



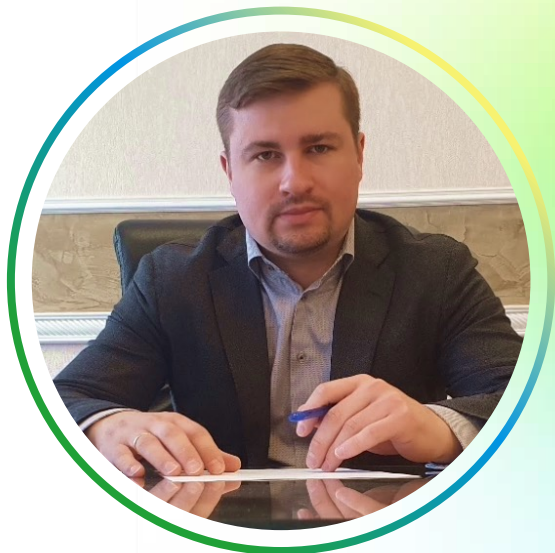
# Как с помощью BPMS (jBPM) заменить продукты SAS?

Олег Терёшкин

Ноябрь, 2022



# Обо мне



Олег Терёшкин

IT-лидер кластера

## Опыт

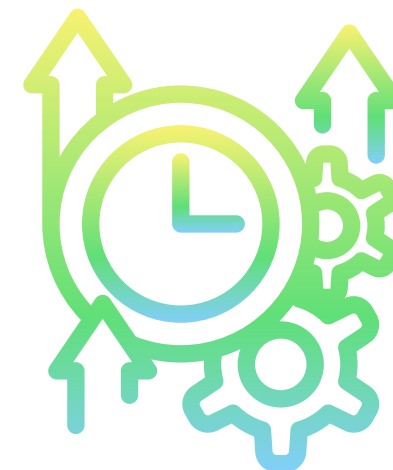
- Низкоуровневая разработка (драйверы)
- Data Engineering
- Разработка микросервисов
- Solution-архитектура

## Образование

Мехмат МГУ



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ  
УНИВЕРСИТЕТ ИМЕНИ М.В. ЛОМОНОСОВА



# План доклада

1 

**Несколько слов о продукте**

2 

**Проблематика и мотивация**

3 

**Демо создания процесса**

4 

**Общая архитектура решения**

5 

**Результаты нагрузочного тестирования**

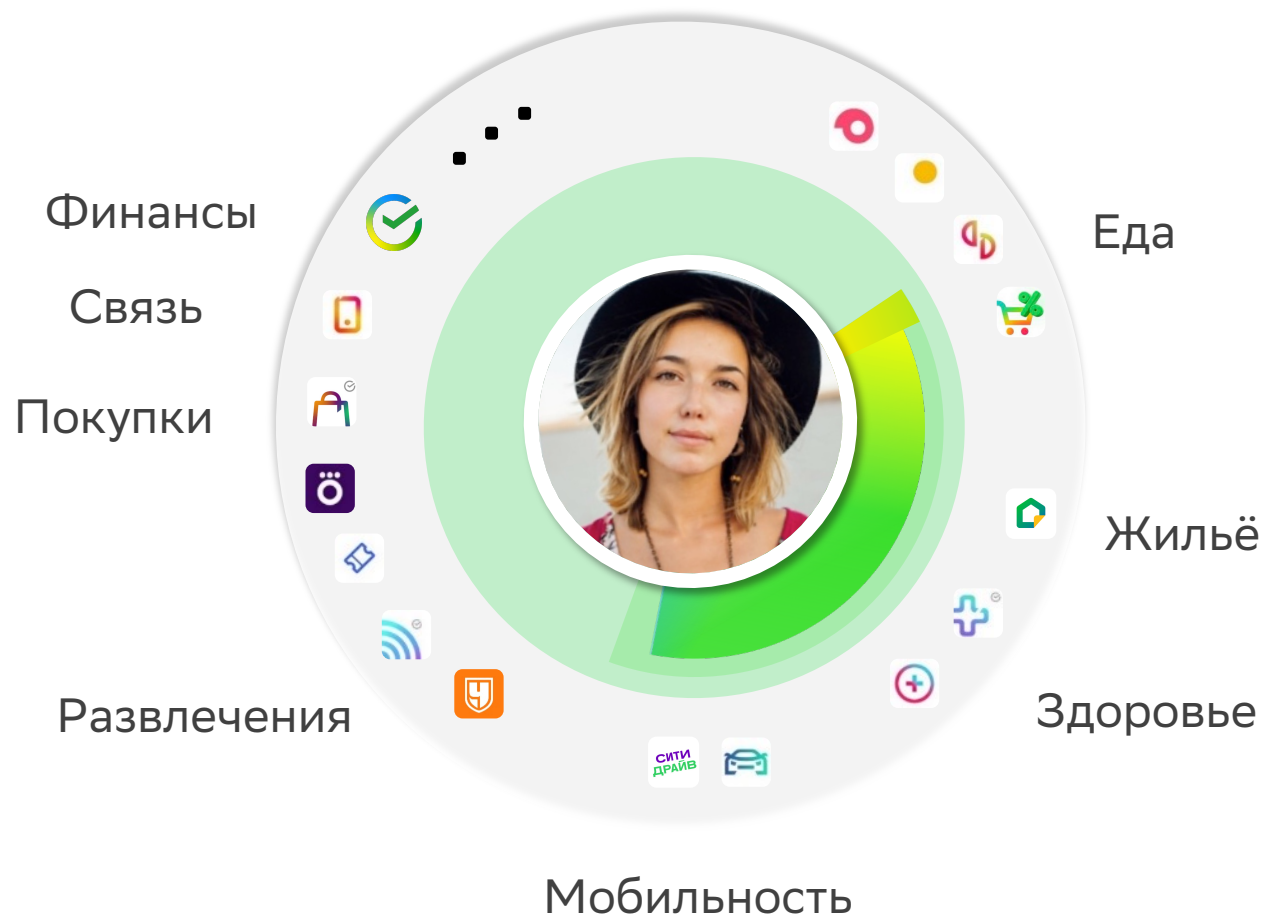
6 

**Разбор проблем и варианты решений**

7 

**Вопросы**

# Платформа персонализированных предложений для клиентов СБЕРа



**1+ млрд**

персональных  
предложений

**106 млн**

клиентов

**10+ тыс.**

рекламных кампаний  
в месяц

**90+**

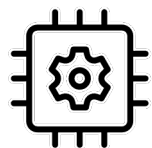
компаний партнёров



# Выручка SAS за 2021 – 3,5 млрд рублей

5

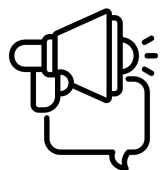
## Основные направления деятельности



Платформа  
исполнения  
моделей



Скоринг



Автоматизация  
маркетинговых  
кампаний

## Основные заказчики



## Наиболее известные продукты



SAS Viya™

**sas** MA  
RTDM

MO

EG

...

## Типовые сценарии, которые решали с помощью SAS

6



### Событие

1

Клиент забыл карту  
в банкомате

2

Клиент оформил  
кредитную карту

3

Клиент не закончил  
заполнение анкеты  
на сайте



### Целевое действие

Сообщить IVR  
о приоритетном ролике

Отозвать действующий  
оффер на кредитную  
карту

Отправить СМС  
с напоминанием о заявке



ушёл с российского рынка. Начинаем задачу миграции <sup>7</sup>

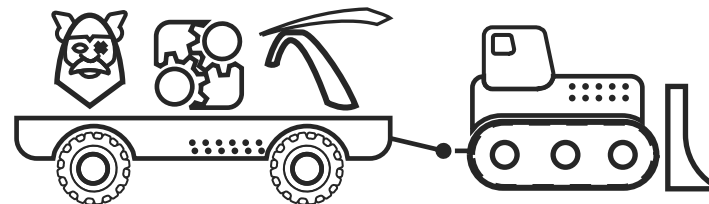


Компания SAS  
покинула  
российский рынок  
в марте 2022

СБЕР остался  
без экспертной  
поддержки  
внедренных  
решений (RTDM,  
Viya, DI, EG, ...)

Экспертная  
команда  
разработки от SAS  
прекратила свое  
существование

Срок окончания  
лицензии по SAS  
RTDM – ноябрь  
2022



Уход  
от vendor  
lock-in  
(SAS2NULL)

# Основные требования к движку

8

- 1 Open-source (Java Stack)
- 2 Low-code/no-code для настройки бизнес-правил (DMN, Drools, Data objects)
- 3 Highload: горизонтально масштабируемое решение под нагрузки ~50 000 TPS
- 4 Интеграции из коробки: Kafka, REST API

**20+ решений на рынке**

Вендоры:



Open-source:



...

## Продукт глазами пользователя



# Артефакты, которые исполняет jBPM

10

## Результат билда – 2 артефакта

Demo.pom  
Demo.jar

M2 Repository Content

Upload   

Name	GAV	▼ Last Modified	Open	Download
Demo-1.0.0-20221116.144028-16.pom	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 16 17:40:28	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221116.144028-16.jar	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 16 17:40:28	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221116.143837-15.jar	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 16 17:38:37	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221116.143837-15.pom	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 16 17:38:37	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.173010-14.jar	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:30:10	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.173010-14.pom	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:30:10	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.172622-13.jar	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:26:23	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.172622-13.pom	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:26:23	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.172259-12.jar	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:22:59	<a href="#">Open</a>	<a href="#">Download</a>
Demo-1.0.0-20221109.172259-12.pom	com.sberconf:Demo:1.0.0-SNAPSHOT	2022 Nov 9 20:22:59	<a href="#">Open</a>	<a href="#">Download</a>

# Структура JAR

XML-схема процесса и low-code ----->

наши классы входящего  
и исходящего сообщений ----->

конфигурации исполнения  
процесса (persistence, audit) ----->

## Структура Demo.jar

 Demo.Main-svg.svg

 Main.bpmn

 TestDMN.dmn

 gRule.rdrl


▼  com

 InputMessage.class


 OutputMsg.class

▼  META-INF

 kie-deployment-descriptor.xml

 kmodule.info

 kmodule.xml

▼  defaultKieBase

 kbase.cache

▼  maven

▼  com.sberconf

▼  Demo

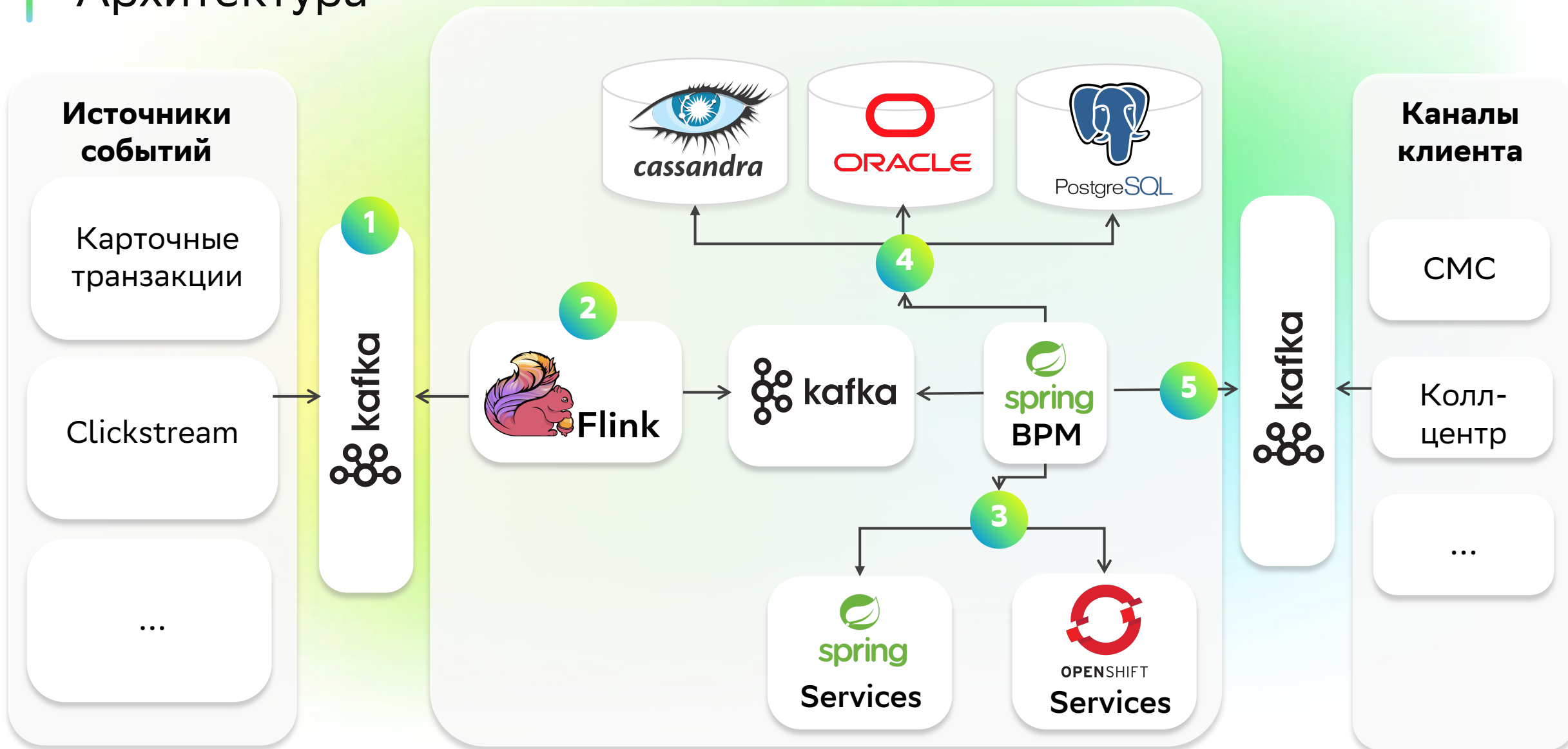
 pom.xml

 pom.properties



# Архитектура

12



## Характеристики железа для нагрузочного тестирования

13

	Кол-во серверов	ЦПУ, кол-во ядер	ОЗУ, ГБ	Диск, ГБ
 kafka	5	4	16	500
 jBPM	11	4	16	100
 Flink	16	4	32	1024

# jBPM — методика проведения нагрузочного тестирования

14

## Поиск максимума:

**1 ч. 10 мин.**

длительность  
теста

**2 000 TPS**

начальный  
уровень  
нагрузки  
на систему

**2 000 TPS**

шаг каждые  
15 минут

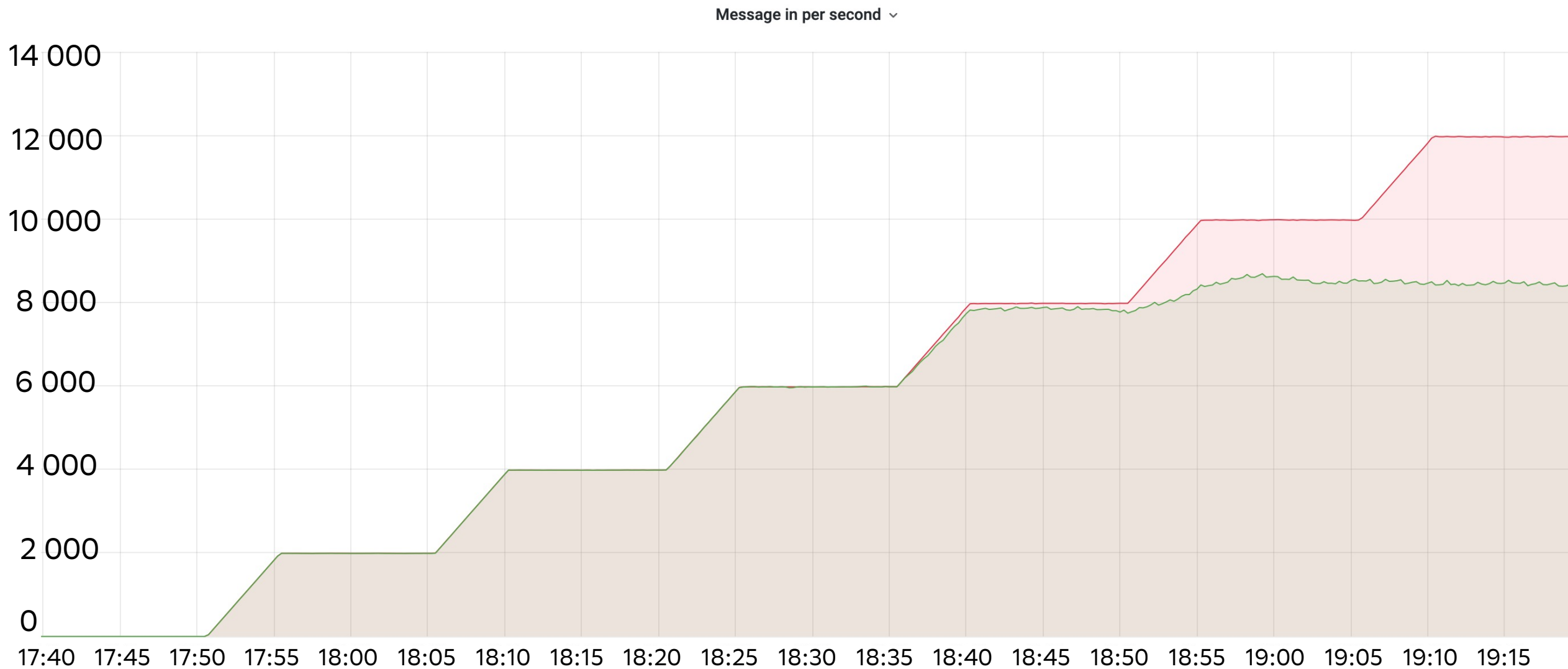
**12 000 TPS**

уровень нагрузки  
на последней  
ступени теста

# 6000 TPS — максимальная зафиксированная производительность

15

**Красная** — входящая нагрузка **Зелёная** — скорость обработки jBPM



# jBPM — методика проведения нагрузочного тестирования

16

## Тест надёжности:

**8 ч.**

длительность  
теста

**120 млн**

общее кол-во  
сообщений

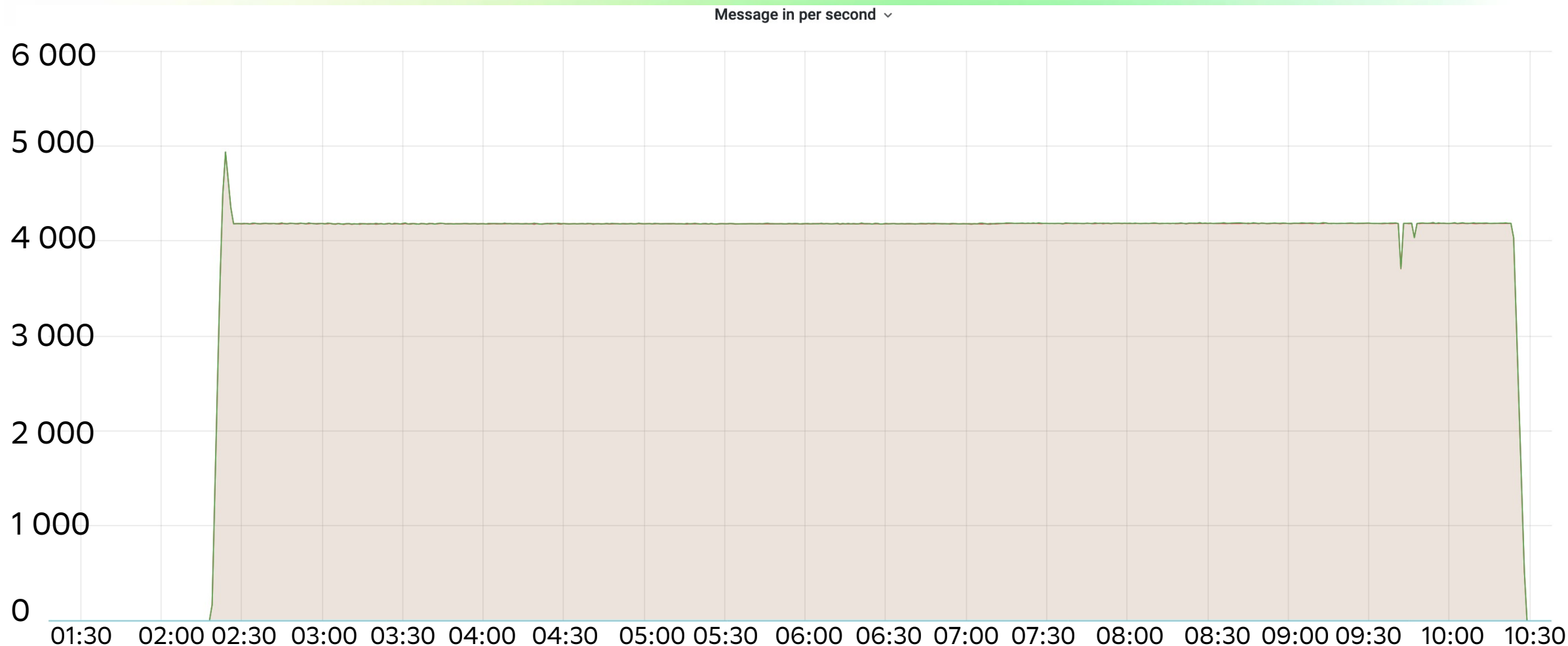
**4 200 TPS**

надежность системы  
на уровне  
производительности  
подтверждена

# 4200 TPS – тест надёжности

**Красная** – входящая нагрузка    **Зелёная** – скорость обработки jBPM

17



sbernba-bpm-log

ikp-output-batch-jBPM

sbernba-esb-errors

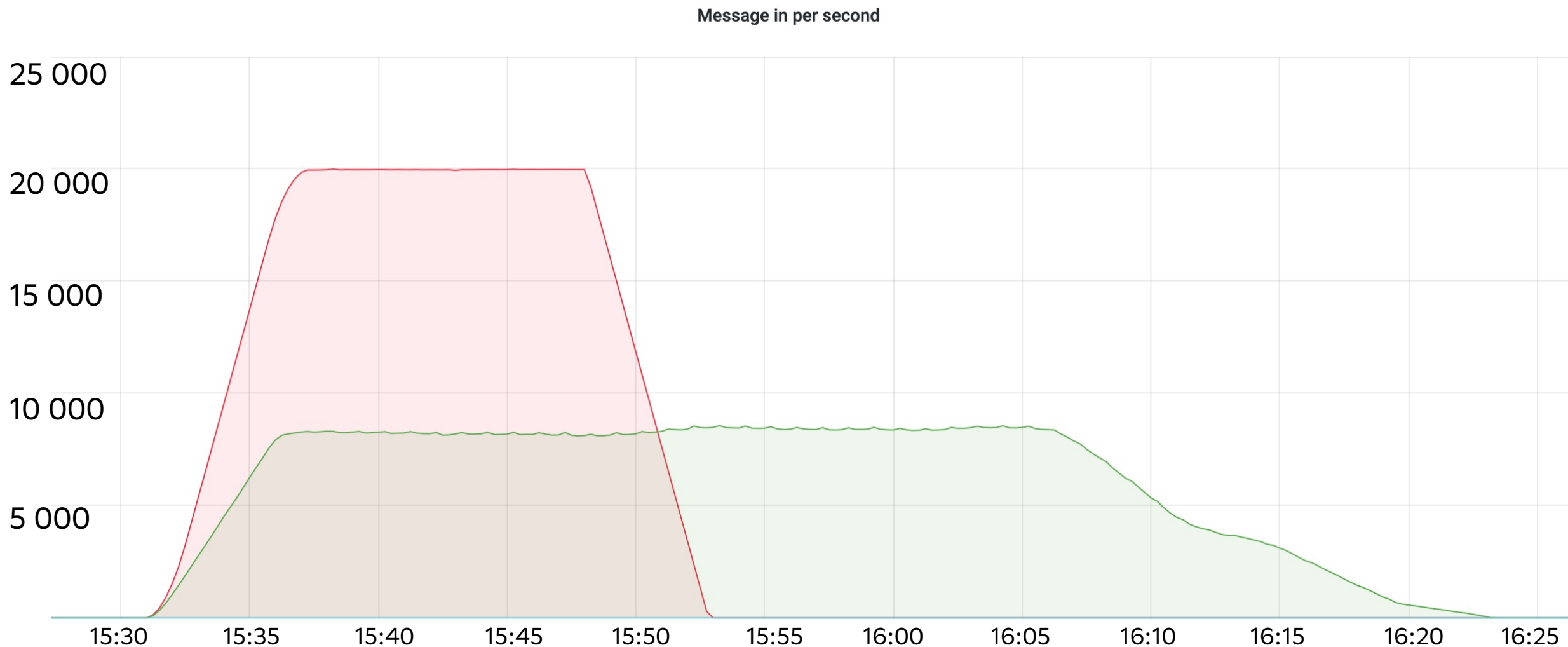
max	avg ▾	current	total
4.94 K	3.62 K	0	2.03 Mil
4.93 K	3.61 K	0	2.03 Mil
0.0246	0.000418	0	0.235



# Потеря сообщений в случае накопления лага

18

**Красная** – входящая нагрузка    **Зелёная** – скорость обработки jBPM





# Шаги в сторону Highload

19

1

## Хранить в памяти сервисную базу данных движка

```
1  # service data source configuration
2  spring.datasource.username=sa
3  spring.datasource.password=sa
4  spring.datasource.url=jdbc:h2:mem:spring-boot-jbpm
5  spring.datasource.driver-class-name=org.h2.Driver
```

2

## Крон по удалению данных из таблиц — не переполнить RAM и не потерять зависшие процессы

```
8  clean.db.jpa.enable=true
9  clean.db.sql.enable=true
10 clean.db.rate=60000
11 clean.db.delay=60000
12 clean.db.sql.requestInfo=DELETE FROM REQUESTINFO where timestamp < systimestamp - interval '5' minute
13 | | | | | | | AND (status <> 'QUEUED' AND status <> 'RETRYING' AND status <> 'RUNNING' )
14 clean.db.sql.errorInfo=DELETE FROM ERRORINFO where timestamp < systimestamp - interval '5' minute
15 clean.db.jpa.period=1m
```

# Шаги в сторону Highload

20

3

## Потеря процессов в случае накопления лага

A Использование  
**Retry**

B **RetryCount**  
= 5,10 ...

C **RetryDelay** =  
...

D Использование  
топики для ошибок  
и отдельного  
процесса для  
зацикливания  
(только осторожно)



## 4

### Исполнение SQL-запросов с использованием bind'ов

```
18 statement.executeQuery("SELECT * FROM s_user where userid = " + i);  
19  
20 pstatement = conn.prepareStatement("SELECT * FROM s_user where userid = ?");  
21 pstatement.setInt(1, i);  
22 pstatement.executeQuery();
```

## Вместо заключения

22

- 1 BPMS jBPM, и не только jBPM, можно использовать под нагрузками, и не только для human-процессов
- 2 Выбирайте решение максимально закрывающее ваши потребности и дорабатывайте его под себя
- 3 Задавайте вопросы в открытое комьюнити, с вашей проблемой/ошибкой уже кто-то сталкивался
- 4 Не пытайтесь перенести весь код на BPMS

Олег Терёшкин  
tg: @o\_terjeshkin