

# Альфа Сайта

## Знакомств за 180 дней

Работа над ошибками

Ринат Абдуллин  
HappyPancake  
[twitter.com/abdullin](https://twitter.com/abdullin)

# Обо мне

- До 2014 - .NET Tech Lead в lokad.com
- С 2014 - разработчик в HappyPancake
- Блог - [abdullin.com](http://abdullin.com) / [twitter.com/abdullin](https://twitter.com/abdullin)
- Подкасты
  - [BeingTheWorst.com](http://BeingTheWorst.com)
  - [DistributedPodcast.com](http://DistributedPodcast.com)
  - Happy Friday : [RD.HappyPancake.com](http://RD.HappyPancake.com)
- Я еще только учусь

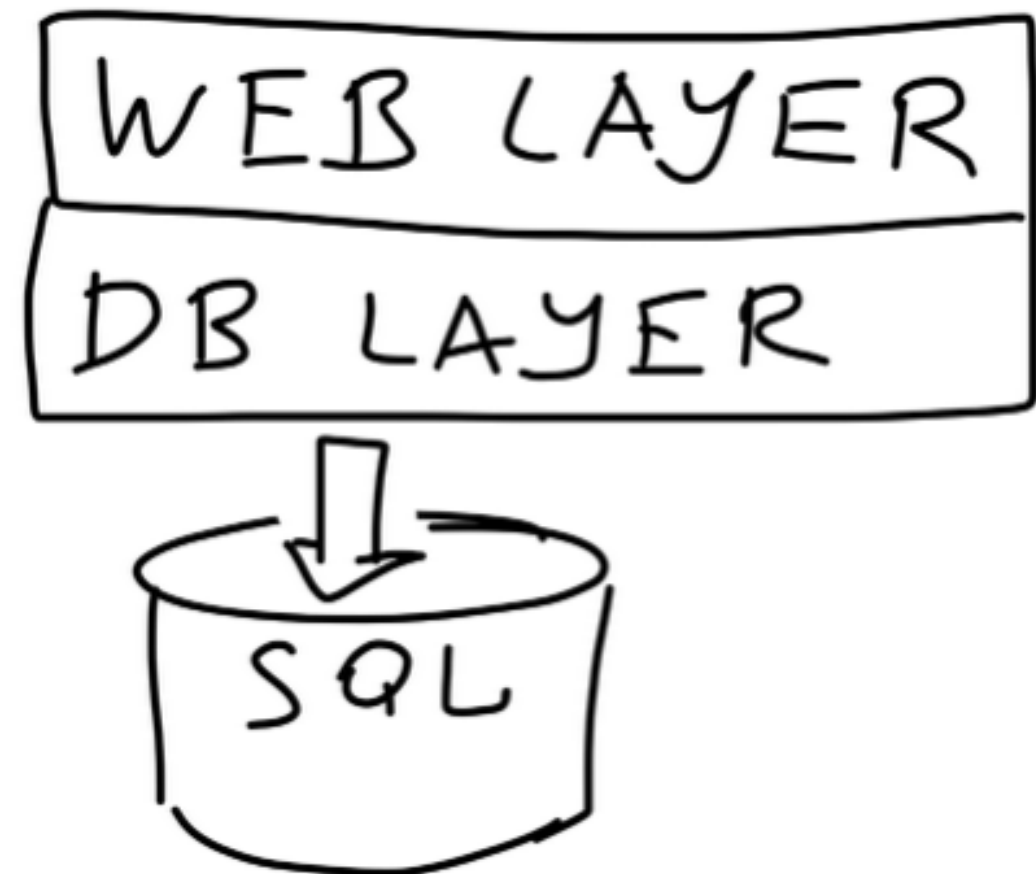
# HappyPancake.com

- Крупнейшая сеть знакомств в Швеции
- Каждый 10-й швед в сети
- Норвегия и Финляндия
- Бесплатная, доход за счет рекламы
- С 2014г - 3 разработчика

The screenshot displays the HappyPancake.com website. At the top, the logo 'HAPPYPANCAKE™' is visible. Below it, there's a login section with a message 'Oops! You failed to login.' and a form containing an email field (with 'rinalabdu...@gmail.com' as a placeholder), a password field, and an 'OK' button. To the right of the login section, there's a green banner with the text 'SVERIGES STÖRSTA NÄTDEJTINGSAJT 100% GRATIS'. Below this banner, there's a 'SIGN UP HERE' button. The main content area is titled 'CREATE A PROFILE' and is divided into several sections: 'User Information' with fields for 'Smeknamn', 'Lösenord', 'Bekräfta lösenord', and 'E-postadress'; 'Information about you' with fields for '1978', '01', '01', 'Woman or man?', 'Select county', and 'Select Suburbs'; 'What kind of date do you want?' with a dropdown for 'Select the type of relation', a 'Woman or man?' dropdown, and an 'Age' range selector (18 to 30); and 'NEW TO HAPPY PANCAKE' with links for 'How it works', 'Conditions', and 'Sign up'. There's also a 'SAFETY & TIPS' section with links for 'Dating security', 'Tips for good presentation member!', 'Our pannkakerecept', and 'Find the right partner'. A 'SIGN UP NOW!' button is located at the bottom right of the profile creation section.

# Проблемы

- ASP.NET + SQL
- Бизнес логика в интересном месте
- Плохо масштабируется
- Страшно допиливать



# Хотелки

- Простая архитектура из небольших модулей
- 99% запросов (без кэша) - быстрее 25ms
- при 25000 запросов в секунду
- в виртуализированной среде
- Linux
- event-sourcing (для аналитики)

# Принципы дизайна

- Сервер состоит из модулей
- Модули - небольшие, работают вместе, реализуют фичи
- Реализация каждого модуля - индивидуальна
- Несколько экземпляров сервера за load-balancer
- Domain-Driven Design

Процесс дизайна

“Дизайн - это процесс изучения.  
Код - его побочный эффект.”

–CQRS Beers



1. Разбили проблему



Нарисовали план

# Фи́чи

- Регистрация
- Профили
- Чат
- Поиск
- Лайки
- Новости

## 2. Накостыляли модули

“Костыль - код, предназначенный для  
изменения в ближайшем будущем.”

–CQRS Beers

3. Выкидывали и  
переписывали код

# Выкинули в процессе

- 3 прототипа
- nanomsg
- FoundationDB
- mySQL
- event-sourcing
- etcd
- Haskell
- Erlang
- C#
- Lua
- Terminator



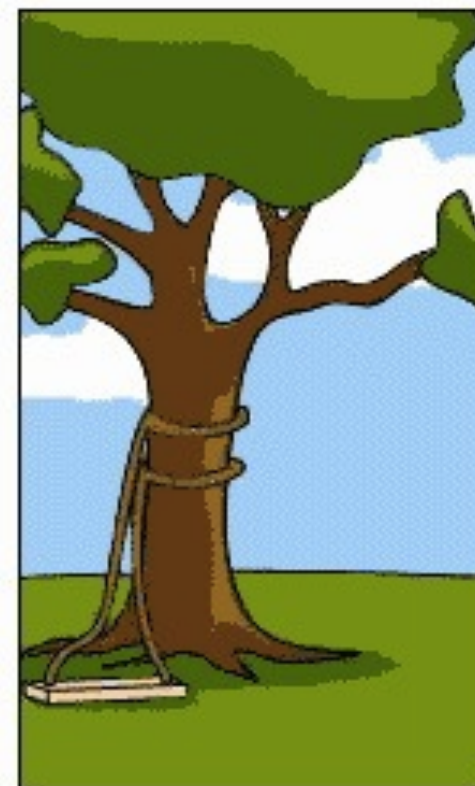
Как объяснил клиент  
чего он хочет



Как понял клиента  
начальник проекта



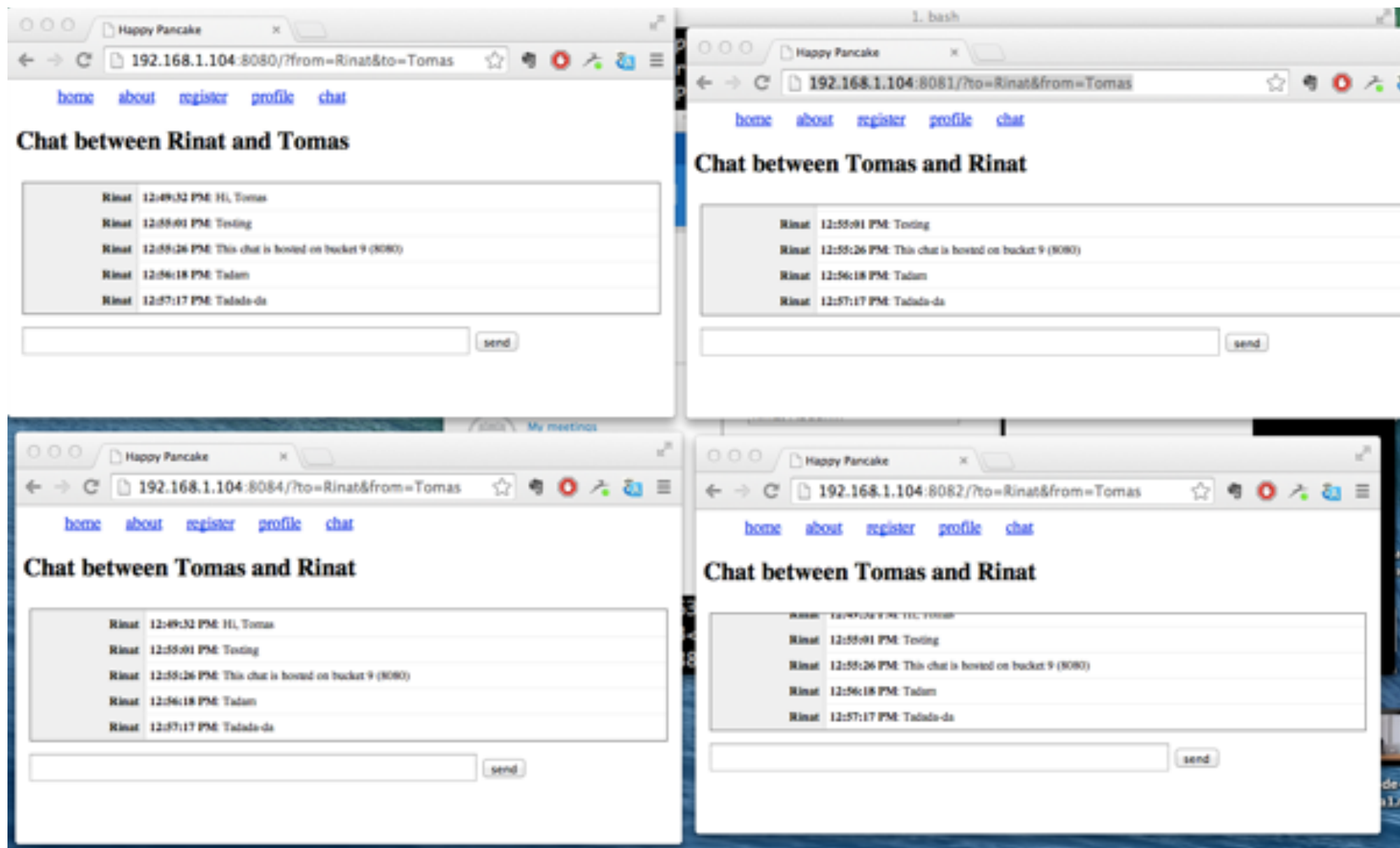
Как описал проект  
аналитик



Как написал  
программист

# Feedback необходим





# Первые результаты

Н. Переписывали  
модули индивидуально

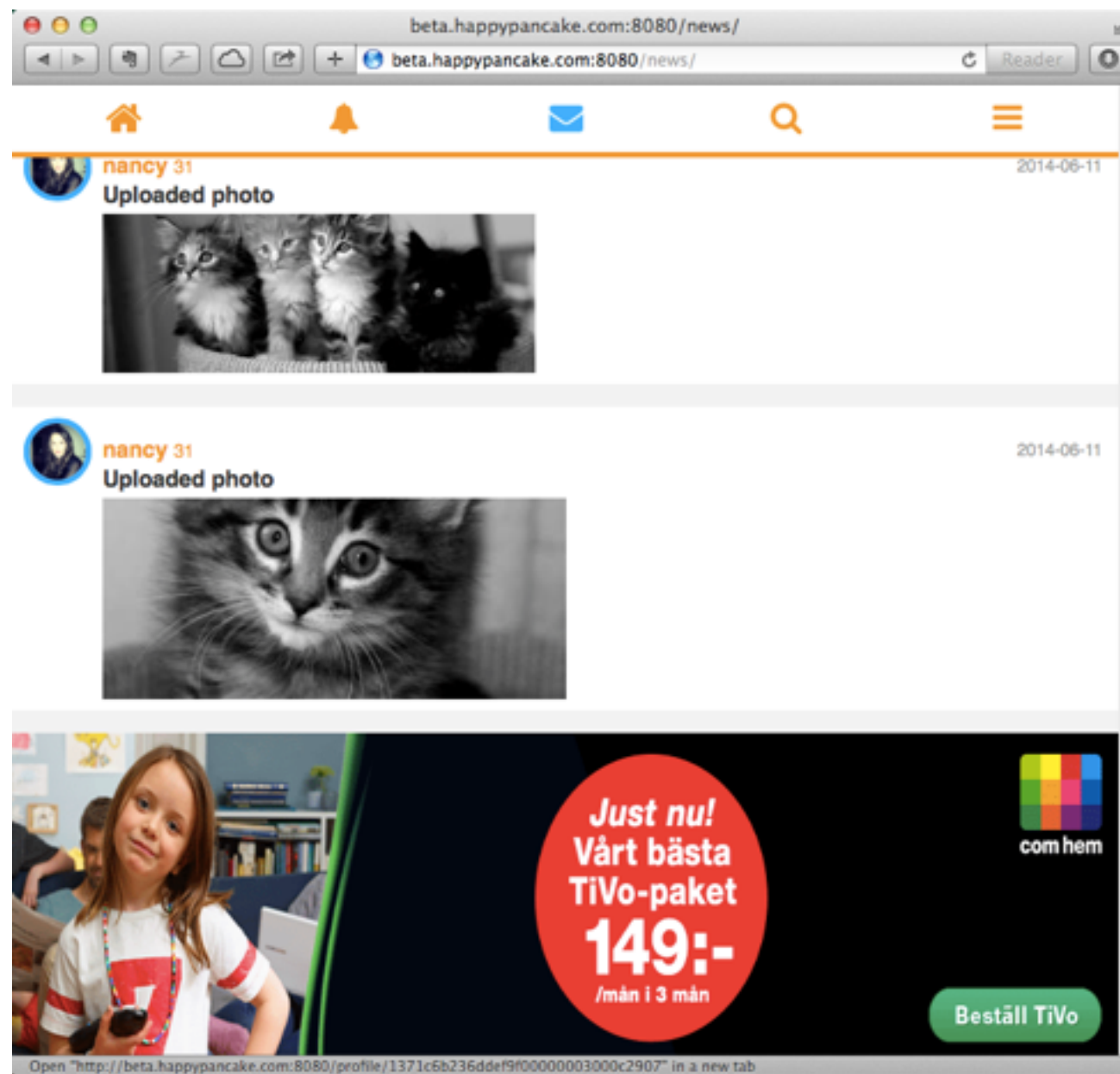
N. Корректировали  
карту

Повторять до  
посинения



Получится рабочая каша





# Первая демка

Стейкхолдеры - довольны!

# Заложено в дизайн

- A/B Testing
- Апгрейды без downtime
- Failover
- Cross-DC Failover
- CEP

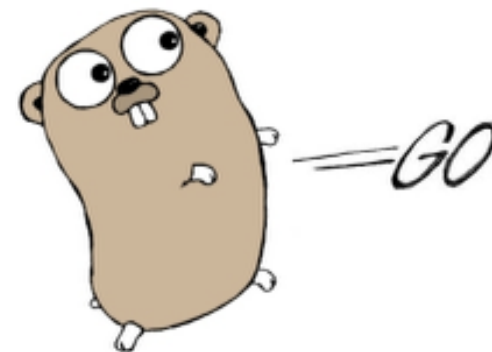
# Web

- jquery
- ajax
- http long polling (много старых телефонов)
- round-robin load balancer
- reverse proxies
- responsive UI
- designed for mobile



# Stack

- Ubuntu 12 LTS
- Go
- NSQ
- PostgreSQL
- Sublime / Vim / Emacs
- Docker
- Statsd + Librato Metrics
- Logsd + LogEntries



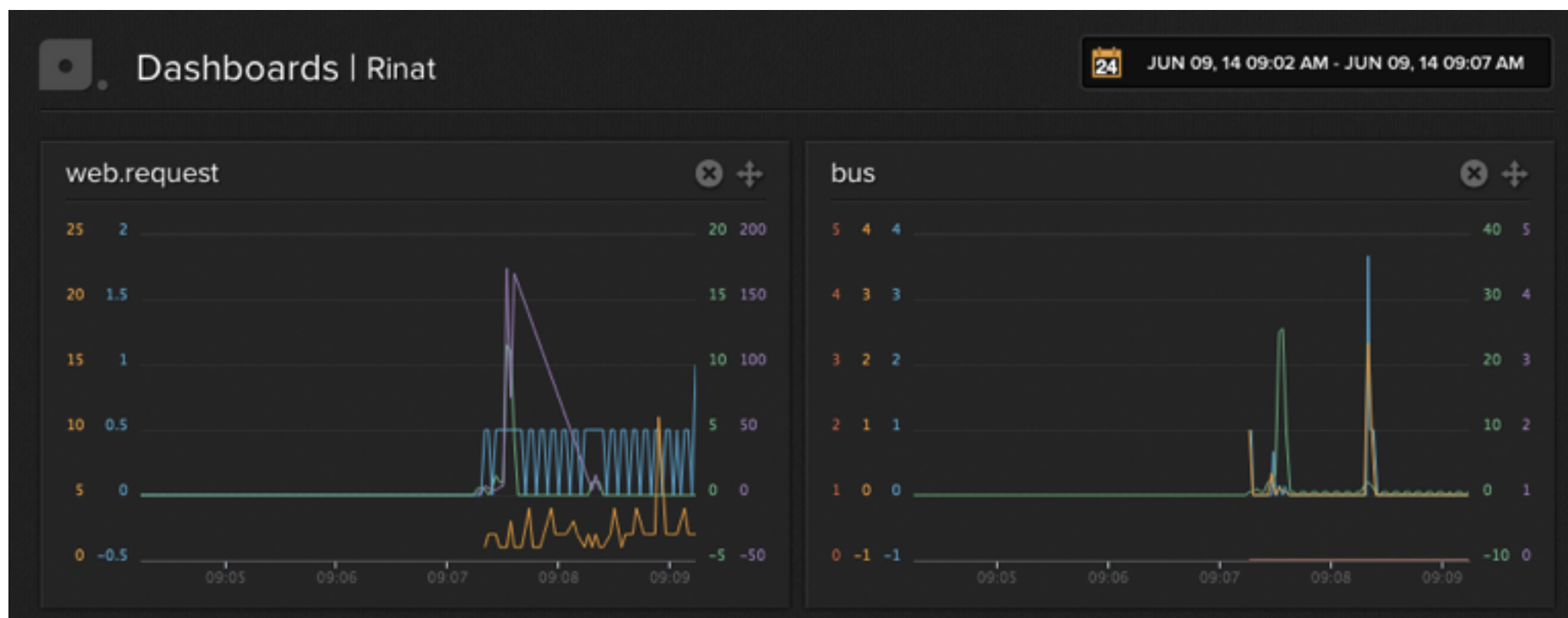
ubuntu



PostgreSQL  
the world's most advanced open source database



docker



# Метрики

```
var buffer []*Record
for {
    select {
    case r := <-spill:
        buffer = append(buffer, r)
        if len(buffer) >= 100 {
            persistMessages(buffer)
            buffer = make([]*Record, 0)
        }
    case <-time.After(time.Second):
        if len(buffer) > 0 {
            persistMessages(buffer)
            buffer = make([]*Record, 0)
        }
    }
}
```

# Go быстр и красив

123456789

NSQ Message Counts

NSQ chat

abdulell@ubuntu: ~

nsqadmin --lookupd-ht...

nsq\_to\_file --topic=test...

nsqd --lookupd-tcp-ad...

nsqlookupd

Mon Apr 21, 19:28

NSQ

Streams

Nodes

Counter

Lookup

NSQ on github

v0.2.26-alpha

36

Messages Processed

Empty Queue

Delete Topic

Pause Topic

Topic Message Queue

NSQd Host	Depth	Memory + Disk	Messages	Channels
✖ ubuntu:4151	0	0 + 0	8	2
Total:	0	0 + 0	8	2

Channel Message Queues

Channel	Depth	Memory + Disk	In-Flight	Deferred	Requeued	Timed Out	Messages	Connections
chat	8	8 + 0	0	0	0	0	8	0
ghost	8	8 + 0	0	0	0	0	8	0

File Edit View Search Terminal Help

Ok

File Edit View Search Terminal Help

2014/04/21 19:13:49 NSQD: querying http://ubuntu:4151

2014/04/21 19:18:46 LOOKUPD: querying http://127.0.0.0

2014/04/21 19:18:48 LOOKUPD: querying http://127.0.0.0

2014/04/21 19:18:48 NSQD: querying http://ubuntu:4151

2014/04/21 19:18:51 LOOKUPD: querying http://127.0.0.0

2014/04/21 19:19:00 LOOKUPD: querying http://127.0.0.0

File Edit View Search Terminal Help

2014/04/21 19:14:13 [ubuntu:4150/test/nsq\_to\_file] h

2014/04/21 19:14:43 LOOKUPD: querying http://127.0.0.0

2014/04/21 19:14:43 [ubuntu:4150/test/nsq\_to\_file] h

2014/04/21 19:15:13 [ubuntu:4150/test/nsq\_to\_file] h

2014/04/21 19:15:43 [ubuntu:4150/test/nsq\_to\_file] h

2014/04/21 19:15:43 LOOKUPD: querying http://127.0.0.0

File Edit View Search Terminal Help

2014/04/21 19:14:42 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:14:57 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:15:12 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:15:27 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:15:42 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:15:57 LOOKUPD(127.0.0.1:4160): sending

2014/04/21 19:16:12 LOOKUPD(127.0.0.1:4160): sending

File Edit View Search Terminal Help

2014/04/21 19:14:42 CLIENT(127.0.0.1:32955): pinged

2014/04/21 19:14:57 CLIENT(127.0.0.1:32955): pinged

2014/04/21 19:15:12 CLIENT(127.0.0.1:32955): pinged

2014/04/21 19:15:27 CLIENT(127.0.0.1:32955): pinged

2014/04/21 19:15:42 CLIENT(127.0.0.1:32955): pinged

2014/04/21 19:15:57 CLIENT(127.0.0.1:32955): pinged


2014/04/21 19:16:12 CLIENT(127.0.0.1:32955): pinged

# NSQ

No SPF

drone.io


[Home](#)[Settings](#)[Help](#)




[New Repository](#)[Logout](#)


hpc happypancake

[Commits](#)[Settings](#)





**faf9a2** about 16 hours ago  
Rename MemberOnline to MemberSeen event







**e4d967** about 16 hours ago  
Optimize removal







**3e2d46** about 16 hours ago  
Fix old entry removal from timeline







**66285e** about 16 hours ago  
Adds online list \* Online list can be seen at: `/online` \*...







**89a1aa** a day ago  
Display own member info for review alerts







**2b6c41** 3 days ago  
bus: boom on errors in node handlers





**400-12** ...





master

# Интеграция

```
11func (x *denormalizer) Name() string { return "member-denormalize" }
10
9func (x *denormalizer) HandleNodeEvent(e hpc.Event) (err error) {
8    switch e := e.(type) {
7        case *hpc.RegistrationRequested:
6            r := newRecord()
5            r.Birthday = e.Birthday
4            r.Gender = e.Gender
3            r.Nickname = e.Nickname
2            r.State = ProfileGhost
1            x.store.Set(e.MemberId, r)
28    x.

    HandleNodeEvent
    case Name      pproved:
    r pub         e(e.MemberId, func(r *record) {
    store         t
    })
    if r.State == ProfileGhost {
        return
    }
    out := &hpc.ProfileTextUpdated{
        EventId: hpc.NewId(),
        MemberId: e.MemberId,
        Text:     e.Text,
        Birthday: r.Birthday,
        Gender:   r.Gender,
    }
    x.pub.MustPublish(out)

    case *hpc.RegistrationApproved:
    r := x.store.Update(e.MemberId, func(r *record) {
21        r.State = ProfileApproved
22        r.Birthday = e.Birthday
23        r.Nickname = e.Name
24        r.Gender = e.Gender
25        r.Email = e.Email
26    })
26    // TODO: publish all pending photo updates
27    out := &hpc.ProfileCreated{
28        EventId: hpc.NewId(),
29        MemberId: e.MemberId,
    }
}

9e user in the system.
8// Member has yet to be approved before he will show up
to the other people
7type RegistrationRequested struct {
6    EventId      Id
5    MemberId     Id
4    Nickname     string
3    Gender       Gender
2    Birthday     Birthday
1    Email        string
118 City         string
1    LookingForGender Gender
2    LookingForAgeRange AgeRange
3}
4

-:--- event.go 20% of 14k (118,0) <N> Git-master
Watching path media
Watching path member
Watching path nav
Watching path news
Watching path online
Watching path profile
Watching path register
Watching path review
Watching path static
Watching path static/css
Watching path static/img
Watching path static/js
Watching path views
Watching path web
go test ./member
ok      github.com/happypancake/hpc/member 0.041s
PASS (0.62 seconds)
go test ./member
ok      github.com/happypancake/hpc/member 0.033s
PASS (0.56 seconds)
go test ./member
ok      github.com/happypancake/hpc/member 0.035s
PASS (0.57 seconds)

-:***- denormalize.go<member> 15% of 1.9k (28,10) <I> Git:master (Go MRev com U:***- *terminal* Bot of 115k (2049,0) <I> (Term: ch
func HandleNodeEvent(e hpc.Event) (err error)
```

# IDE

Emacs / Vim / Sublime

# Вопросы?

<http://abdullin.com/long/happypancake>