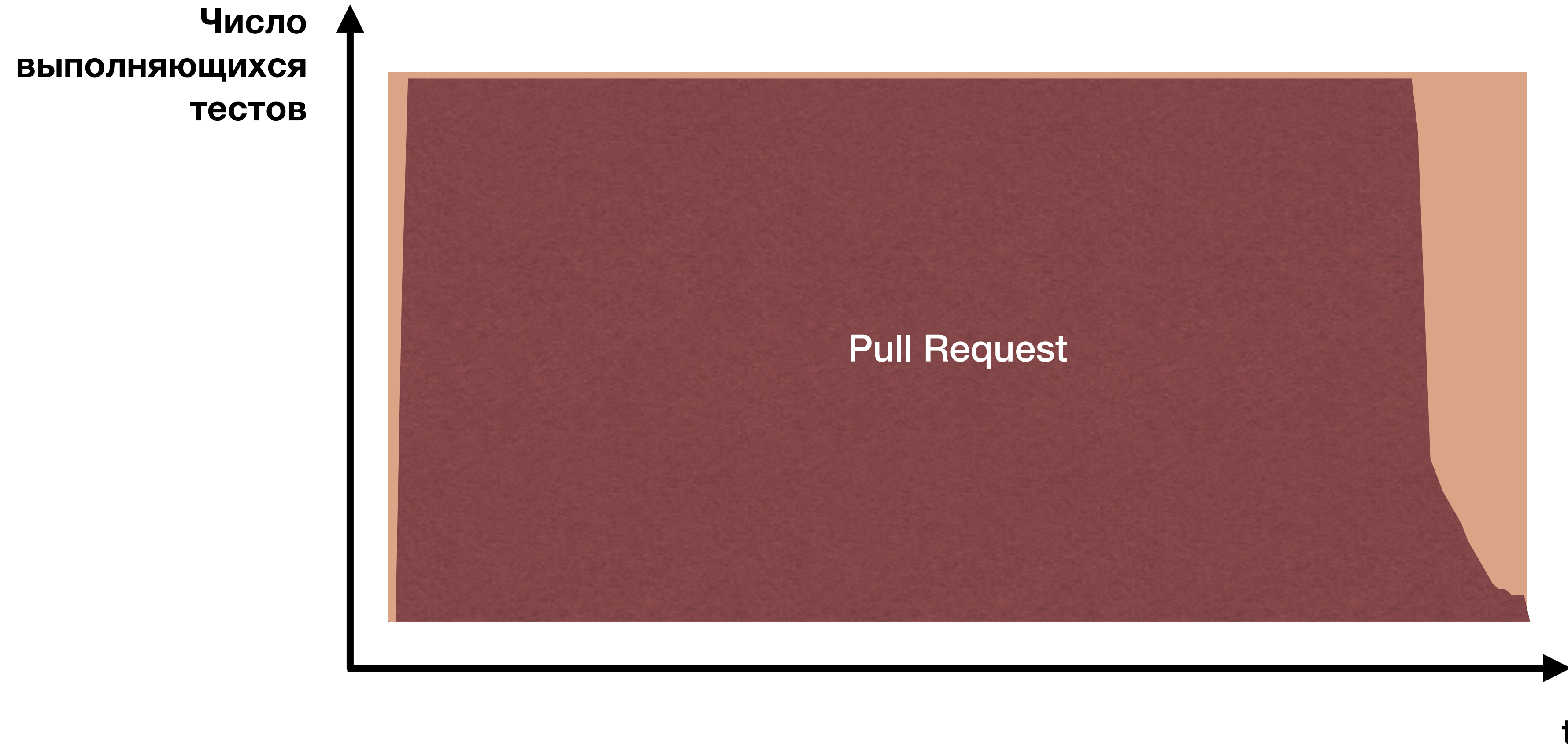


Выполнение тестов

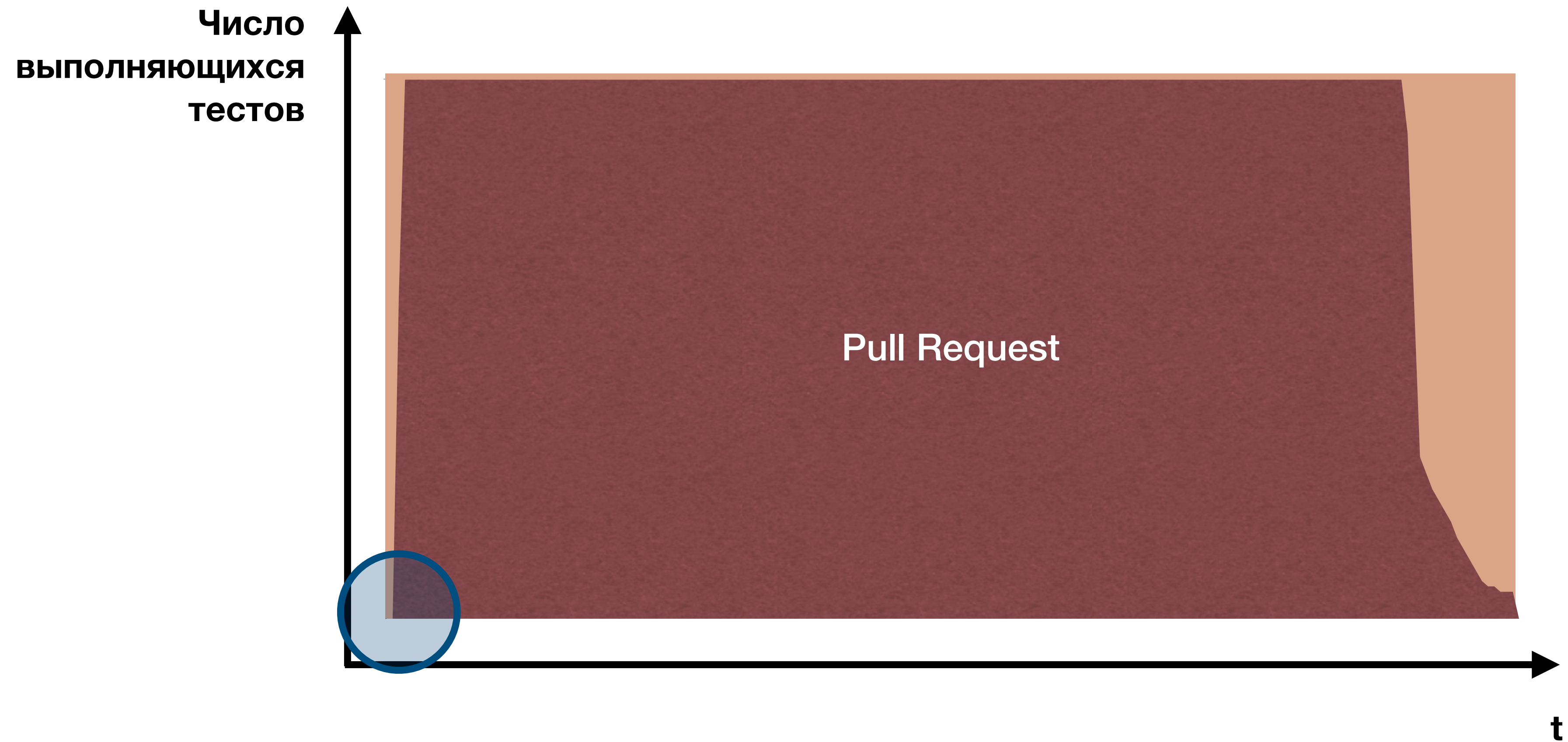


Pull Request

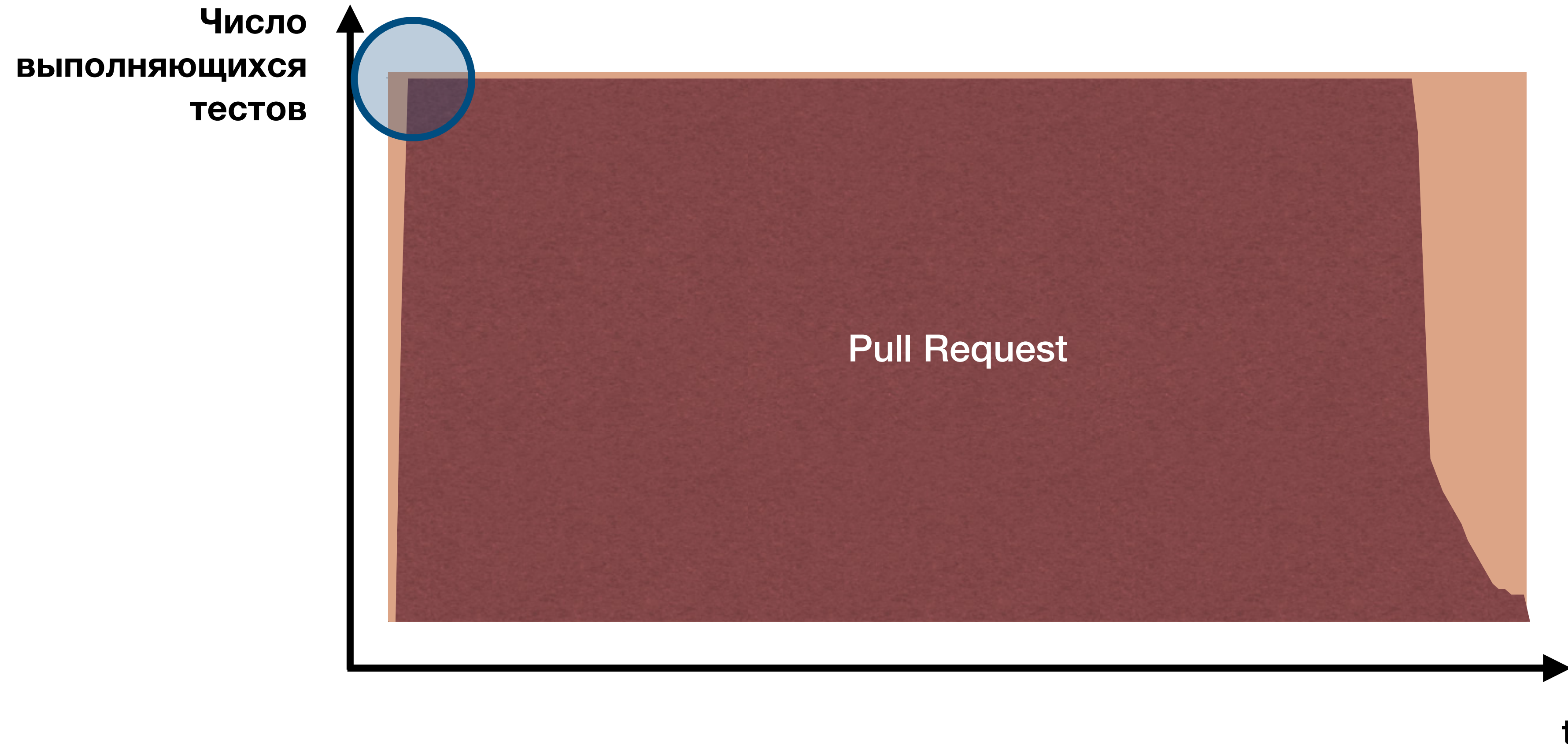
Выполнение тестов



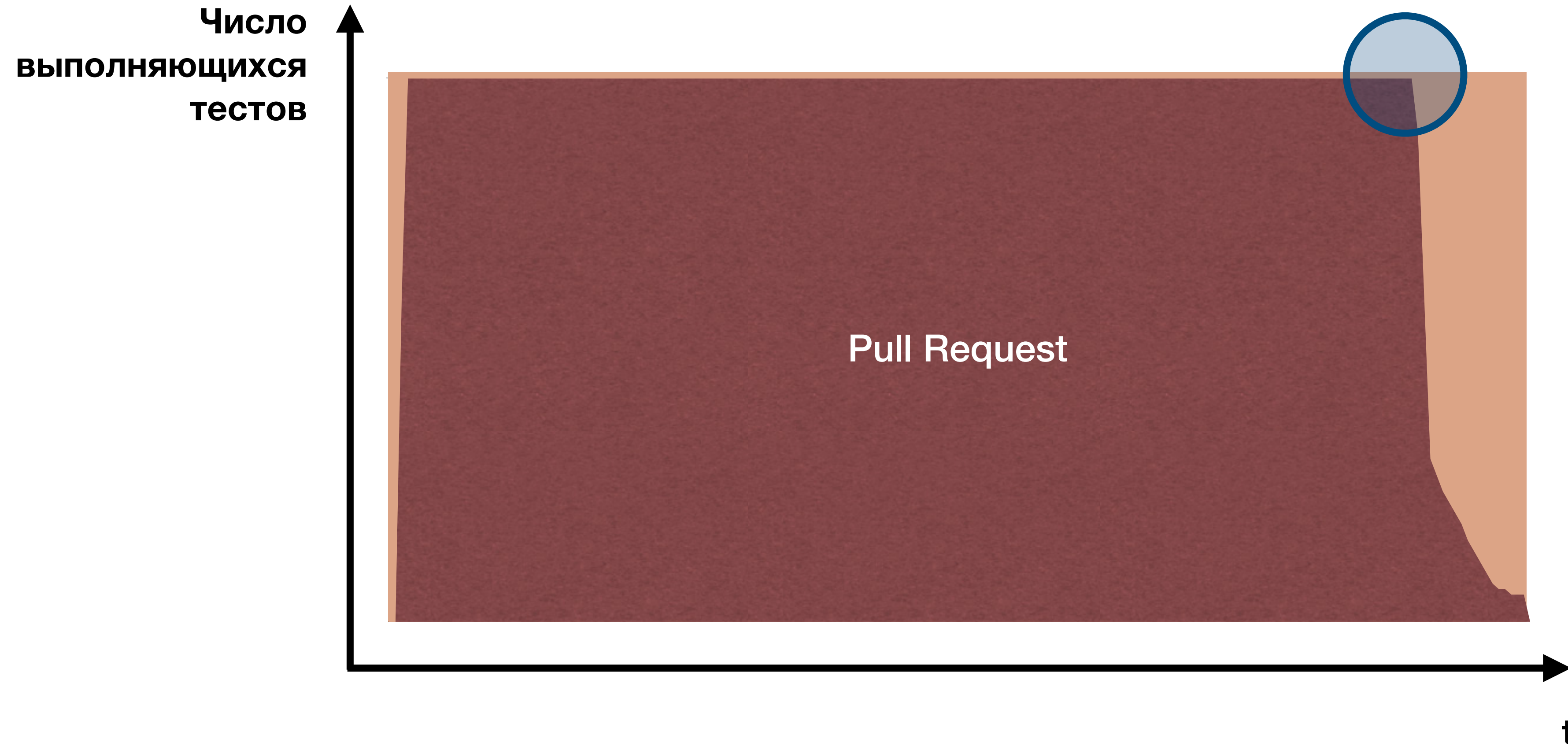
Выполнение тестов



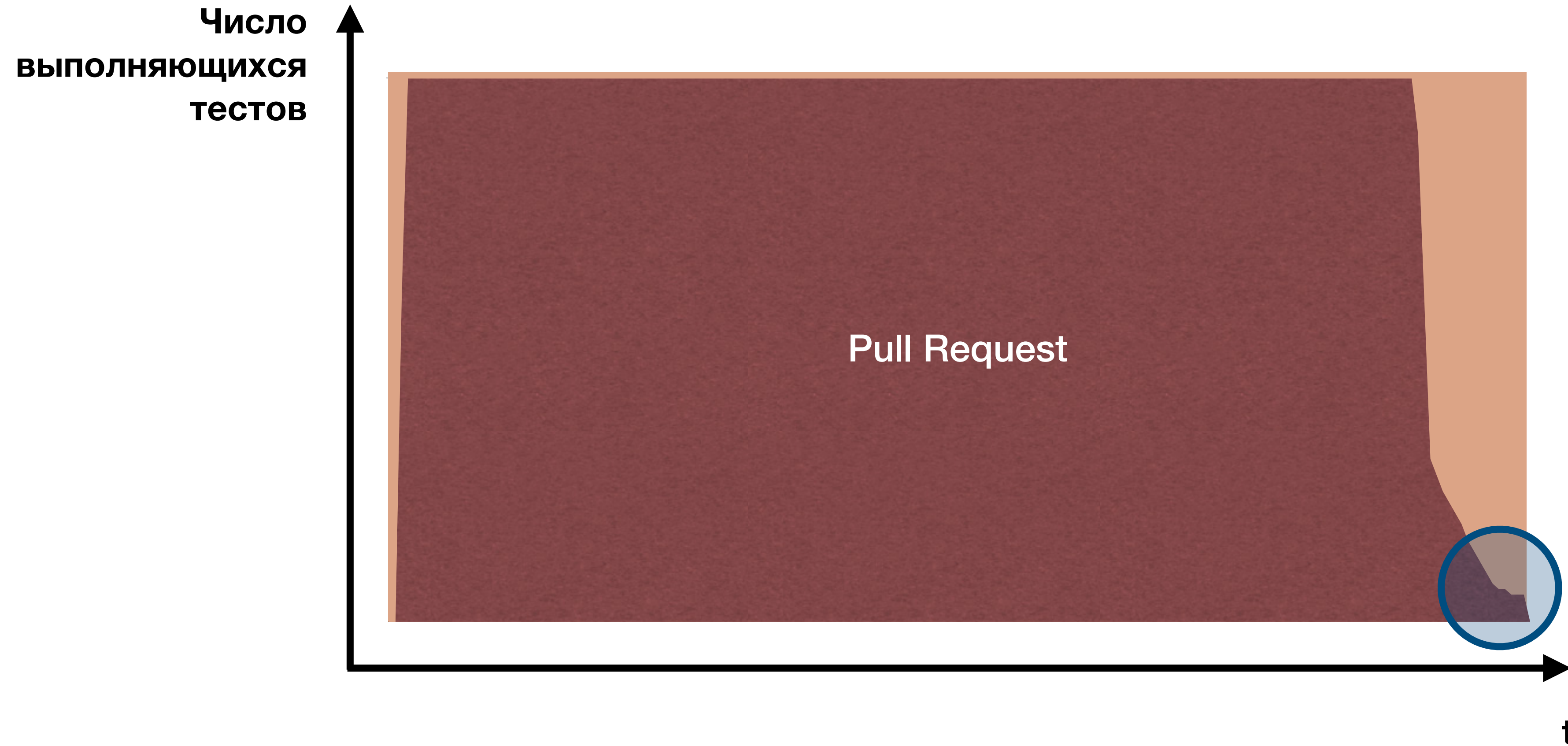
Выполнение тестов



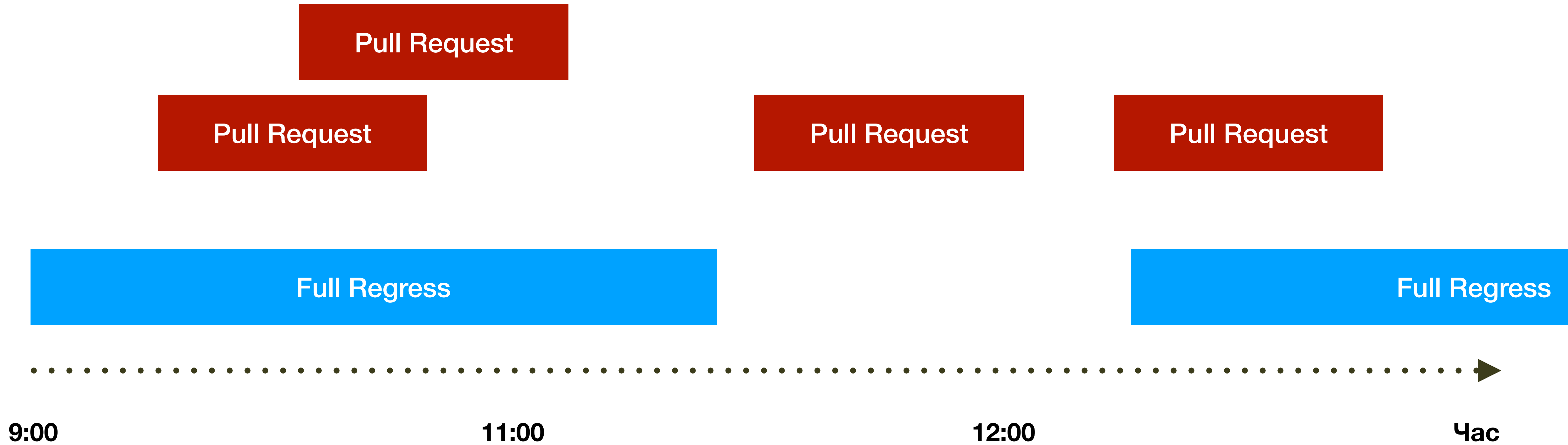
Выполнение тестов



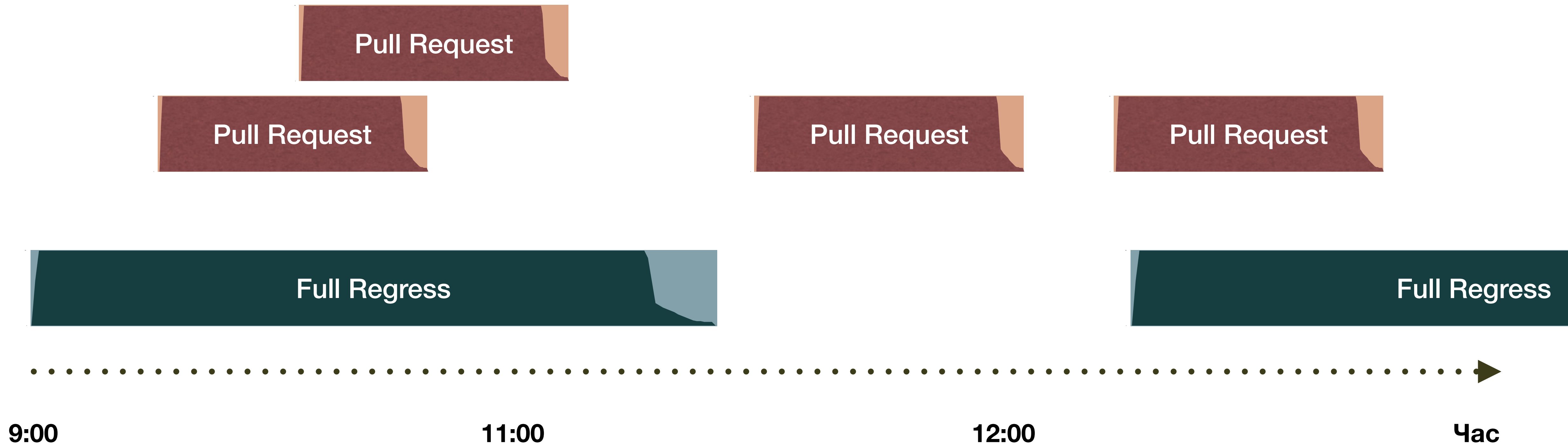
Выполнение тестов



Джобизация

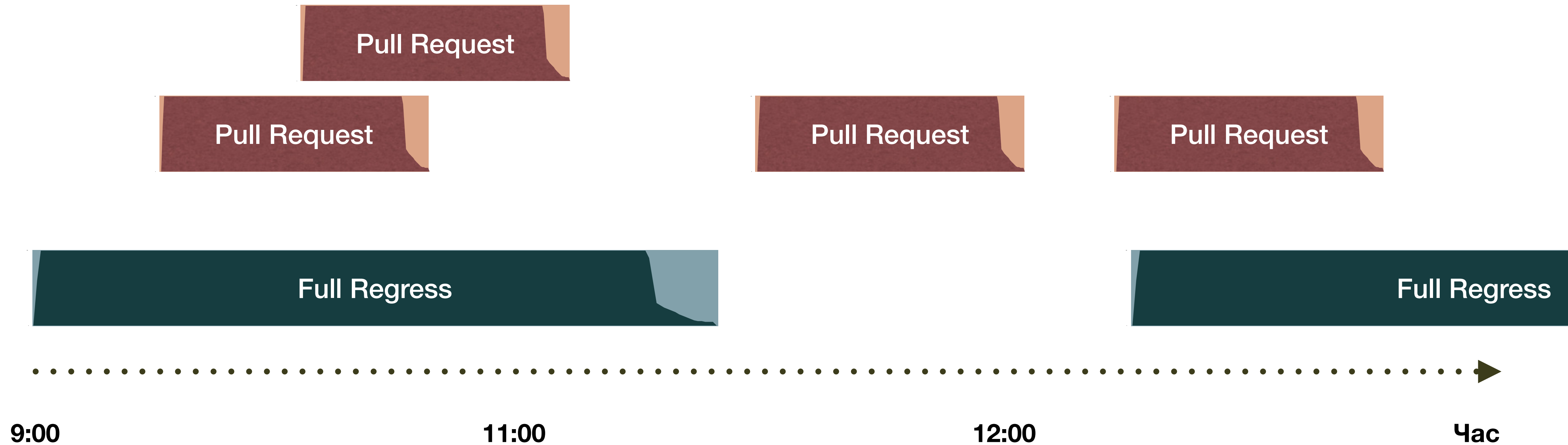


Джобизация

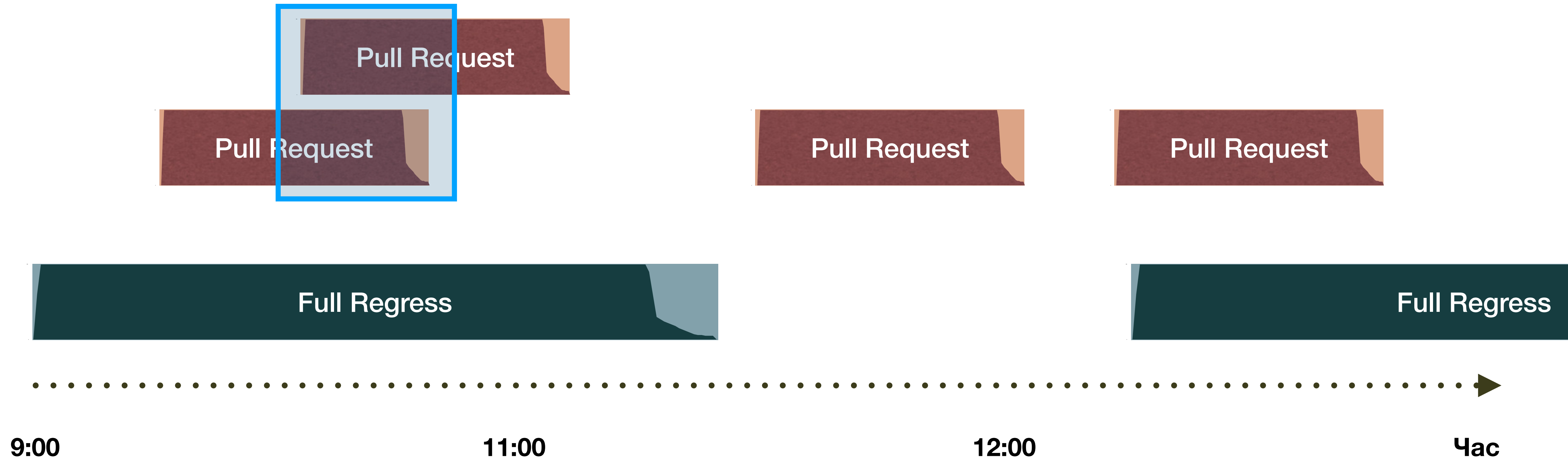


Если каждая джоба – БИЛД

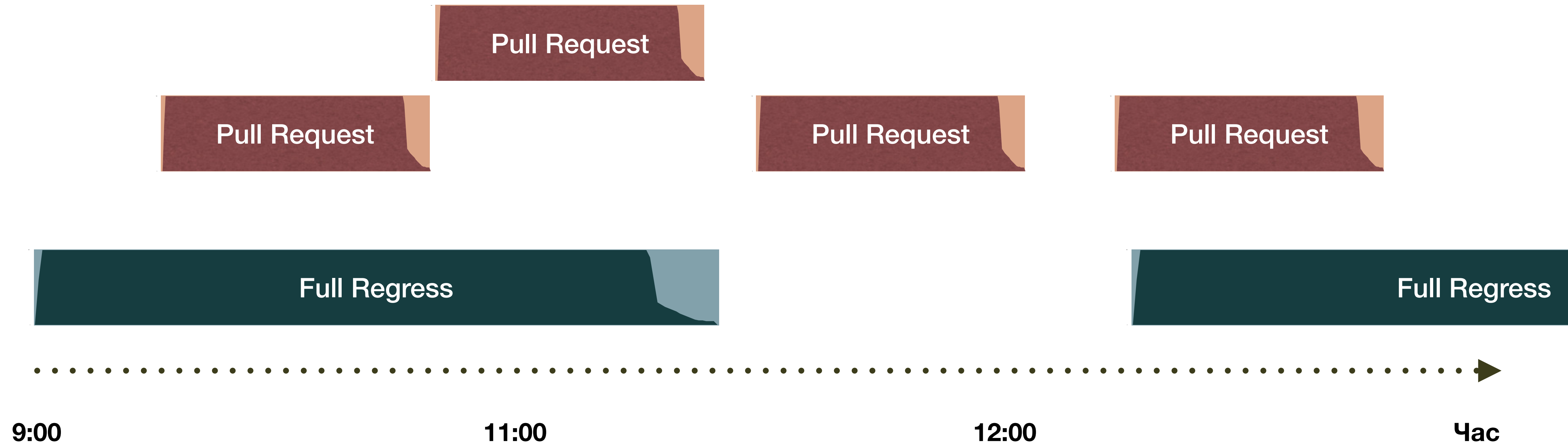
Если каждая джоба – БИЛД



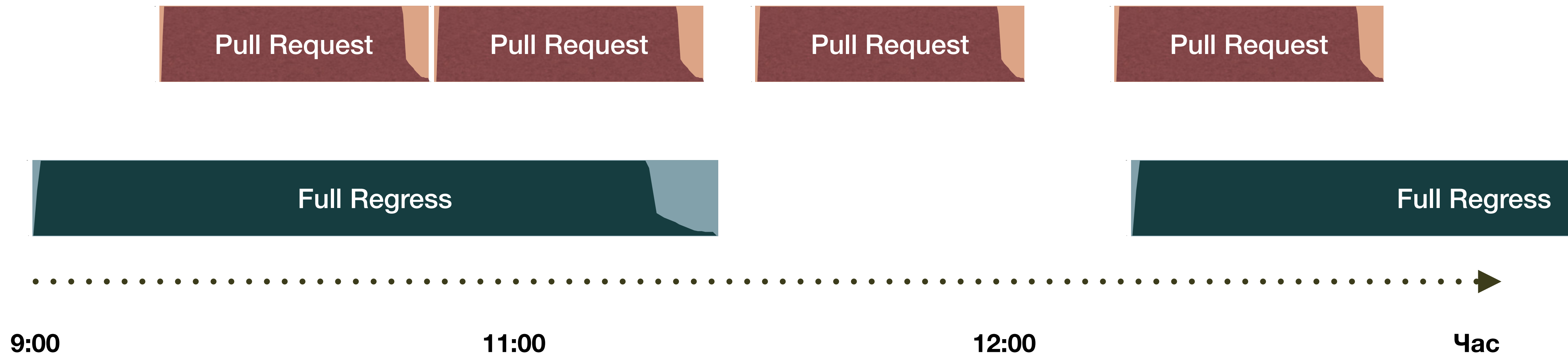
Если каждая джоба – БИЛД



Если каждая джоба – БИЛД



Если каждая джоба – БИЛД





Pull Request

Pull Request

Full Regress

9:00

Час

ТУТ ВОТ
ПРОСТОЙ ТЕХНИКИ
ДЕСЯТКИ МИНУТ

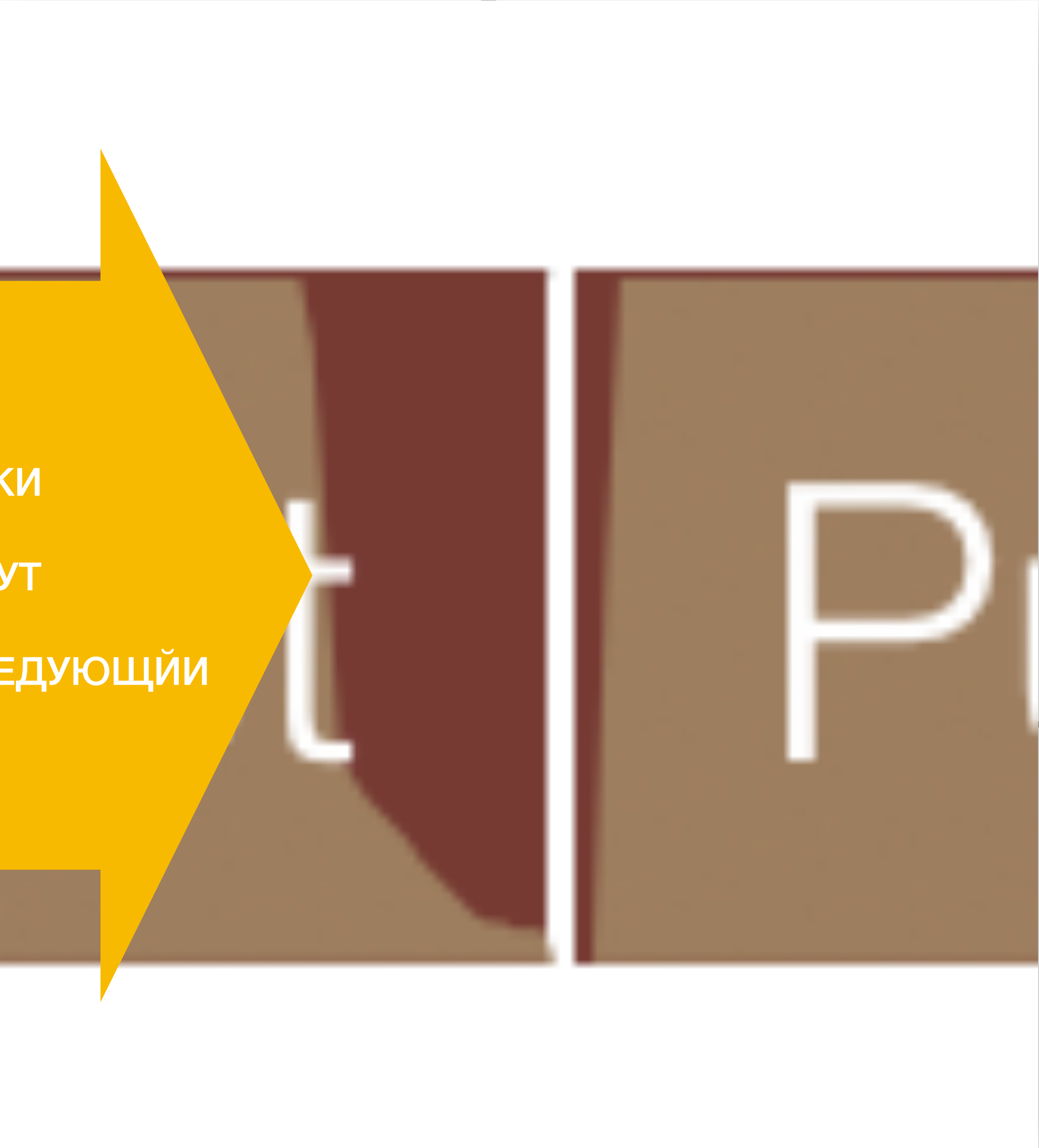
9:00

Pull Request

Full Regress

Час

ТУТ ВОТ
ПРОСТОЙ ТЕХНИКИ
ДЕСЯТКИ МИНУТ
МОЖНО СТАРТОВАТЬ СЛЕДУЮЩЕЙ
ПР!!!



Pull Request

Full Regress

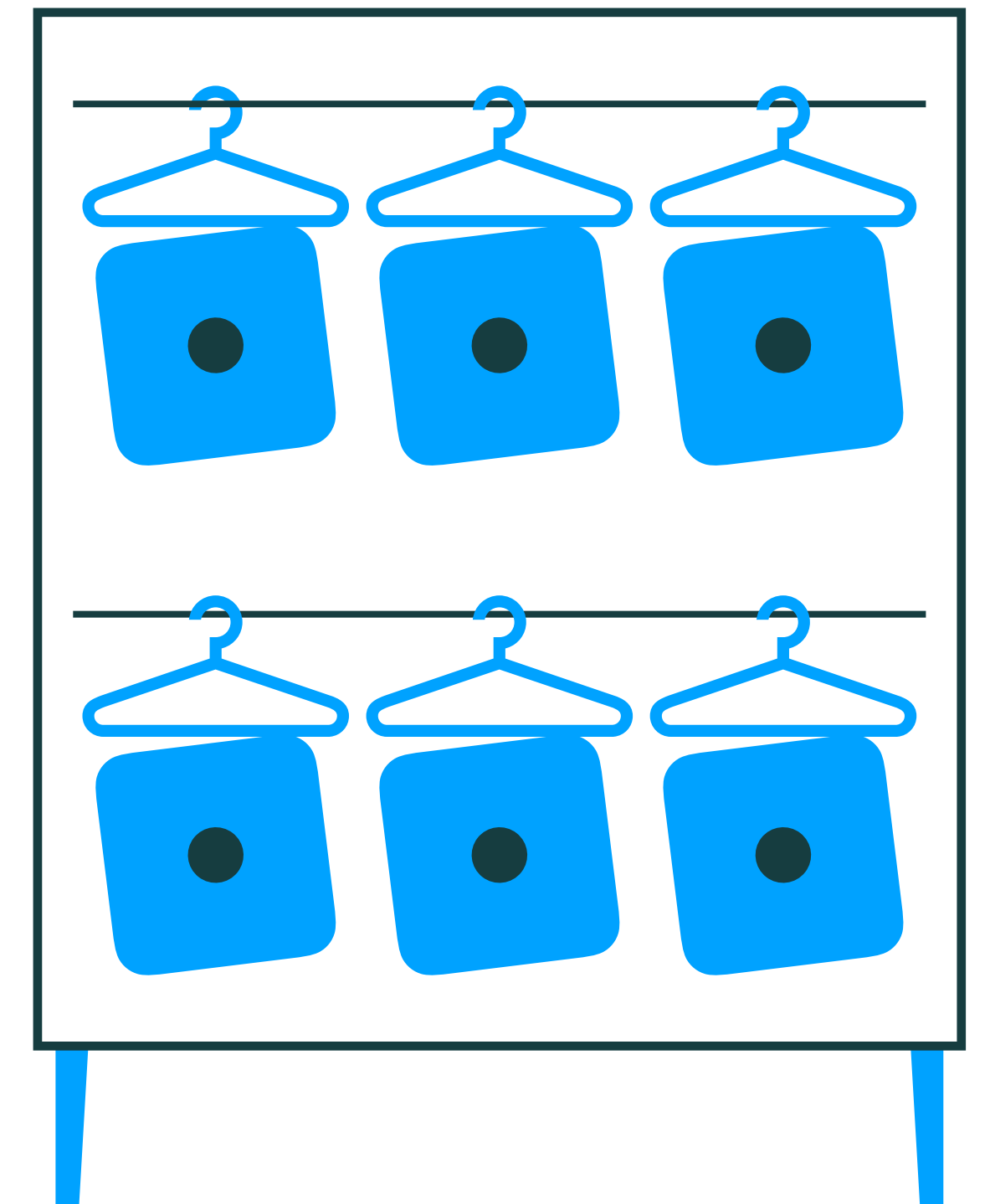


Час

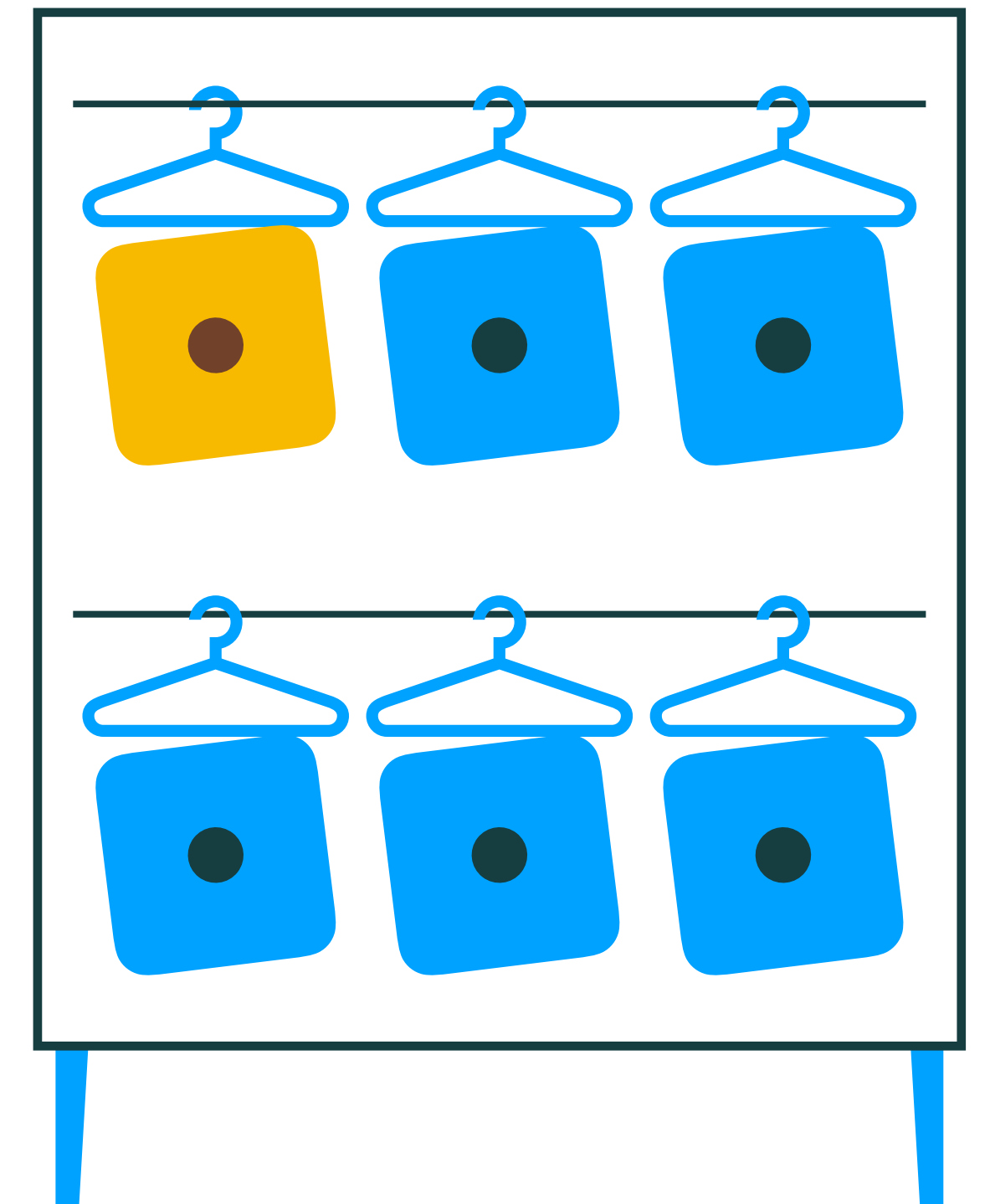
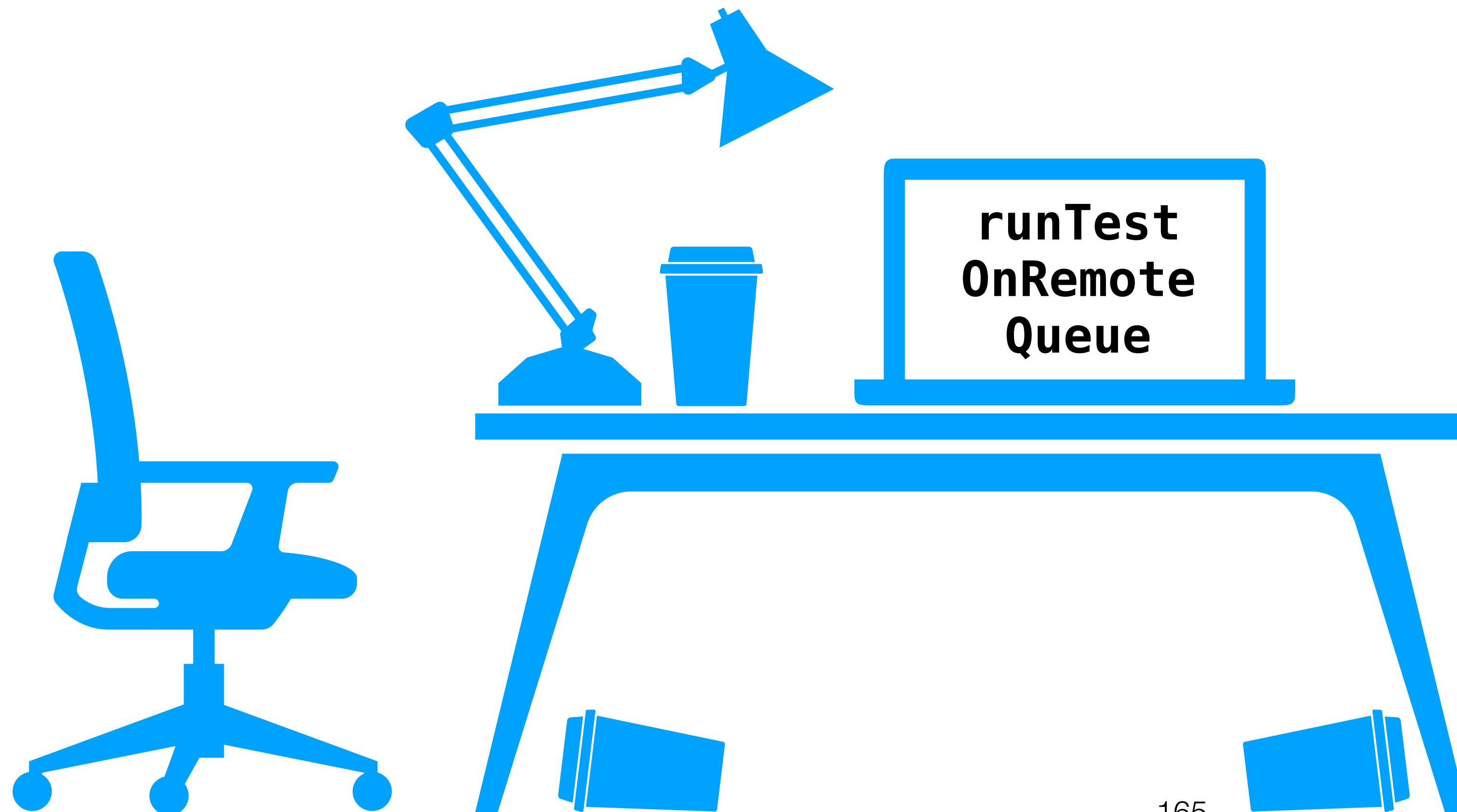
Запуск тестов распределенно

ООО «С ОБЩЕЙ ОЧЕРЕДЬЮ»

Общая очередь



Общая очередь



Общая очередь

```
AvitoRunner distRunTests \  
...
```

Общая очередь

```
AvitoRunner runTestsOnRemoteQueue \  
  ... \  
  --queue-server-destination "queue_server_destination.json"
```

Общая очередь

```
AvitoRunner runTestsOnRemoteQueue \  
... \  
--queue-server-destination "queue_server_destination.json"
```

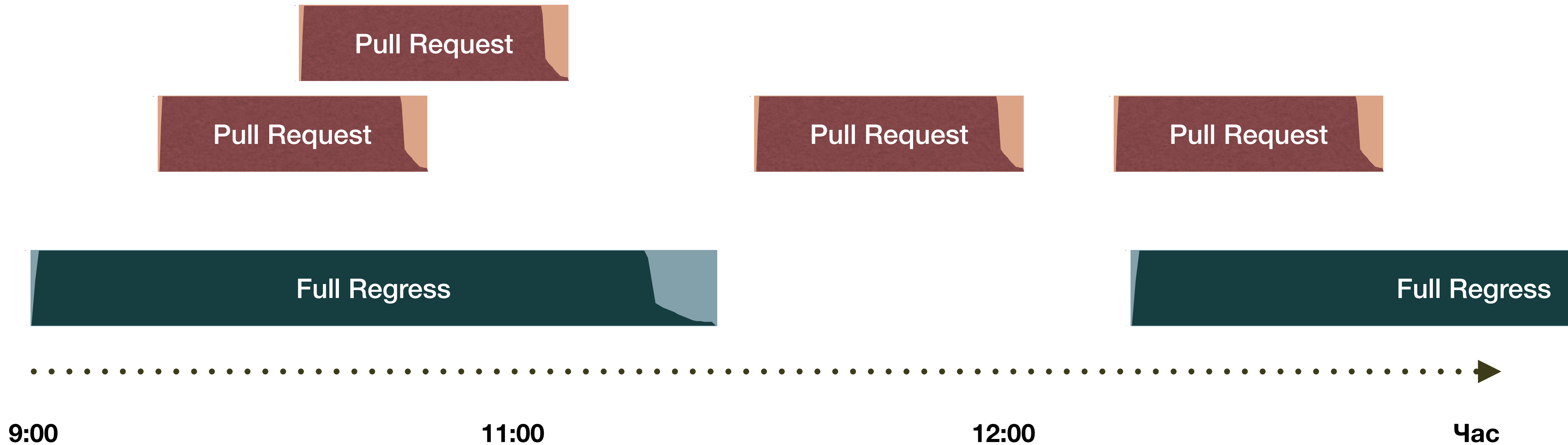
```
[  
  {  
    "host": "build-agent-macmini-01",  
    "port": 22,  
    "username": "remote_worker",  
    "password": "awesomepassword",  
    "remote_deployment_path": "/Users/emcee/remote_ui_tests"  
  }  
]
```

Возможности общей очереди

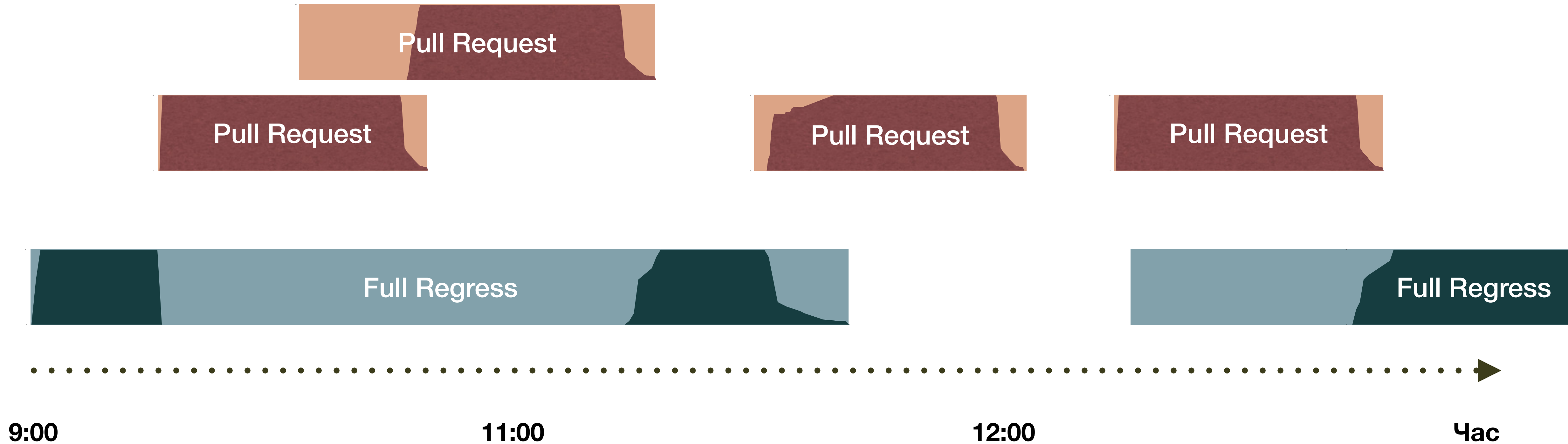
- Обслуживание нескольких джобов
 - Балансировка
 - Приоритизация
- Метрики выполнения тестов
- Лучший контроль ресурсов

Джоббы в Етсее

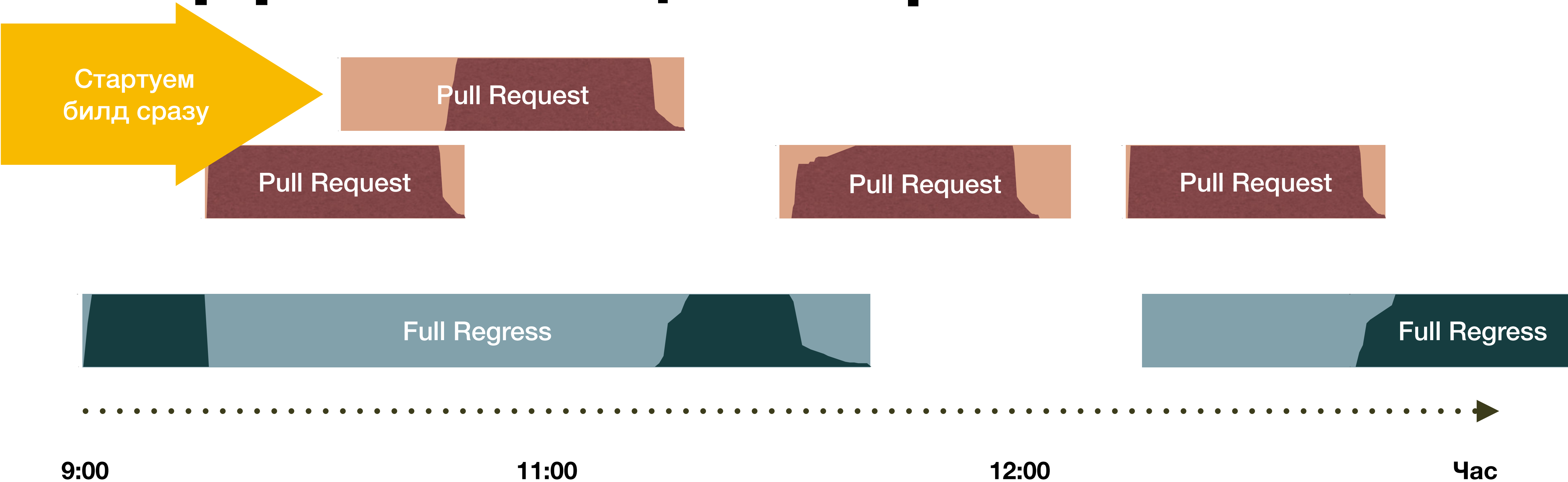
Джоббизация через Emcee



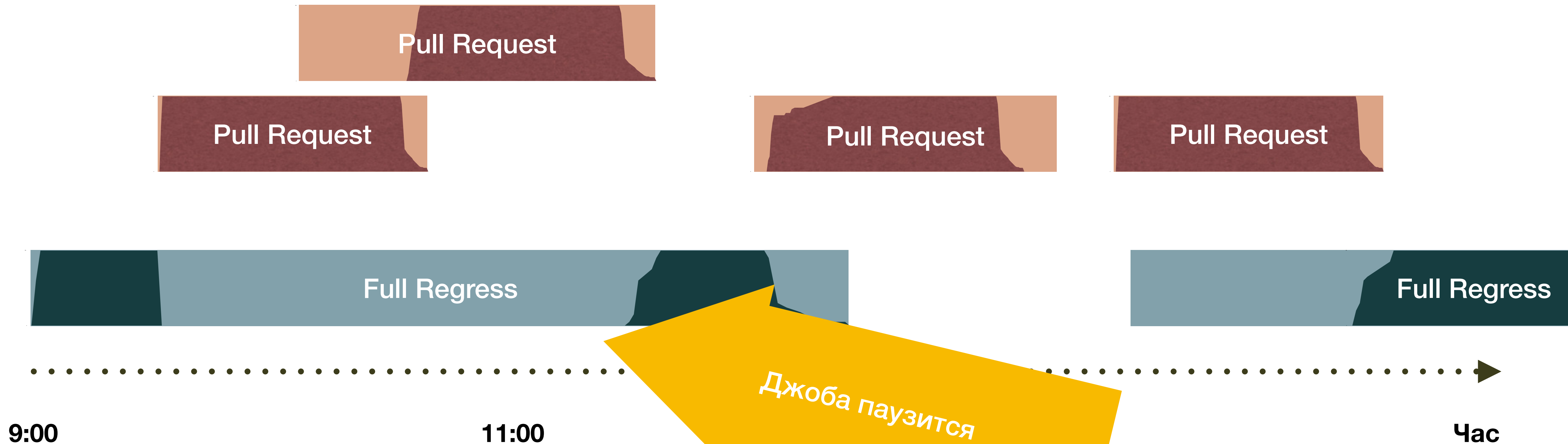
Джобизация через Emcee



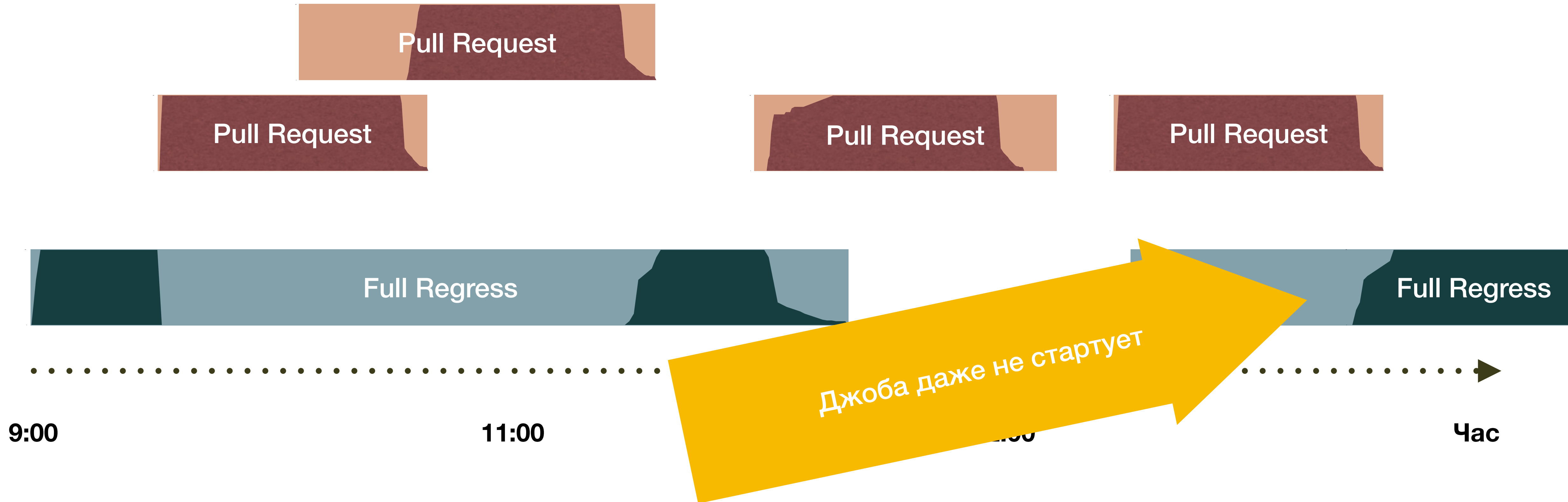
Джобизация через Етсее



Джобизация через Emcee



Джобизация через Emcee



Приоритеты

- Тесты на Pull Request-е - средний приоритет
- Full Regress - низкий приоритет

Приоритеты

- Тесты на Pull Request-е - средний приоритет
- Full Regress - низкий приоритет
- А высокий?



Все рестарты тестов на ПР
имеют высокий приоритет

Контроль над ресурсами

- Всегда N симуляторов на каждом воркере
- Тесты проходят быстрее и становятся зеленее

runTestsOnRemoteQueue vs distRunTests

(общая очередь)

(индивидуальная очередь)

- 3 билда по 220 UI тестов в каждом
- 2 симулятора на каждом воркере
- Для distRunTests ограничение в 2 конкурентных билда в ТимСити

runTestsOnRemoteQueue vs distRunTests

(общая очередь)

(индивидуальная очередь)

0 5 10 15 20 25 30 35 40 45 50 min

runTestsOnRemoteQueue vs distRunTests

(общая очередь)

(индивидуальная очередь)



Индивидуальные очереди (distRunTests)

0 5 10 15 20 25 30 35 40 45 50 min

runTestsOnRemoteQueue vs distRunTests

(общая очередь)

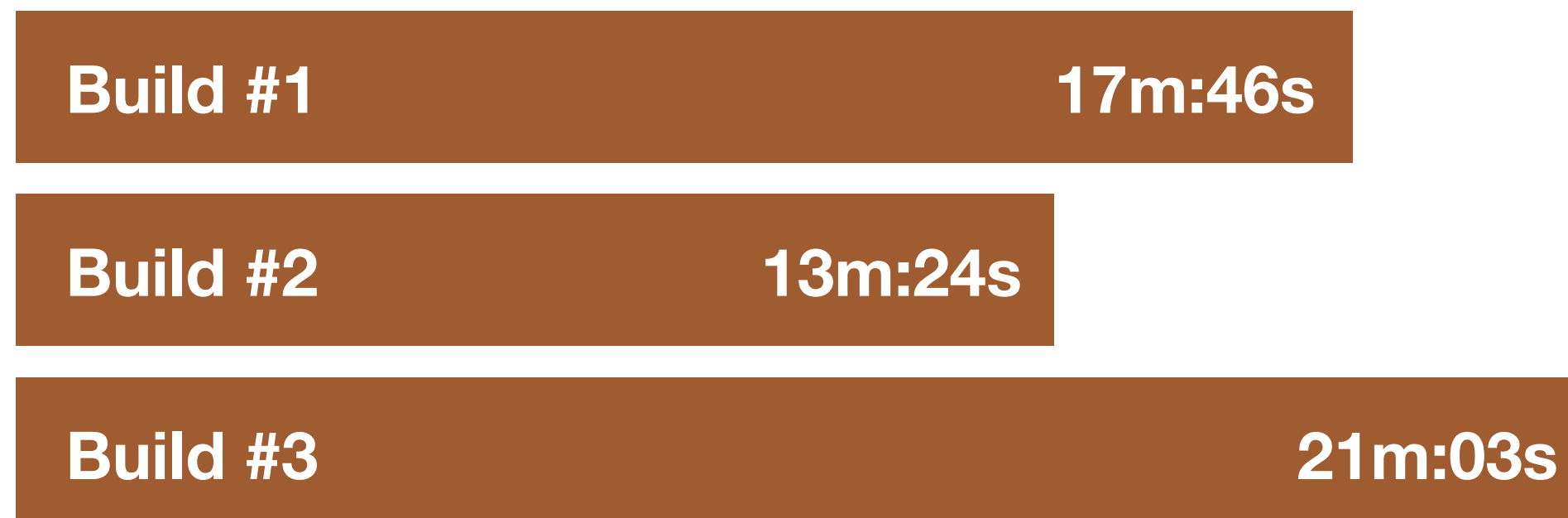
(индивидуальная очередь)



Индивидуальные очереди (distRunTests)

0 5 10 15 20 25 30 35 40 45 50 min

Общая очередь (runTestsOnRemoteQueue)



Ништяки Етсее

Инженерные подходы, которые изменили мир

Адресация артефактов

```
AvitoRunner distRunTests \  
    --app "http://example.com/MyApp.zip#files/MyApp.app"
```

- Сами организуете HTTP хостинг (Artifactory)

Адресация артефактов

```
AvitoRunner distRunTests \  
  --app "http://example.com/MyApp.zip#files/MyApp.app"
```

- Сами организуете HTTP хостинг (Artifactory)

Адресация артефактов

```
AvitoRunner distRunTests \  
  --app "http://example.com/MyApp.zip#files/MyApp.app"
```

- Сами организуете HTTP хостинг (Artifactory)

Пер-воркер кэш



Изи демонизация

- Запуск демона очереди с любого билд агента
- Очередь умирает сама при долгом простое

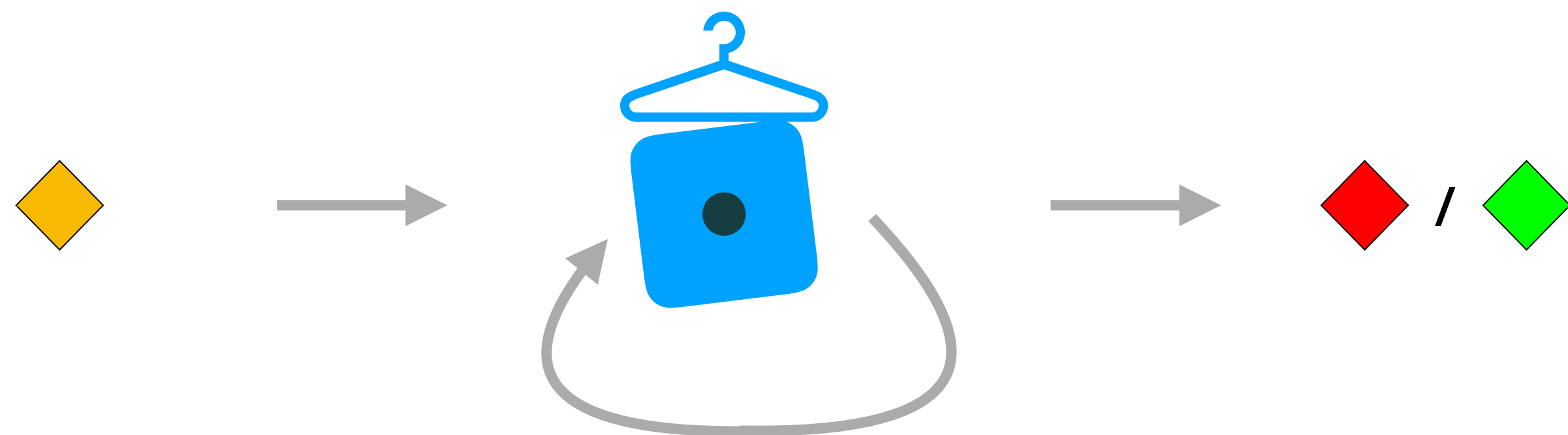
Изи демонизация

- Запуск демона очереди с любого билд агента
- Очередь умирает сама при долгом простое
- Ничего не персистентно (кроме кэша)

Версионирование — как хотите так и делайте

- Можно сбилдить
- `$REPO_ROOT/.emceeverision` `43ba53cfe`
- `http://artifactory/emcee/emcee-43ba53cfe.zip`
- Обновление через ПР

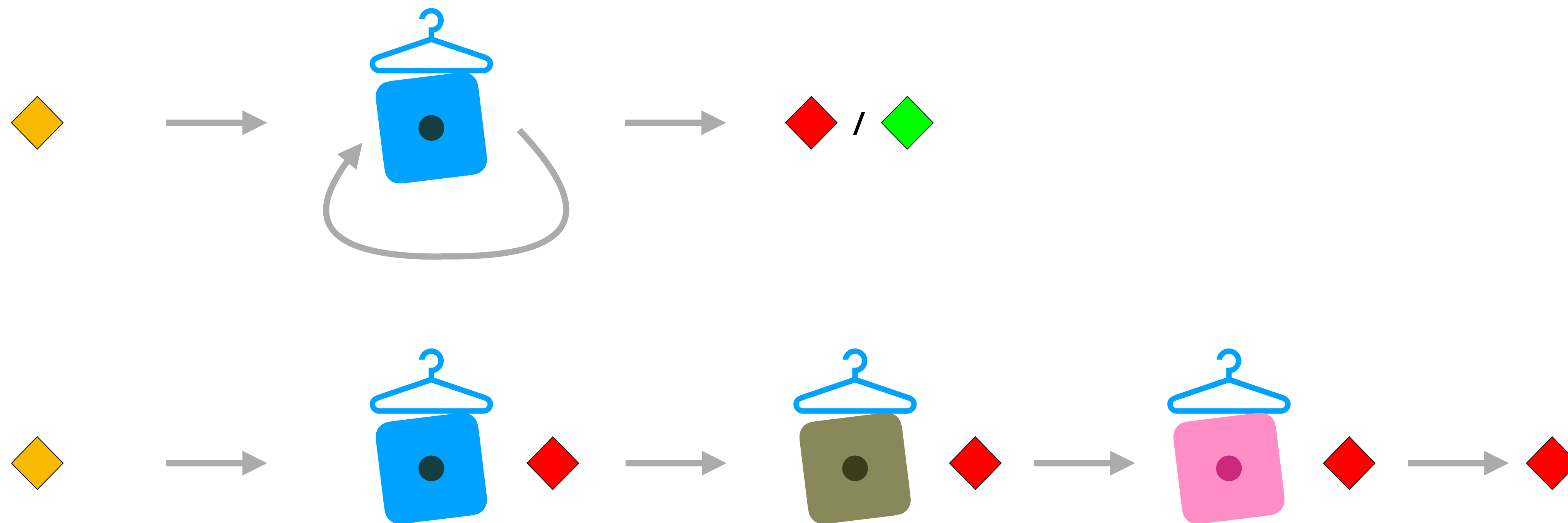
Worker-aware перезапускки тестов



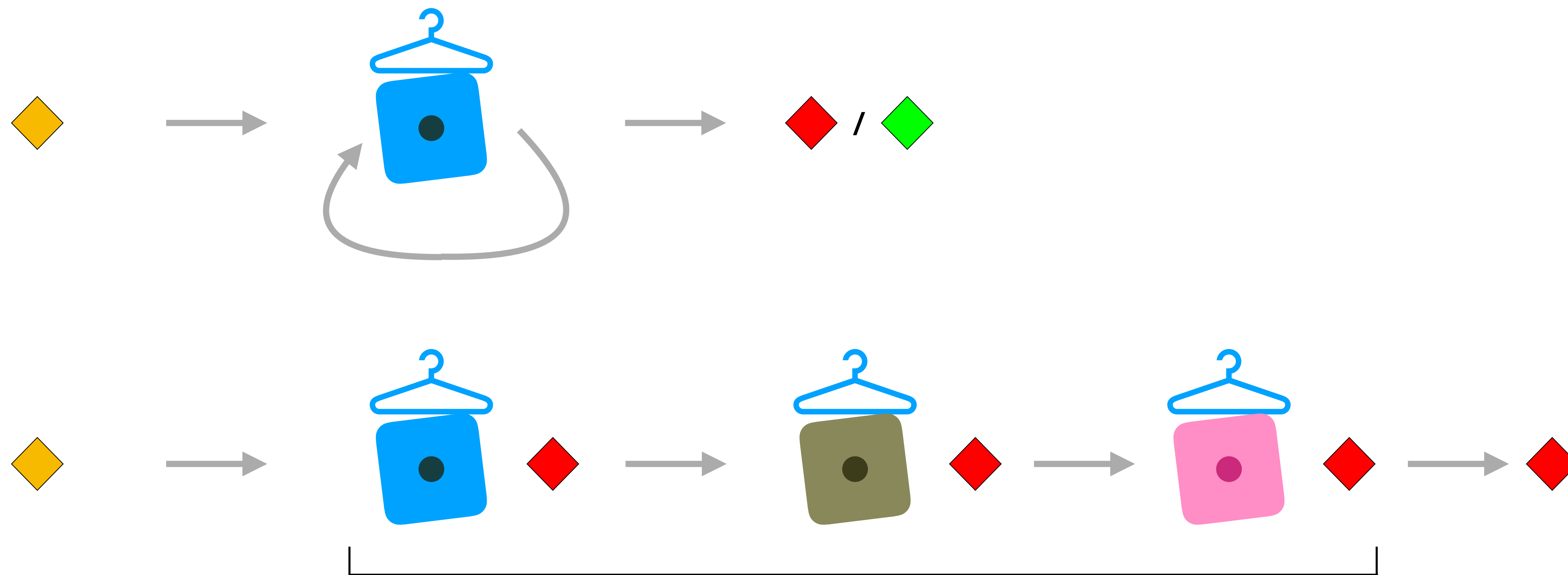
Worker-aware перезапускки тестов



Worker-aware перезапуска тестов



Worker-aware перезапуски тестов



N прыжков по воркерам

Runtime Dump

- Неслепой, умный, гибкий листинг тестов из `.xcstest`
- Валидация наличия тестов
- Тегирование тестов в коде
- Что угодно, то и делай

Список тестов в xctest банлде

```
[
  {
    "path" : ".../UITests/MONP/MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions"
  },
  {
    "path" : ".../UITests/Perf/PER_Logging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "PER_Logging"
  },
]
```

Список тестов в xctest банлде

```
[
  {
    "path" : "...\\UITests\\MONP\\MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions",
    "caseId" : 1121,
    "tags" : [
      "regression"
    ]
  },
  {
    "path" : "...\\UITests\\Perf\\PER_Logging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "PER_Logging",
    "caseId" : 3331,
    "tags" : [
      "performance"
    ]
  },
]
```

Список тестов в xctest банлде

```
[
  {
    "path" : ".../UITests/MONP/MONP_Extensions.swift",
    "testMethods" : [
      "test_extension1",
      "test_extension2"
    ],
    "className" : "MONP_Extensions",
    "caseId" : 1121,
    "tags" : [
      "regression"
    ]
  },
  {
    "path" : ".../UITests/Perf/PER_Logging.swift",
    "testMethods" : [
      "test"
    ],
    "className" : "PER_Logging",
    "caseId" : 3331,
    "tags" : [
      "performance"
    ]
  },
]
```

Плагины

- Отдельные процессы
- Работают на воркерах и на очереди
- Общаются через JSON stream / web socket
- Swift Package Manager модули

Мониторинг

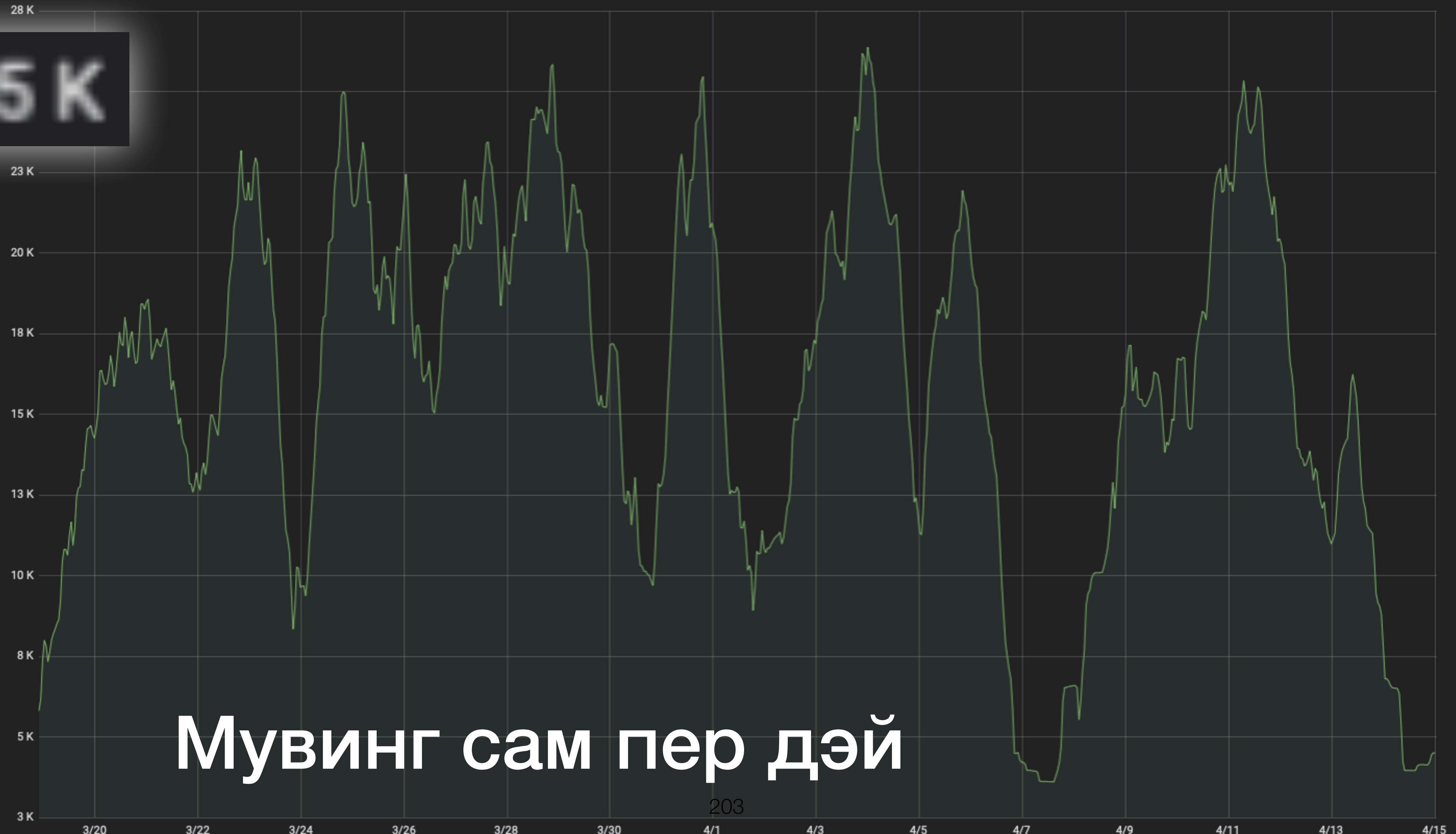
- Graphite
 - События начала/окончания тестов
 - История тестов
 - Длительность тестов
- Sentry
 - Ошибки и ворнинги

Мониторинг

- Graphite `$EMCEE_GRAPHITE_PREFIX, $EMCEE_GRAPHITE_SOCKET`
 - События начала/окончания тестов
 - История тестов
 - Длительность тестов
- Sentry `$EMCEE_SENTRY_DSN`
 - Ошибки и ворнинги

Число тестов за 24 часа

25 K



Мувинг сам пер дэй

Резюмировать

- Распределенные системы - это интересно и сложно
- Emcee позволил нам упереться в железо
- <https://github.com/avito-tech/Emcee>

Спасибо за внимание!

Владислав Алексеев

- beefon@me.com

- @beefon в Твиттере и Телеге

Еще очень классные доклады:

Нативные UI тесты, которые работают – Артём Разинов, Авито

<https://www.youtube.com/watch?v=036qVHFtNxc>

iOS UI тесты в Авито — Артем Разинов, Авито

<https://www.youtube.com/watch?v=xqRfyt1SEZQ>

Инфраструктура UI-тестов в Авито — я

<https://www.youtube.com/watch?v=aBcJI8ObSeU>

