

Yandex



# Streaming event matching

Dmitry Schitinin

# About

<https://stat.yandex.ru>

# About

Backend infrastructure team

- › Services
- › Libraries
- › Frameworks

# About

Backend infrastructure team

- › Services
- › Libraries
- › Frameworks

<https://stat.yandex.ru>

@ Yandex.Classifieds

- › auto.ru
- › auto.yandex.ru
- › rabota.yandex.ru
- › realty.yandex.ru
- › travel.yandex.ru

# Introduction

# Audience

- › Software engineers/architects
- › Event stream processing
- › Distributed systems



## Chevrolet Camaro V

1 310 000 Р

2010

3.6 АТ (312 л.с.) бензин, задний привод,  
купе, красный

Москва, 12 часов

### Chevrolet Camaro в Москве и Московской области

Таможня: Растаможен

Только с фото

Состояние: Кроме битых

Получать письма каждый час

✓ Поиск сохранён

Удалить



## Chevrolet Ca

6.2 АТ (405 л.с.)  
купе, жёлтый

Москва

012

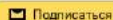


# 2 688 объявлений

Сортировать: по релевантности ▾

за всё время ▾

таблицей



Реклама

 [Автомобили Chevrolet на Авто.ру](#)

Авто с пробегом [Новые авто](#) [Добавить авто](#) [Расширенный поиск](#)  
Частные объявления! Более 450 000 объявлений о продаже машин на Авто.ру.  
[auto.ru](#)



2012

[Chevrolet Aveo](#) ✓

16 000 км, седан, 1.6 MT (115 л.с.), бензин, передний привод

440 000 ₽



13 часов назад

Санкт-Петербург

[AUTO.RU](#)



2013

[Kia Rio](#) ✓

42 500 км, седан, 1.6 AT (123 л.с.), бензин, передний привод

510 000 ₽



9 июля

Санкт-Петербург

[AUTO.RU](#)



2012

[LADA \(VA3\) 2107](#)

30 км, седан, 1.6 MT (74 л.с.), бензин, задний привод

185 000 ₽



10 часов назад

Санкт-Петербург

ПАРАМЕТРЫ

новые  с пробегом  битые

цена от 1 000 000 ₽ ▾

в наличии  с фото  
 включая нерастаможенные

Продавец

только частные лица ▾

Год выпуска

2012 ▾ до ▾

Пробег

от ▾ до ▾

Коробка передач

автомат  механика

КУЗОВ ▾

МАРКА ▾

ДВИГАТЕЛЬ ▾

ДЕТАЛИ ▾





## 1-комнатная 39...

**6 000 000** **₽**

● Василеостровская,  
17 мин. пешком

4 этаж из 6

Санкт-Петербург,  
Биржевая лин., 1к1

24 января

Подпишитесь, чтобы не пропустить выгодные предложения

dimas@yandex-team.ru



Подписаться



# Features

## Features

- › **Stream** of offers (tens per second)

## Features

- › **Stream** of offers (tens per second)
- › Up to **500K** subscriptions

## Features

- › **Stream** of offers (tens per second)
- › Up to **500K** subscriptions
- › Complex **multi field** conditions

## Features

- › **Stream** of offers (tens per second)
- › Up to **500K** subscriptions
- › Complex **multi field** conditions
- › **Low** latency (seconds)

Motivation

# Customer subscriptions



## Customer subscriptions

- › **Short** notification delays (minutes)

## Customer subscriptions

- › **Short** notification delays (minutes)
- › **Instant** subscription modifications

## Customer subscriptions

- › **Short** notification delays (minutes)
- › **Instant** subscription modifications
- › Notifications **transport**
  - › email, SMS, push, pigeons, owls, etc.

# Non-functional requirements

## Non-functional requirements

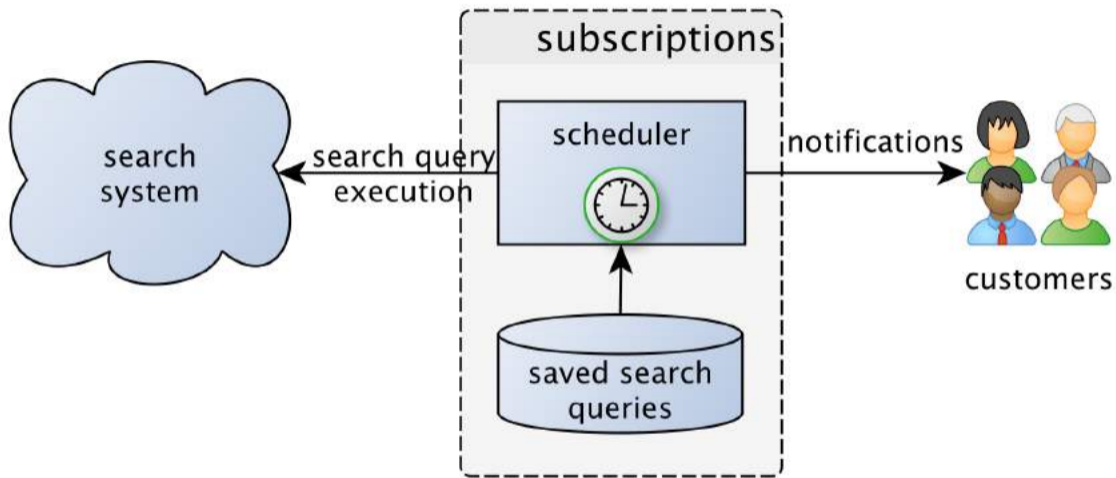
- › Scalability by **subscriptions**
  - › millions

## Non-functional requirements

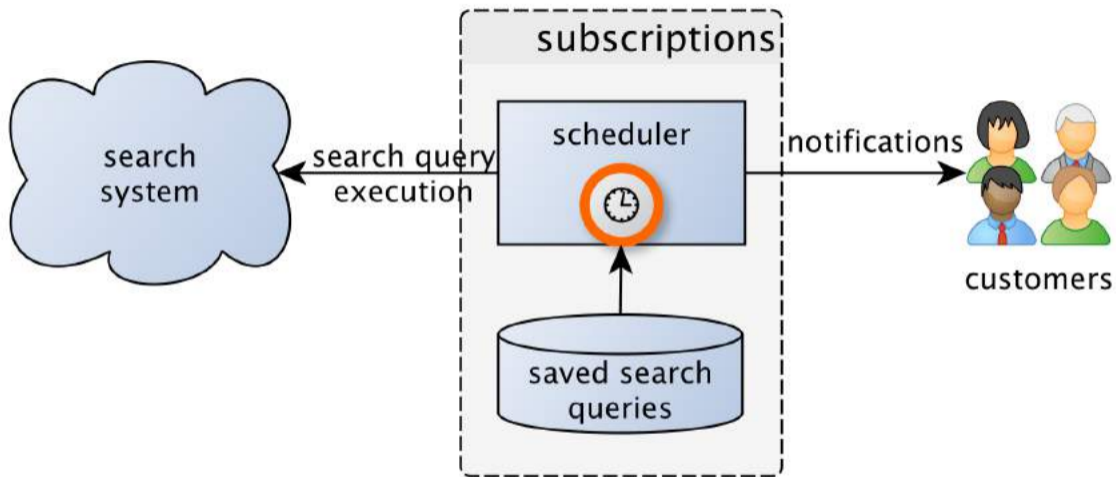
- › Scalability by **subscriptions**
  - › millions
- › Scalability by **events**
  - › tens-hundreds per second

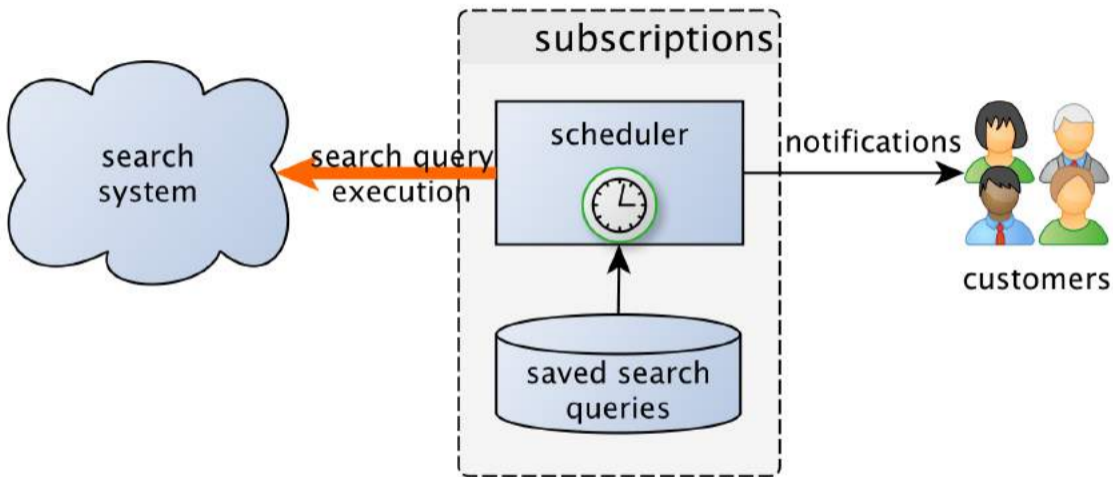
## Non-functional requirements

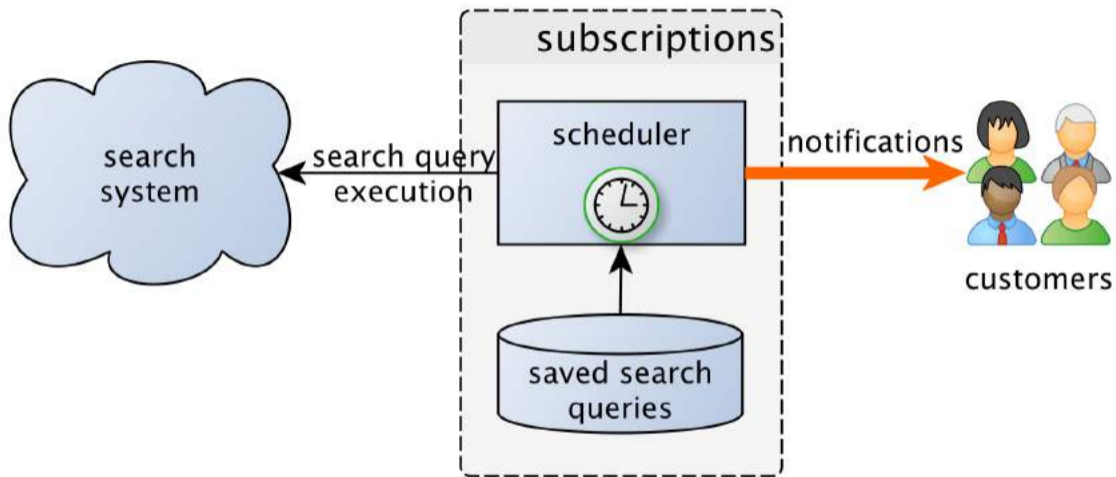
- › Scalability by **subscriptions**
  - › millions
- › Scalability by **events**
  - › tens-hundreds per second
- › **Fault** tolerance
  - › nodes, network, data centers

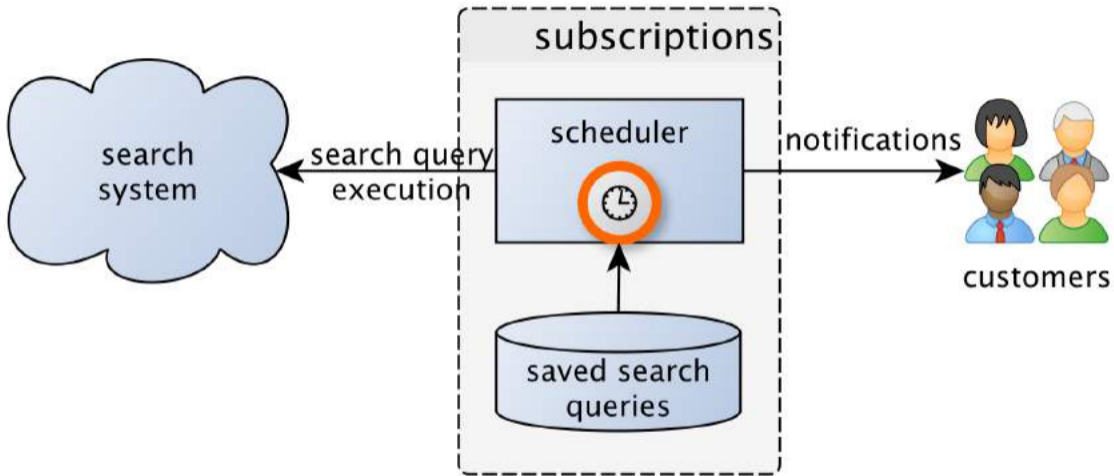


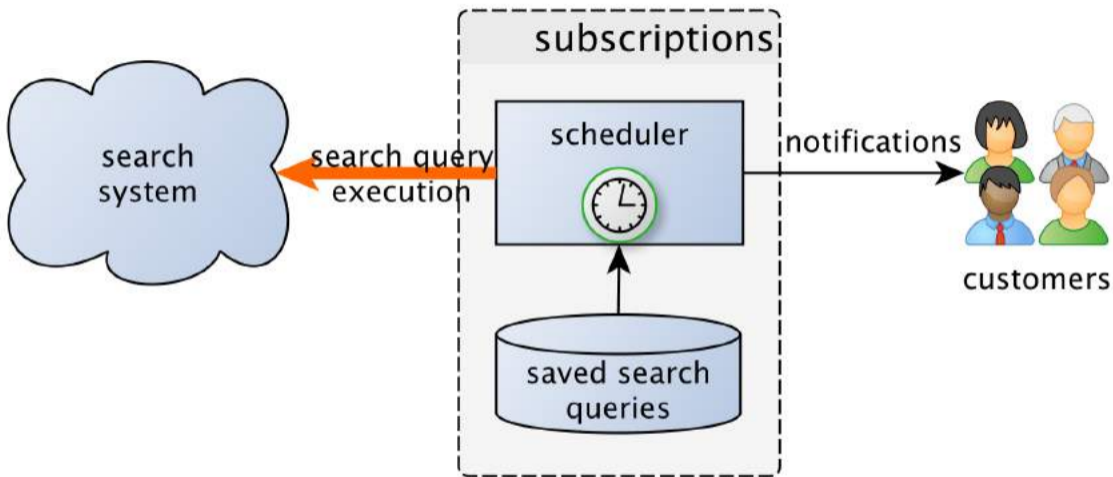


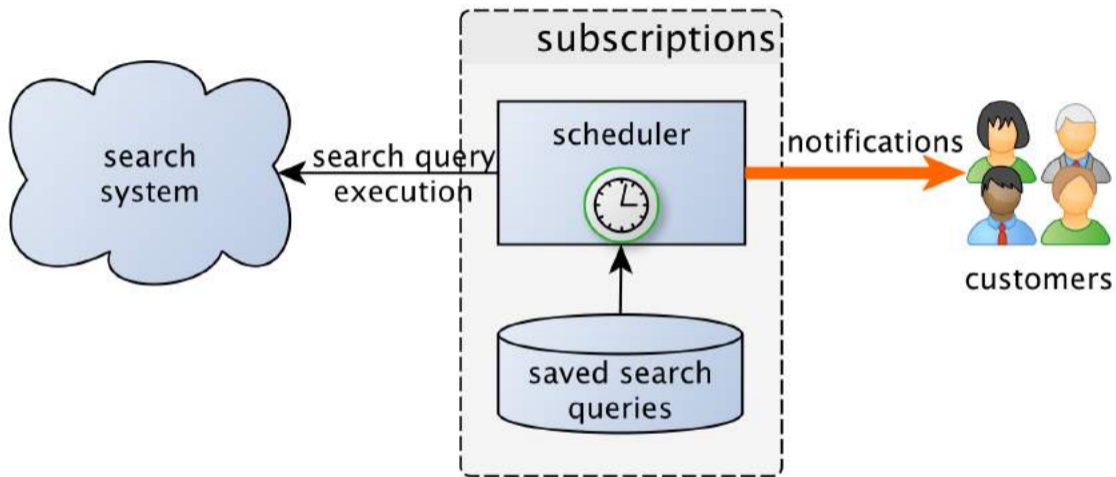












## Naive solution

### Pros:

- › Simplicity
- › Working

## Naive solution

### Pros:

- › Simplicity
- › Working

### Cons:



## Naive solution

### Pros:

- › Simplicity
- › Working

### Cons:

- › High notification latency (hours, days)

## Naive solution

### Pros:

- › Simplicity
- › Working

### Cons:

- › High notification latency (hours, days)
- › Search system extra load

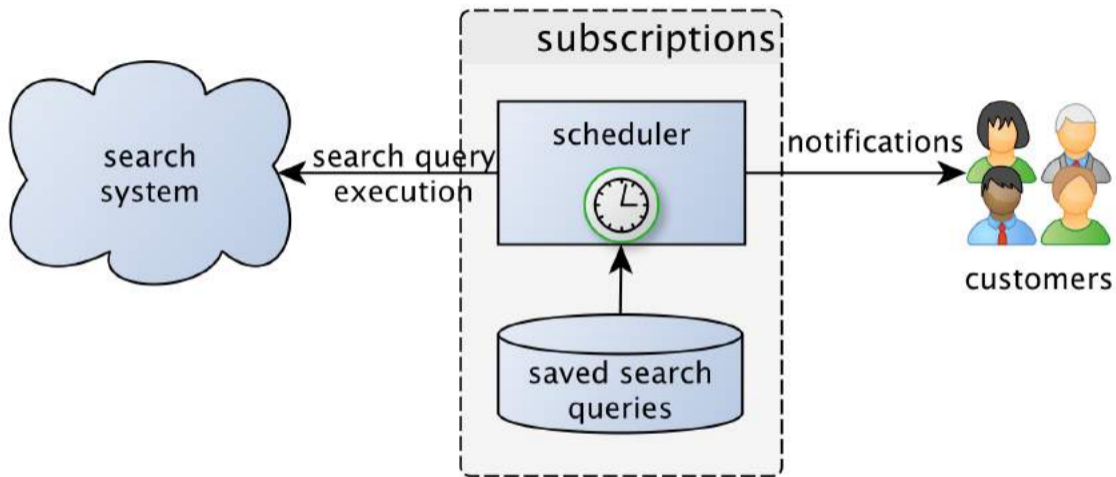
## Naive solution

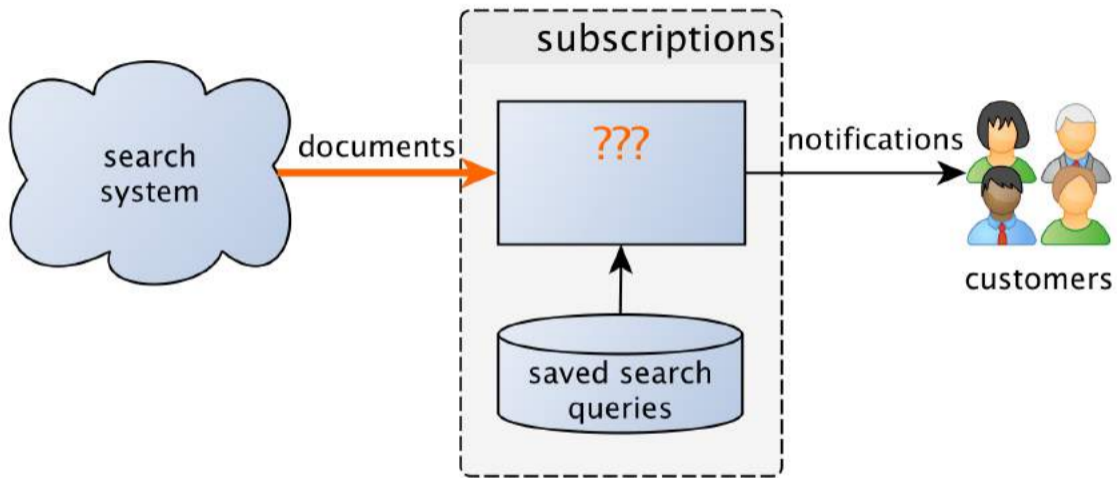
### Pros:

- › Simplicity
- › Working

### Cons:

- › High notification latency (hours, days)
- › Search system extra load
- › Scalability?





Not so naive approach

Not so naive approach

Pros:

# Not so naive approach

## Pros:

- › Low latency



## Not so naive approach

### Pros:

- › Low latency
- › No extra work

## Not so naive approach

### Pros:

- › Low latency
- › No extra work
- › Possibly scalable

## Not so naive approach

### Pros:

- › Low latency
- › No extra work
- › Possibly scalable

### Cons:

## Not so naive approach

### Pros:

- › Low latency
- › No extra work
- › Possibly scalable

### Cons:

- › More complex

## Not so naive approach

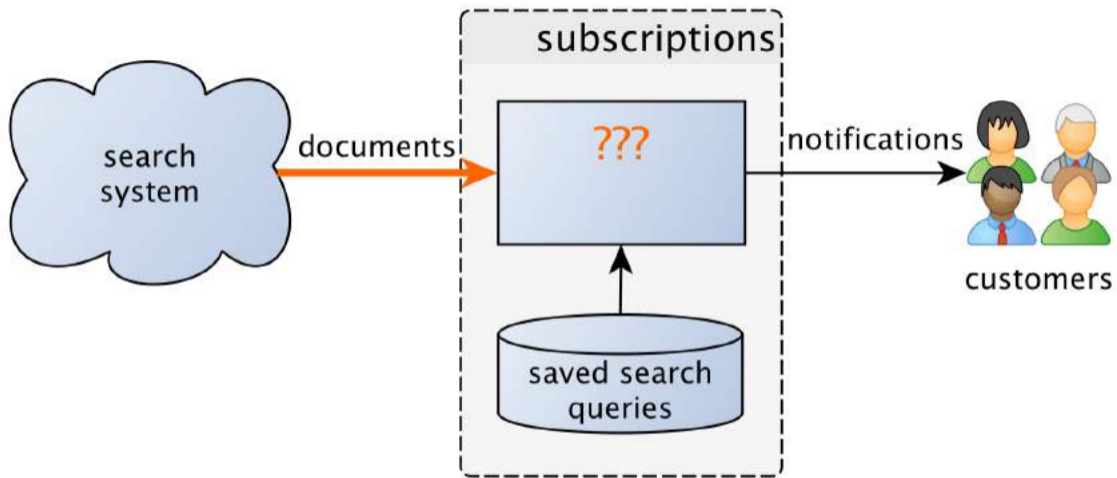
### Pros:

- › Low latency
- › No extra work
- › Possibly scalable

### Cons:

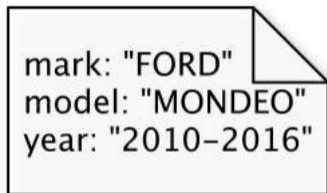
- › More complex, **but not complicated**

Matching



# Data model

## › Event:



## › Subscription:

mark=FORD&model=FOCUS&model=MONDEO&year=2015



## Статьи

Моя библиотека

За все время

С 2016

С 2015

С 2012

Выбрать даты

По релевантности

По дате

 включая патенты показать цитаты**Matching events in a content-based subscription system**[MK Aguilera](#), [RE Strom](#), [DC Sturman](#), [M Astley...](#) - [Proceedings of the ...](#), 1999 - [dl.acm.org](#)

... Pre-processing makes sense in most **pub/sub** environments, where subscriptions tend to change infrequently ... know, there are no other algorithms for the **matching** problem with **sub**-linear time ... The **content-based** subscription systems that have been de-veloped so far have not ...

Цитируется: 800 [Похожие статьи](#) [Все версии статьи \(18\)](#) [Цитировать](#) [Сохранить](#)**An efficient multicast protocol for content-based publish-subscribe systems**[G Banavar](#), [T Chandra](#), [B Mukherjee...](#) - [Distributed ...](#), 1999 - [ieeexplore.ieee.org](#)

... **pub/sub**. In a companion paper [2] we present an efficient solution to the **matching** problem for these systems. There are two straightforward approaches to solving the multicasting problem for **content-based** systems: (1) The **match**-first approach, where the event is first **matched** ...

Цитируется: 670 [Похожие статьи](#) [Все версии статьи \(17\)](#) [Цитировать](#) [Сохранить](#)**Efficient event routing in content-based publish-subscribe service networks**[F Cao](#), [JP Singh](#) - ... . [Twenty-third Annual Joint Conference of the ...](#), 2004 - [ieeexplore.ieee.org](#)

... it is expected to be difficult to form only a few groups to **match** ever server's ... not been comprehensive comparison and evaluation of different event routing schemes for **content-based pub-sub** network. ... In ideal multicast, each event is sent to **matching** servers through IP multicast ...

Цитируется: 169 [Похожие статьи](#) [Все версии статьи \(12\)](#) [Цитировать](#) [Сохранить](#)

A better way

## A better way

### Commonalities:

- > `mark=FORD&model=FOCUS`
- > `mark=FORD&year=2010`
- > `year=2010&price_from=500000`

## A better way

### Commonalities:

- > `mark=FORD&model=FOCUS`
- > `mark=FORD&year=2010`
- > `year=2010&price_from=500000`

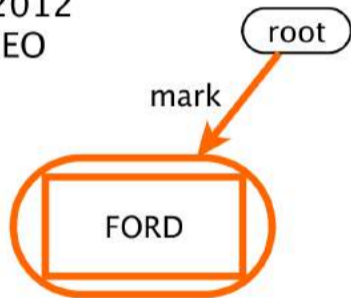
### Data structures

- > BDD
- > DAG
- > Search tree
- > Indices

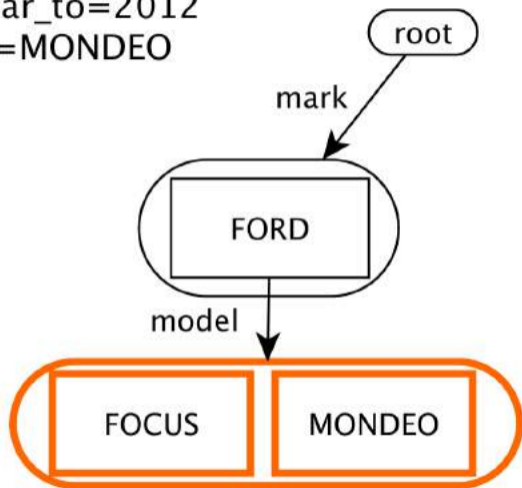
 S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016  
S2: year\_from=2010&year\_to=2012  
S3: mark=FORD&model=MONDEO

root

★ S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016  
S2: year\_from=2010&year\_to=2012  
S3: mark=FORD&model=MONDEO



- ★ S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016  
S2: year\_from=2010&year\_to=2012  
S3: mark=FORD&model=MONDEO

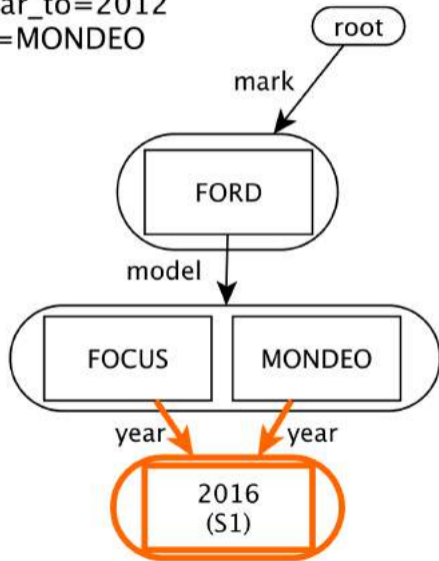




S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

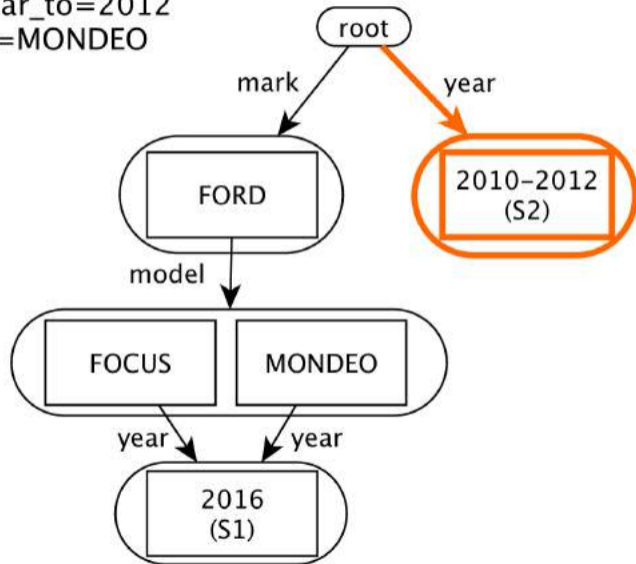
S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO





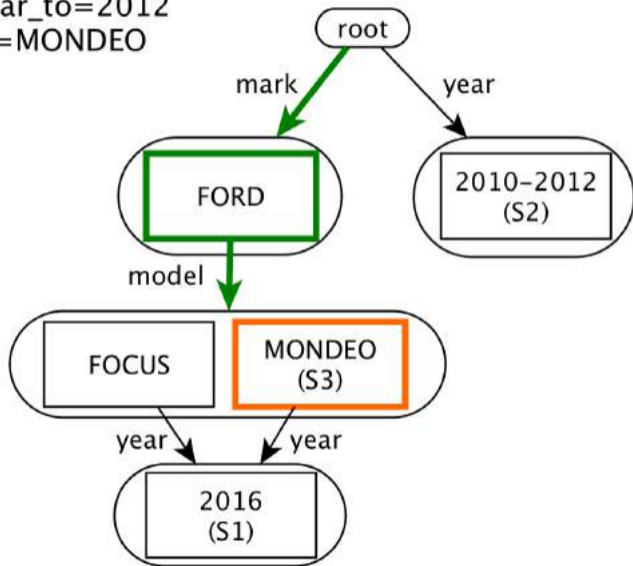
- ★ S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016  
S2: year\_from=2010&year\_to=2012  
S3: mark=FORD&model=MONDEO



S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

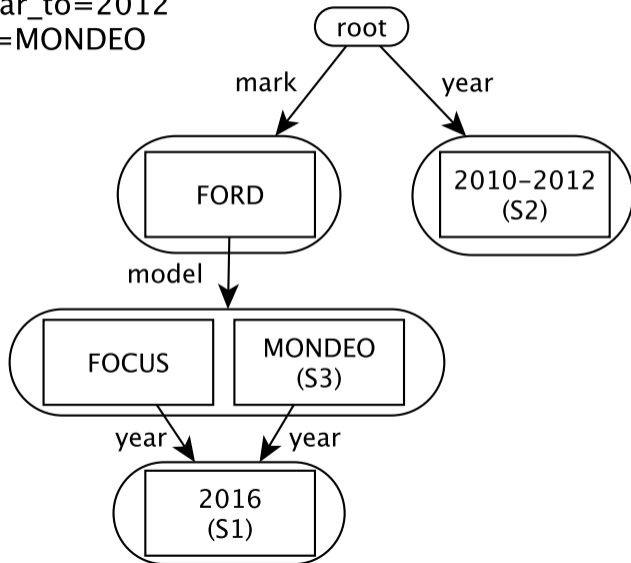
S3: mark=FORD&model=MONDEO



S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO



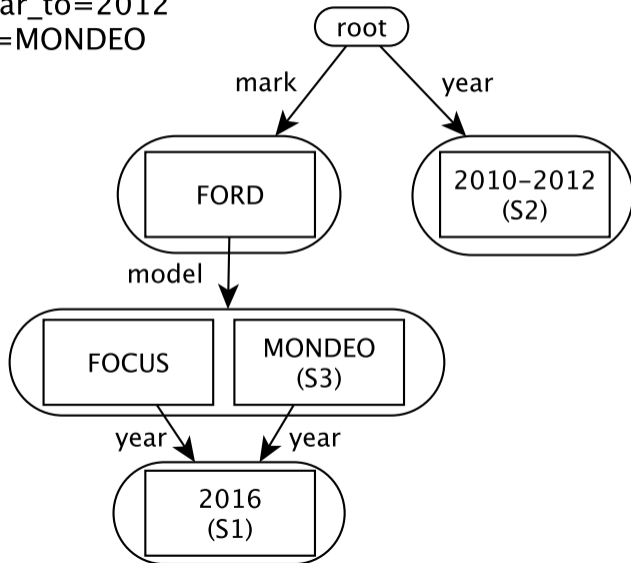
S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO

mark: "FORD"  
model: "MONDEO"  
year: "2016"

Subscriptions  
{ }



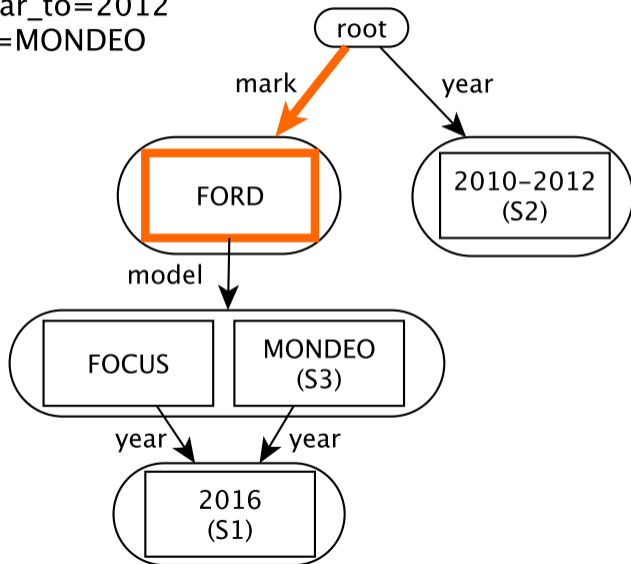
S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO

mark: "FORD"  
model: "MONDEO"  
year: "2016"

Subscriptions  
{ }



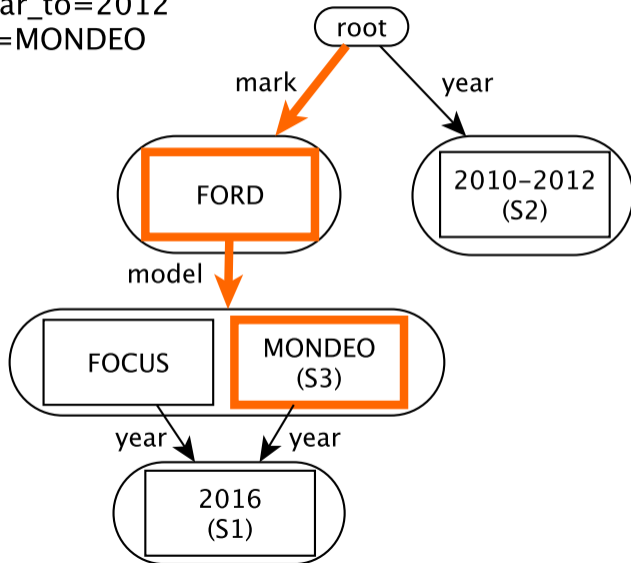
S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO

mark: "FORD"  
model: "MONDEO"  
year: "2016"

Subscriptions  
{S3}



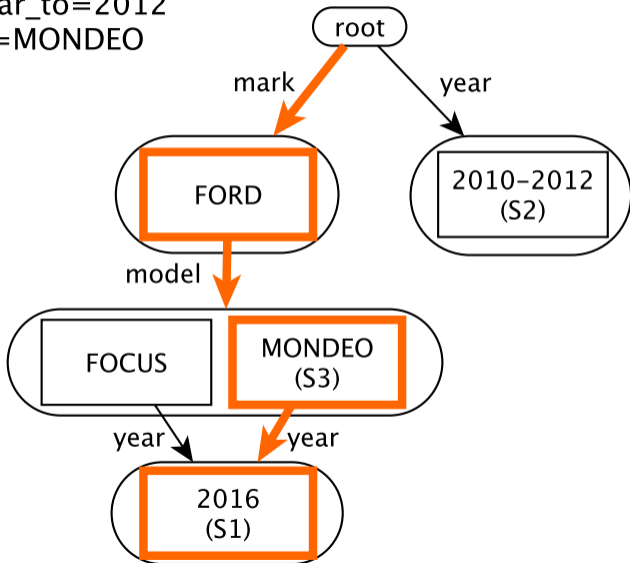
S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO

mark: "FORD"  
model: "MONDEO"  
year: "2016"

Subscriptions  
{S3, S1}



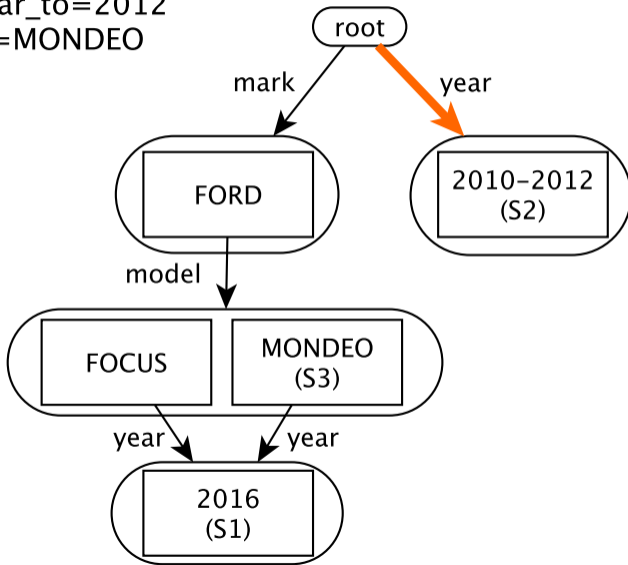
S1: mark=FORD&model=FOCUS&model=MONDEO&year=2016

S2: year\_from=2010&year\_to=2012

S3: mark=FORD&model=MONDEO

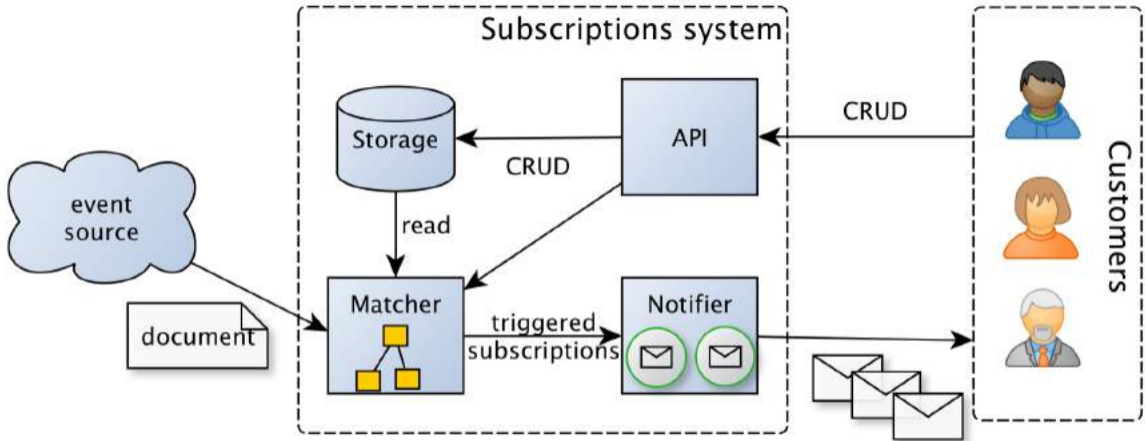
mark: "FORD"  
model: "MONDEO"  
year: "2016"

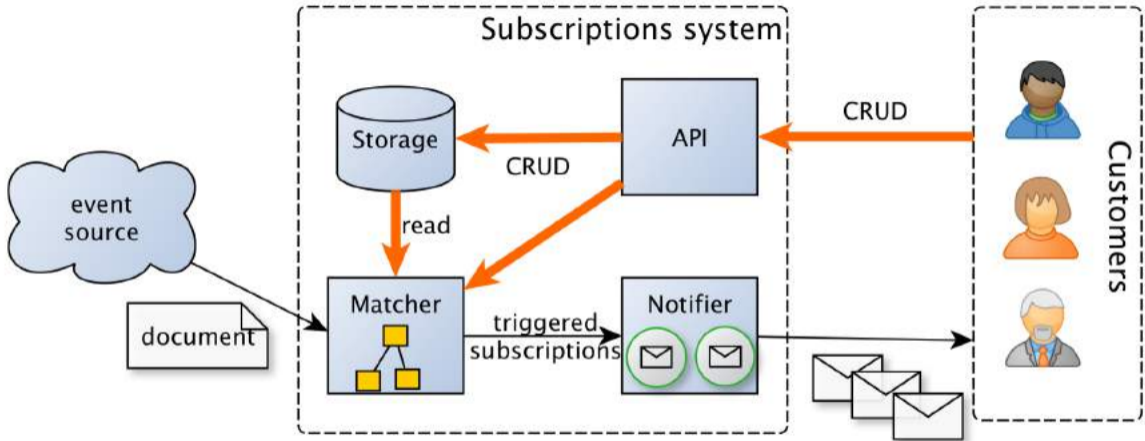
Subscriptions  
{S3, S1}

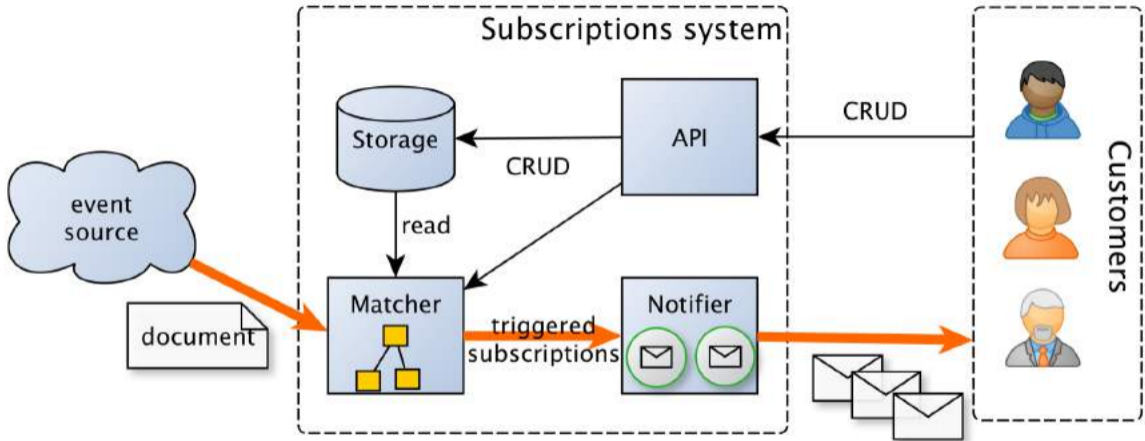




Architecture

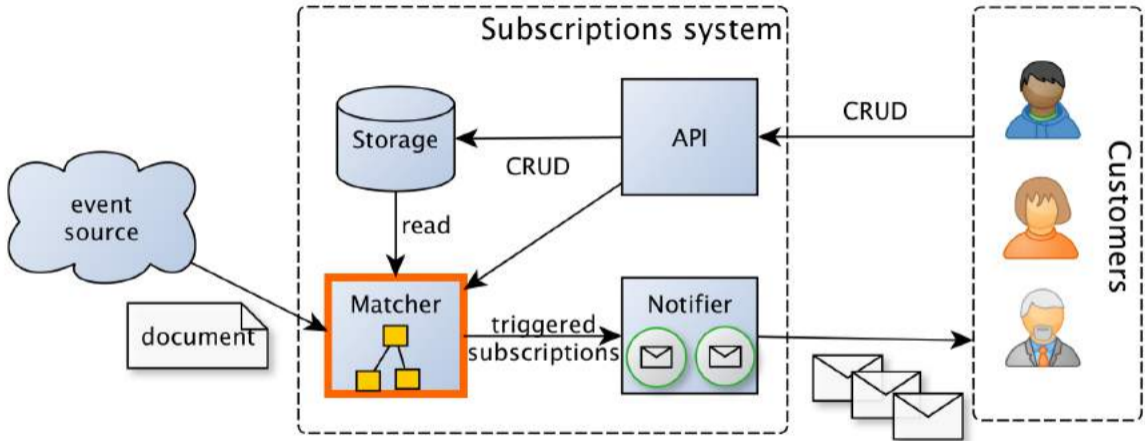


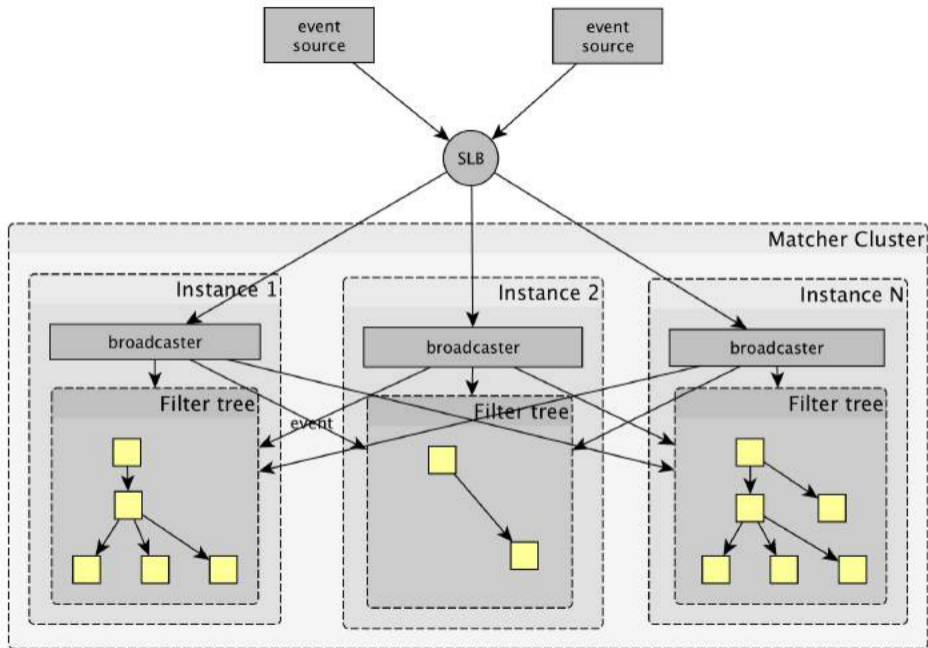


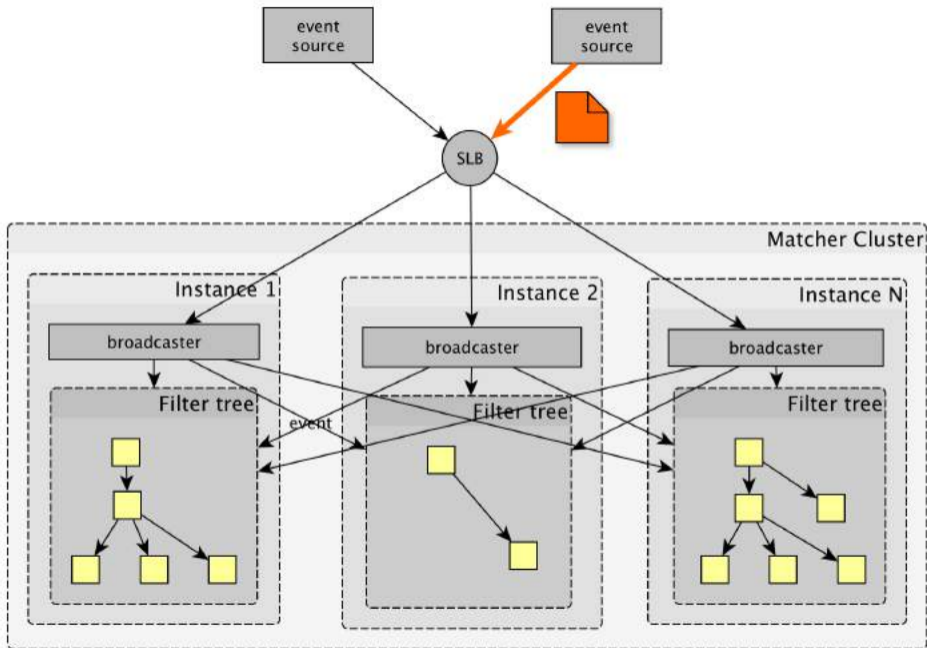


Architecture

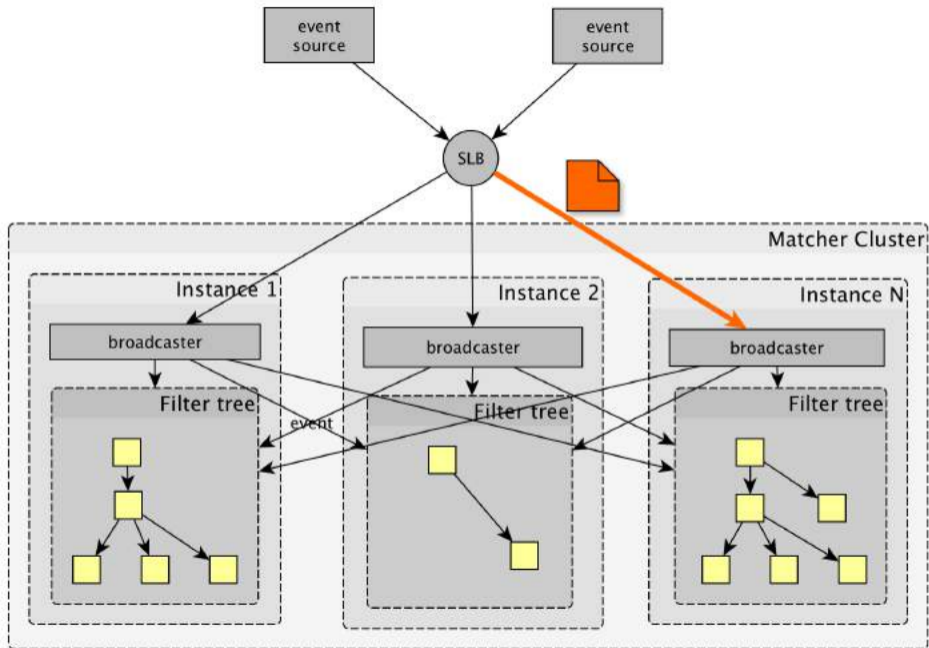
Matcher

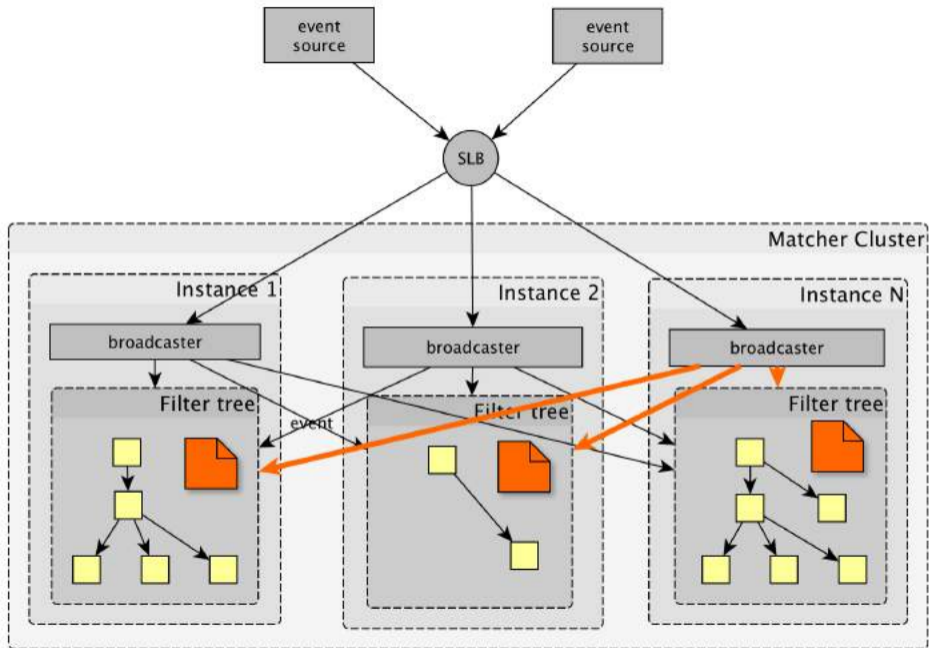


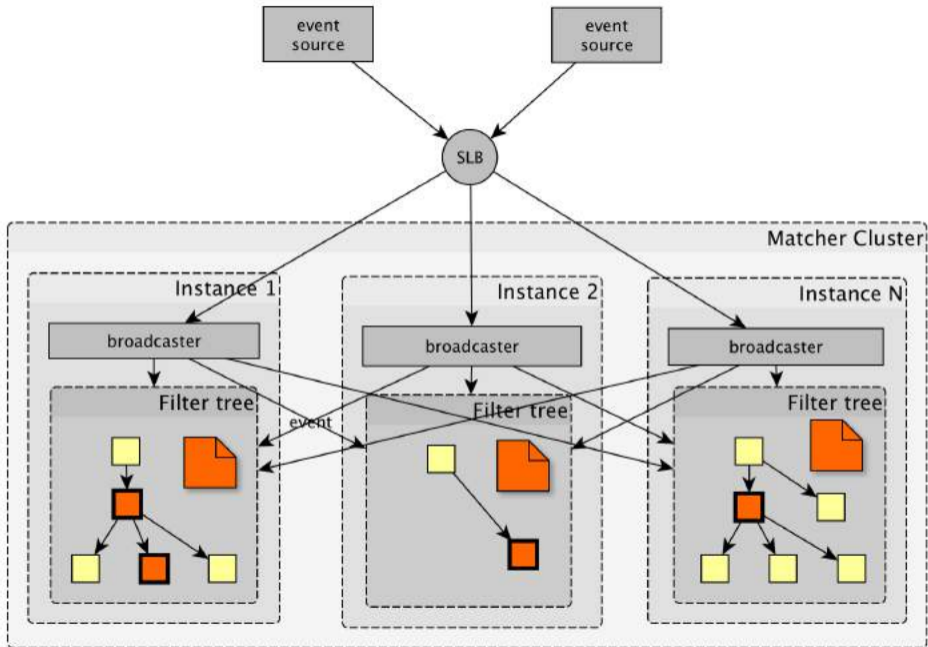












# Resilience

# Resilience

› Subscriptions => Tokens

# Resilience

- › Subscriptions  $\Rightarrow$  Tokens
- › Token  $\approx$  responsibility

# Resilience

- › Subscriptions => Tokens
- › Token  $\approx$  responsibility
- › Nodes distribute tokens

Node A

Token 1

Token 4

Token 5

Node B

Token 6

Token 2

Token 3

Token 7



Node A

Token 1

Token 4

Token 5

Node B

Token 6

Token 2

Token 3

Token 7

Node C

Node A

Token 1

Token 4

Token 5

Node B

Token 6

Token 2

Token 3

Node C

Token 7

Node A

Token 1

Token 4

Node B

Token 6

Token 2

Token 3

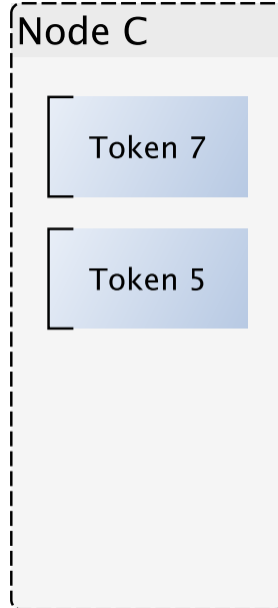
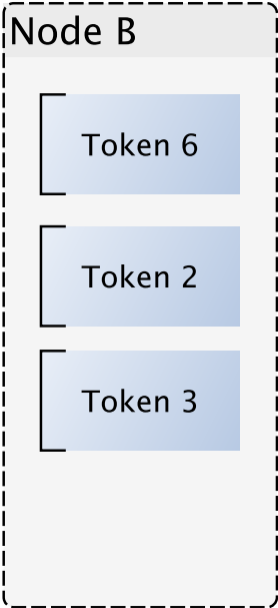
Node C

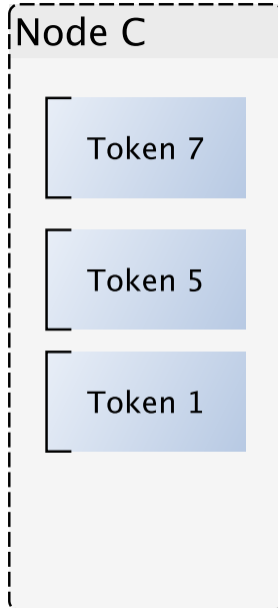
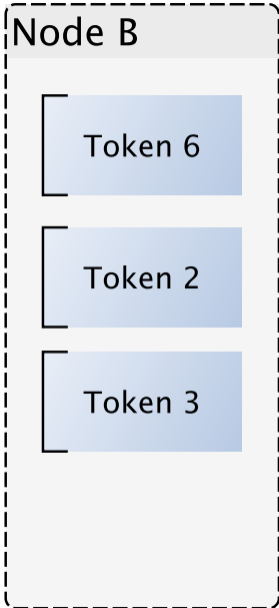
Token 7

Token 5

Token 1

Token 4





## Node B

Token 6

Token 2

Token 3

Token 4

## Node C

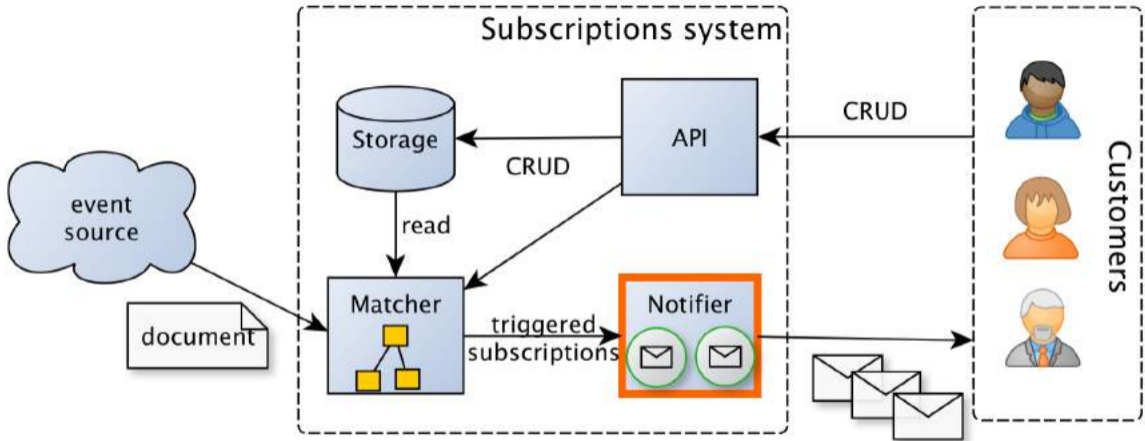
Token 7

Token 5

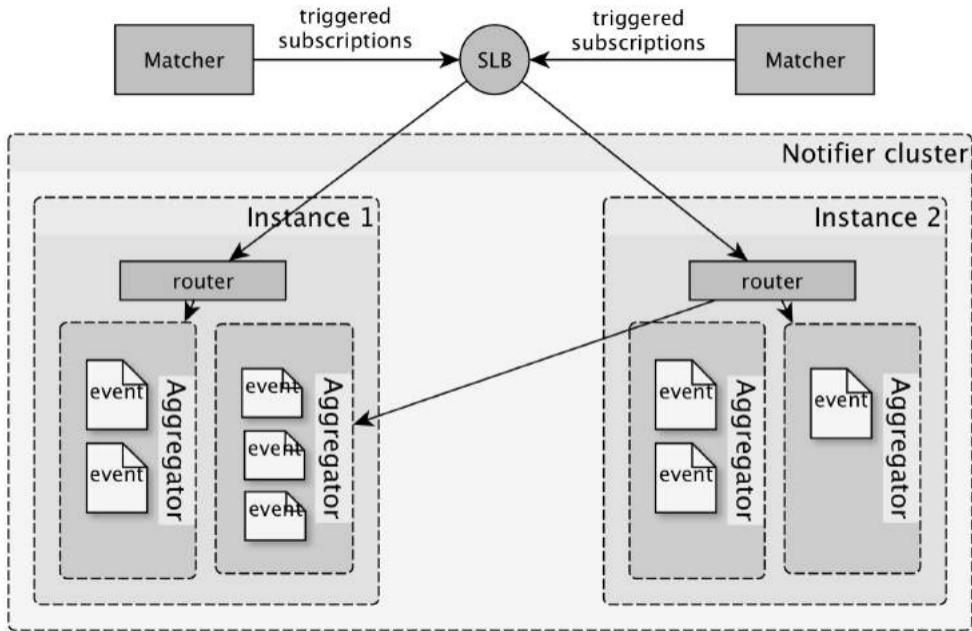
Token 1

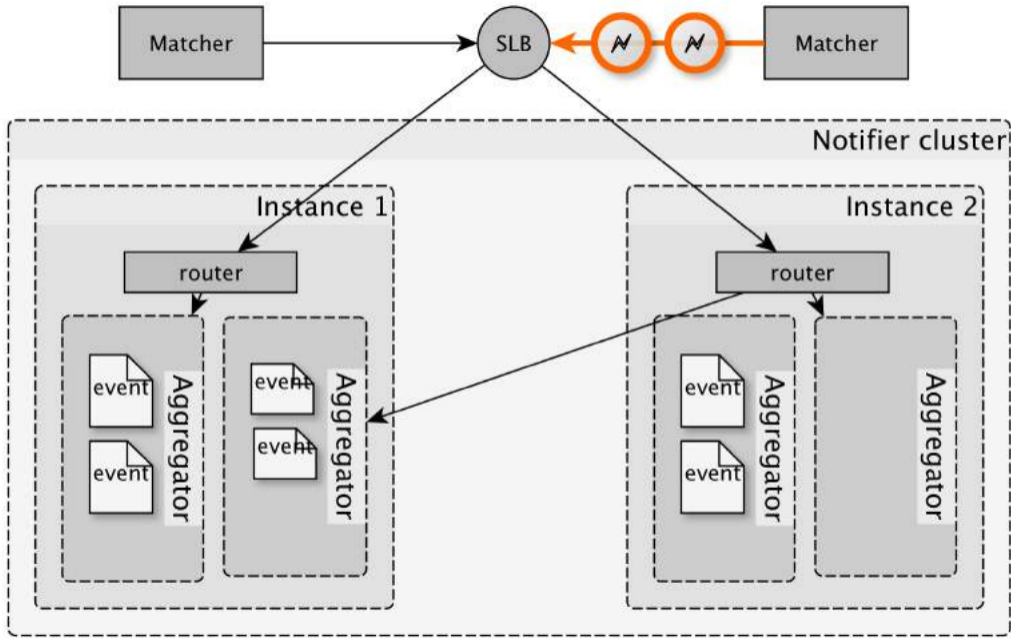
Architecture

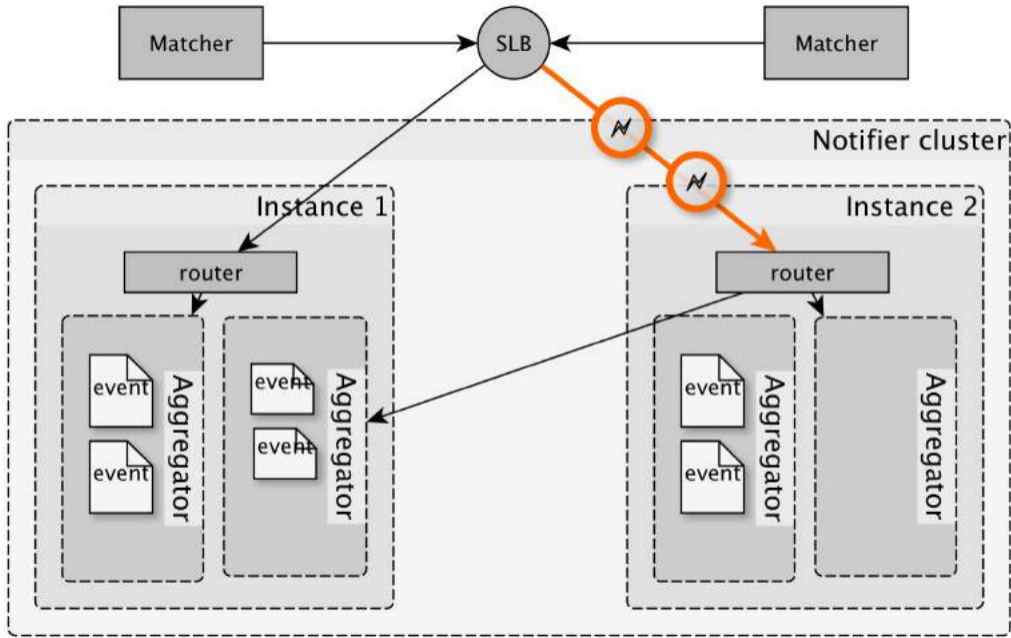
Notifier

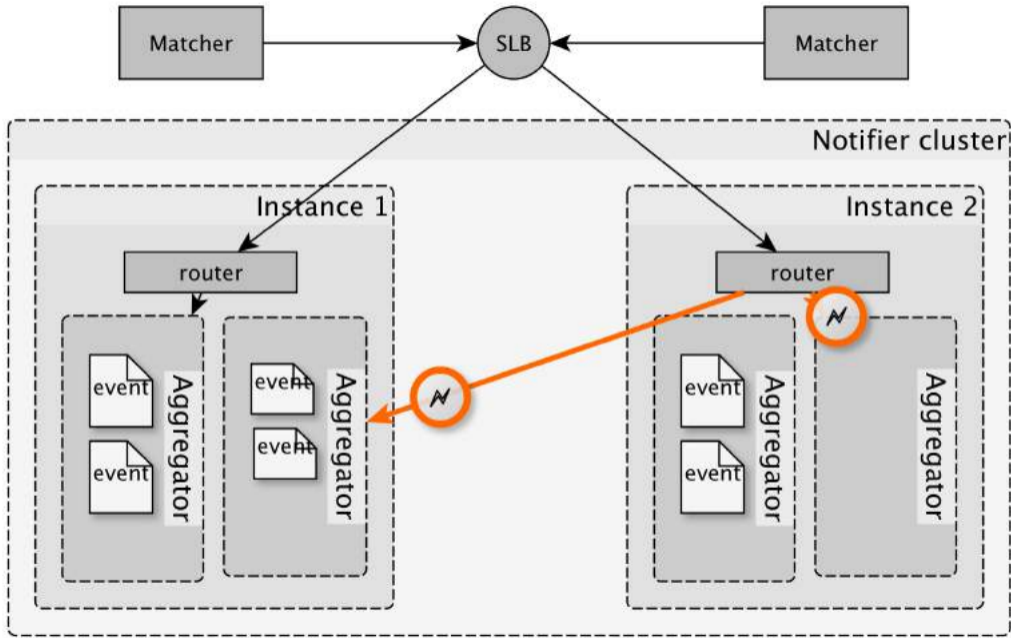


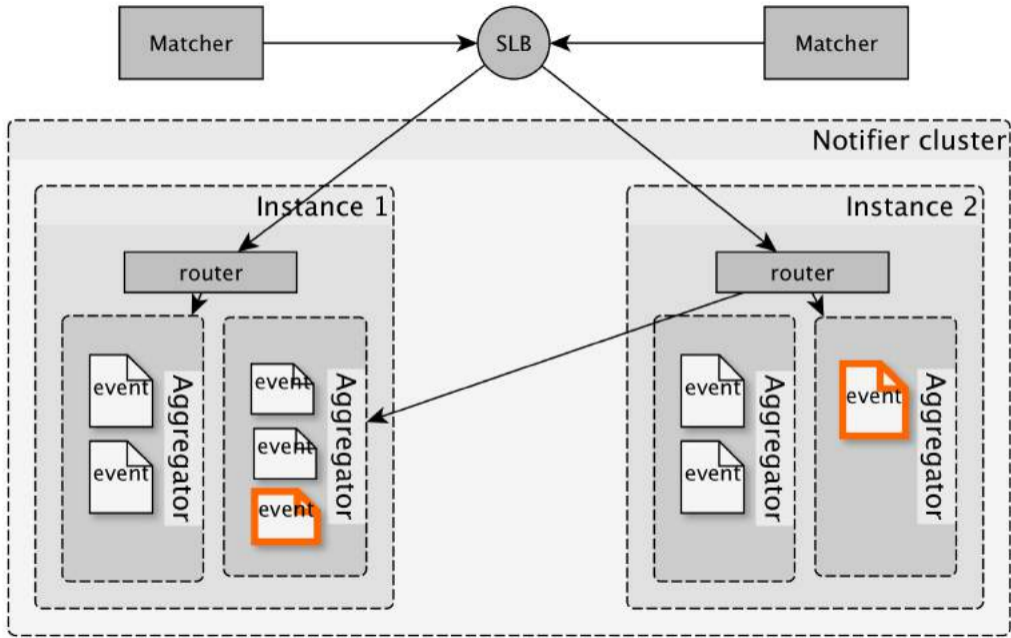












# Notifier cluster

## Instance 1

router



event

Aggregator

event

event

Aggregator

event

event

aggregated state dump



persistent storage

## Instance 2

router



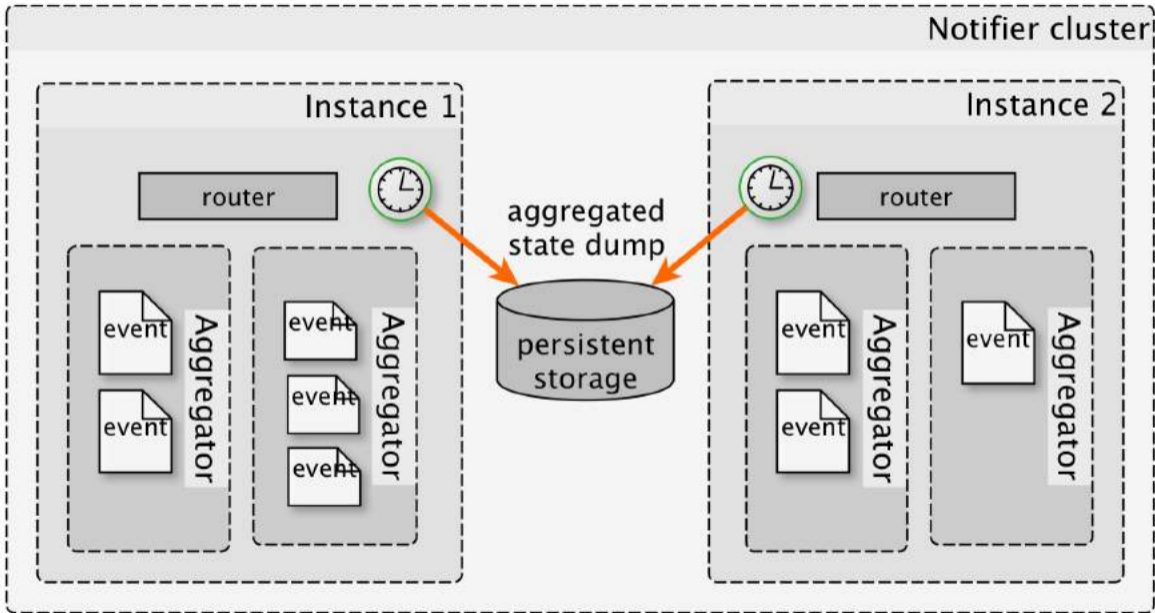
event

Aggregator

event

event

Aggregator



Implementation

# Scala

---

<sup>1</sup><http://www.scalacheck.org>



# Scala

- › Functional & concise

---

<sup>1</sup><http://www.scalacheck.org>

# Scala

- › Functional & concise
- › Persistent data structures for matching tree  
(1Kloc)

---

<sup>1</sup><http://www.scalacheck.org>

# Scala

- › Functional & concise
- › Persistent data structures for matching tree  
(1Kloc)
  - › No need for synchronization

---

<sup>1</sup><http://www.scalacheck.org>

# Scala

- › Functional & concise
- › Persistent data structures for matching tree  
(1Kloc)
  - › No need for synchronization
- › Property-based testing (matching algorithm)

---

<sup>1</sup><http://www.scalacheck.org>

Akka

# Akka

› `actors` (events)

# Akka

- › `actors` (events)
- › `akka-remoting` (networking)

# Akka

- › `actors` (events)
- › `akka-remoting` (networking)
- › `protobuf` (serialization)



# Akka

- › `actors` (events)
- › `akka-remoting` (networking)
- › `protobuf` (serialization)
- › `akka-fsm` (aggregation)

# Zookeeper

- › Exclusive token acquisition
- › Service discovery



Conclusion

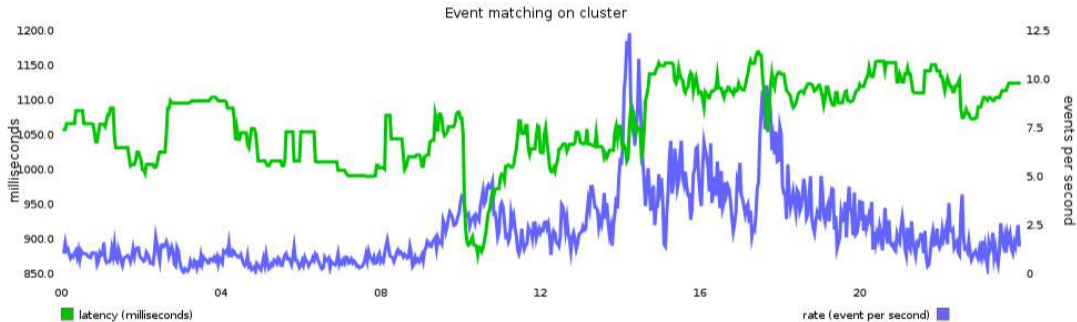
# Achievements

- › Low customer notification latency (seconds)
- › No extra load
- › All on 3 nodes x 2 DC

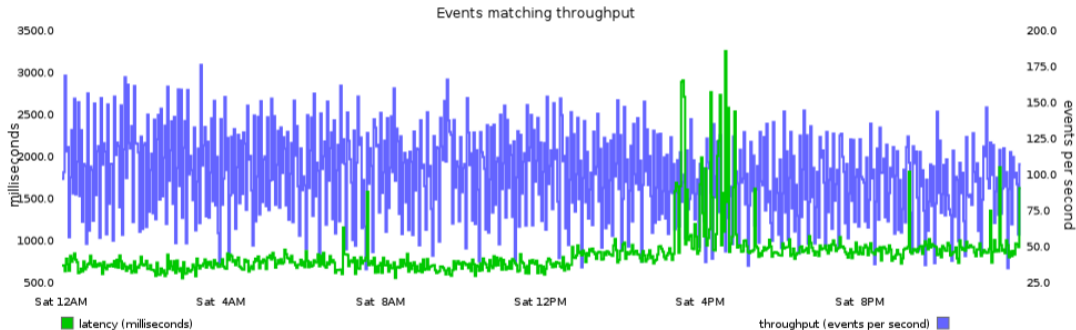
# Achievements

- › Low customer notification latency (seconds)
- › No extra load
- › All on 3 nodes x 2 DC
  - › 200K subs — 1G heap

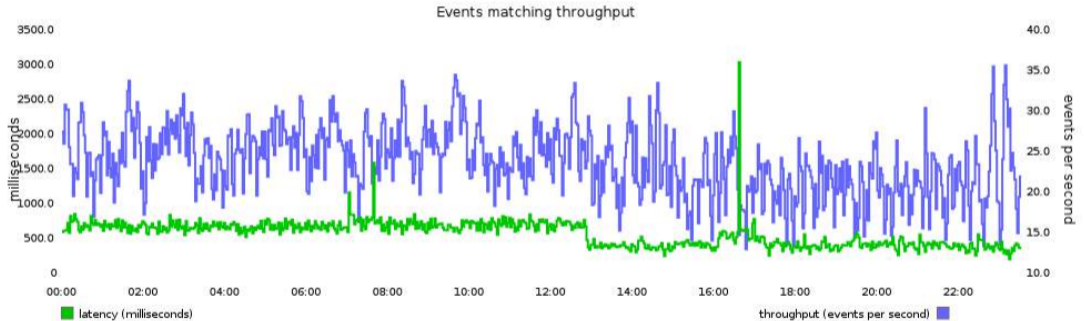
# 300K subscriptions in production



# 300K subscriptions stress testing (cluster)



# 150K subscriptions stress testing (node)





# Contacts

Dmitry Schitinin

Backend infrastructure team @ Yandex.Classifieds



[dimas@yandex-team.ru](mailto:dimas@yandex-team.ru)