

Эволюционный дизайн

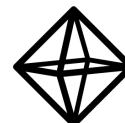
От любви до ненависти один шаг



@tolkv



@lavcraft



octoberry





@aatarasoff



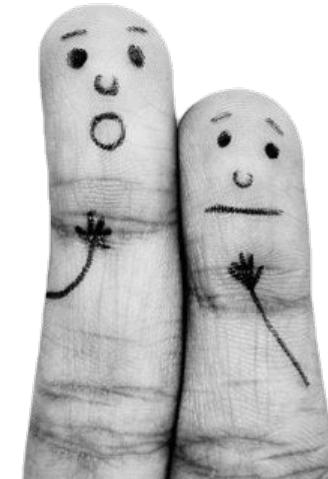
@aatarasoff



</> DEVELOPERBLOG.INFO

DISCLAIMER

No warranty guarantee





Эволюционный дизайн

От любви до ненависти один шаг



- делается с первого раза
- не меняется годами
- не зависит от людей
- не зависит от процессов

У нас конечно же так

Потому что



У нас конечно же так

Потому что

- мы работаем в лучшем банке



У нас конечно же так

Потому что

- мы работаем в лучшем банке
- и все остальные банки завидуют нам



У нас конечно же так

Потому что

- мы работаем в лучшем банке
- и все остальные банки завидуют нам
- ...
- Стойте, кажется это про другое



Закон “Почему”

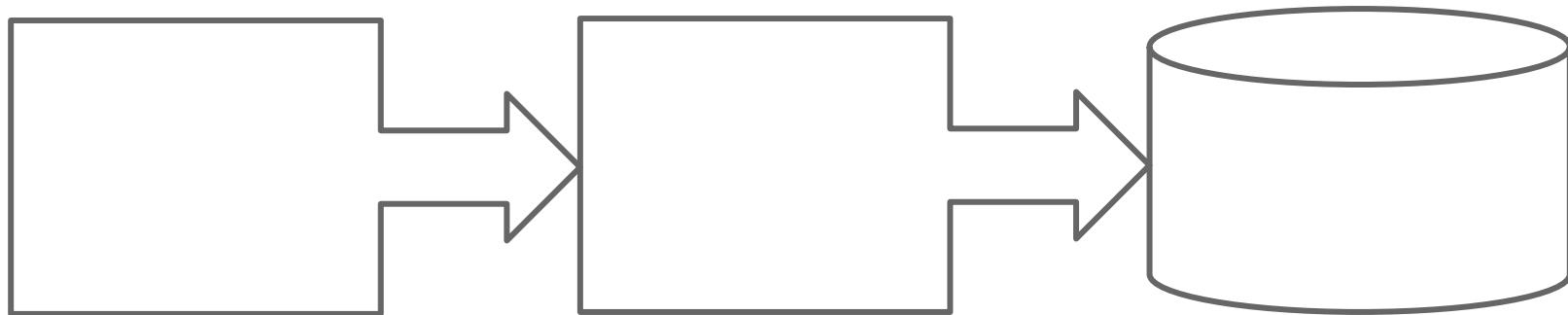
“Почему JQuery?”



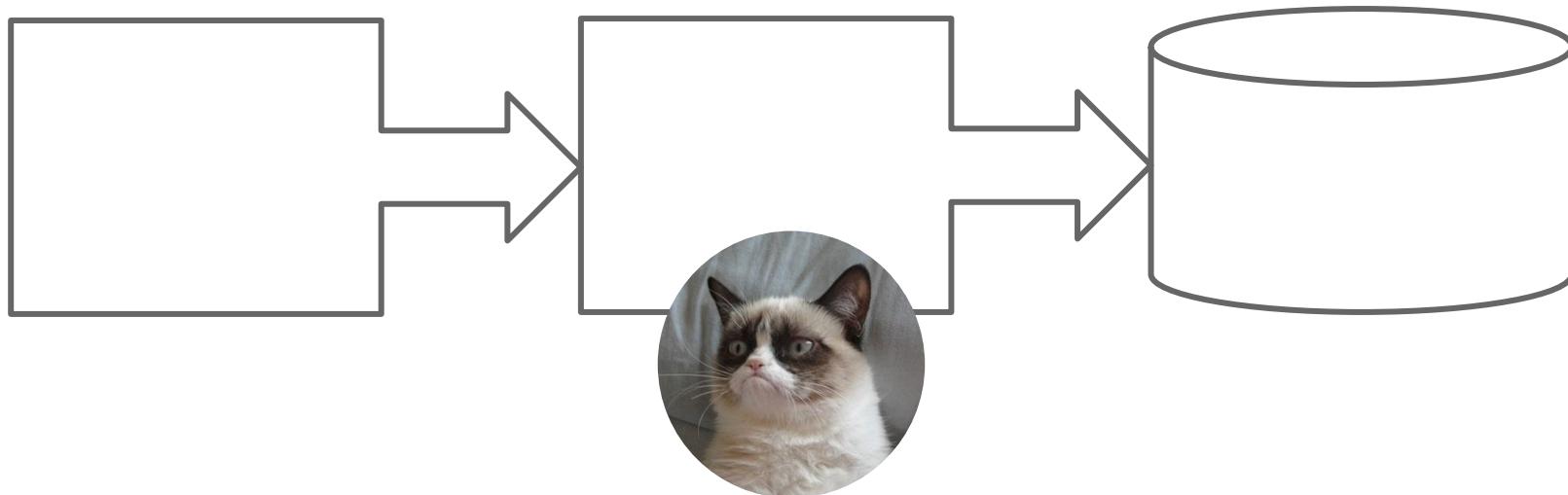


Точка зрения: архитектор

Работа идеального архитектора



Работа идеального архитектора

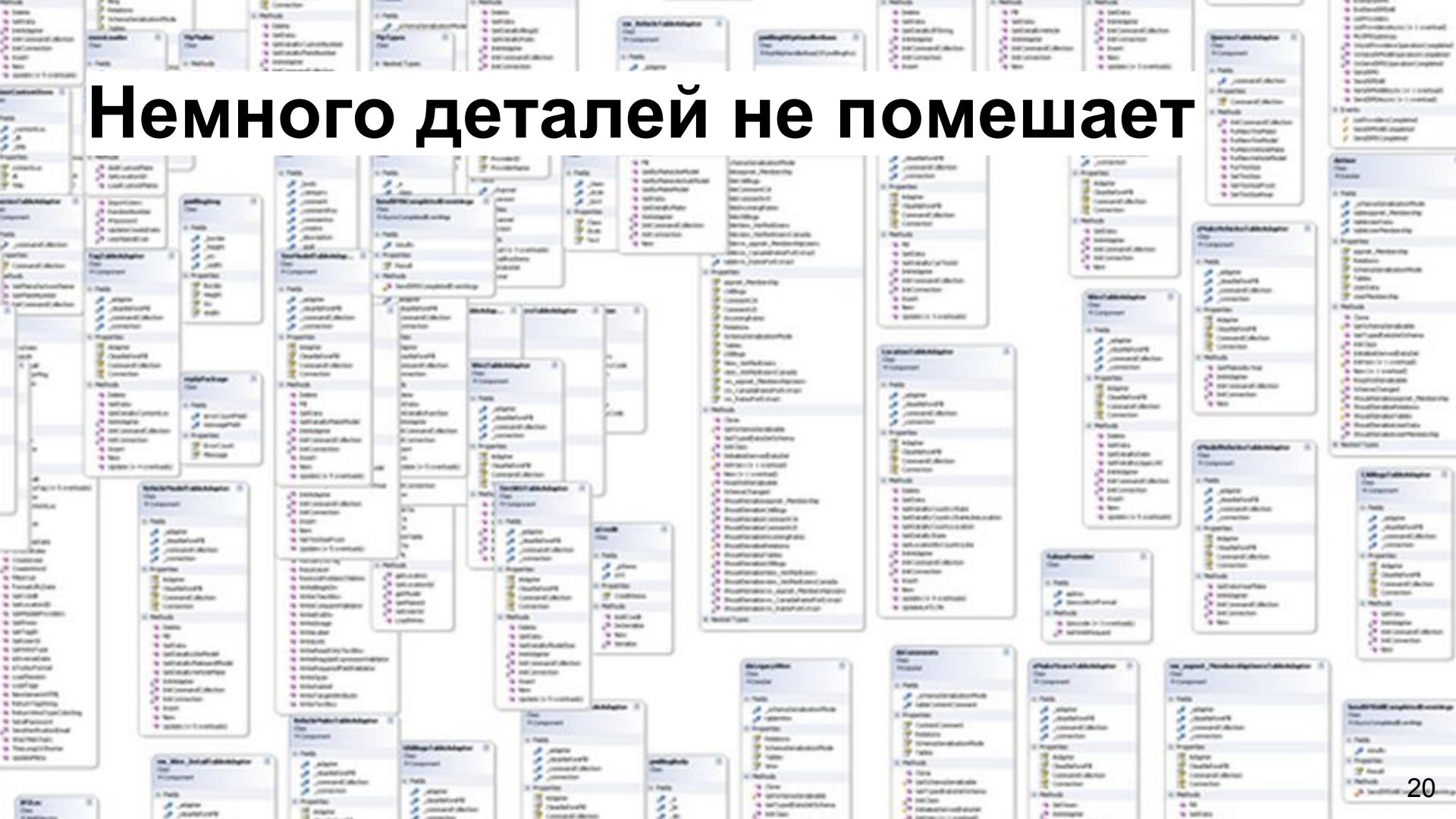


Работа идеального архитектора

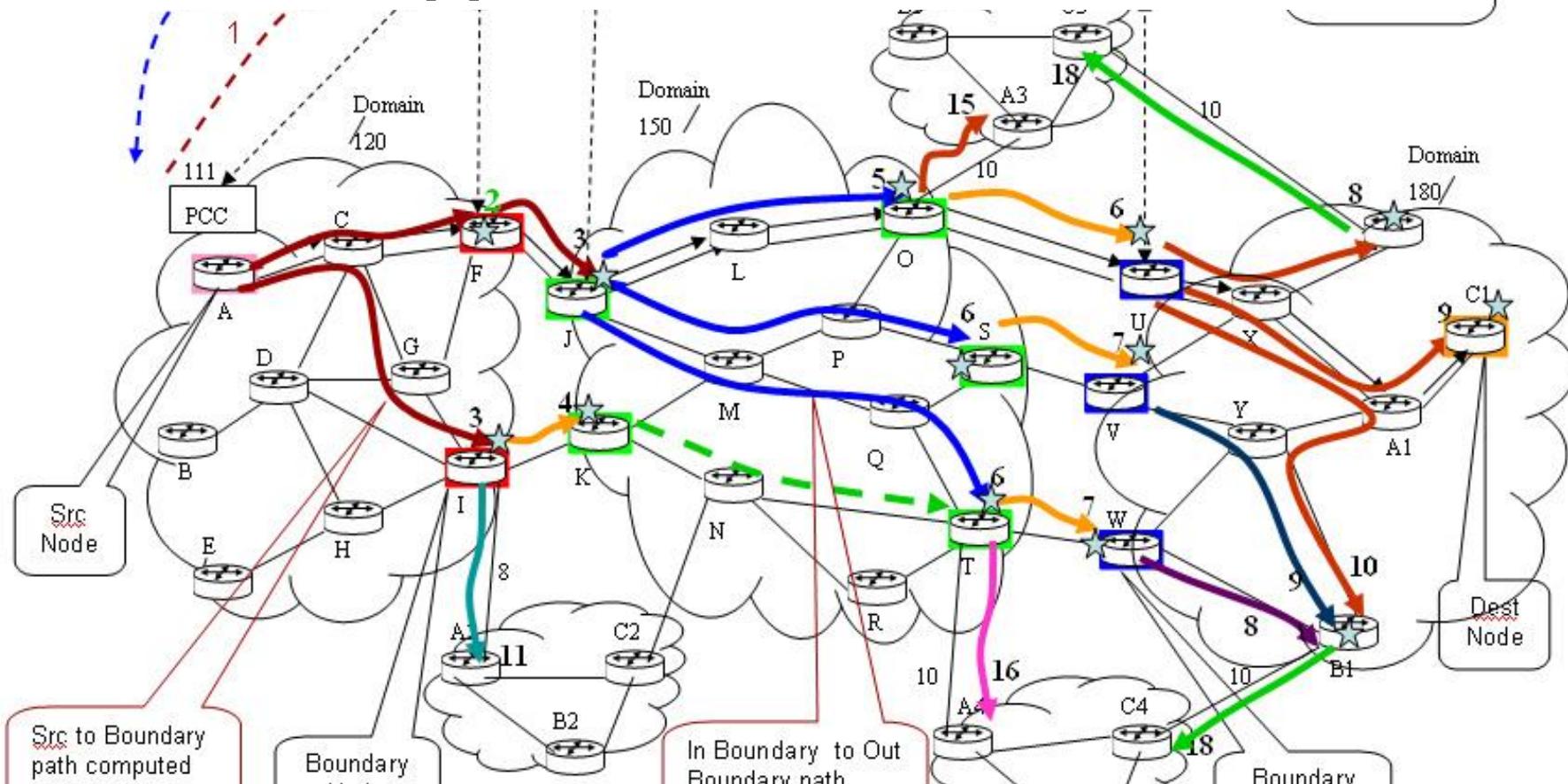


Точка зрения: разработчик

Немного деталей не помешает



Немного деталей не помешает



Архитектура



Архитектура

Что это?



Что же такое архитектура?

$A = F(Rq)$?

Что же такое архитектура?

$$A = F(Rq)$$

$$A = F(FRq, NFRq)$$

Что же такое архитектура?

$A = F(Rq)$

$A = F(FRq, NFRq)$

$A = F(FRq, NFRq, ?)$



https://www.youtube.com/watch?v=_Kex5hwGE-w

$$A = F(FRq, NFRq, ?)$$

Yo yoman										Zn zipkin
Lz lazybones	Ss start.spring.io	Ja java					Eu eureka	Ed etcd	Sl sleuth	
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes		
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos		
So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing		

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing	
------------------	-----------------------------	-------------------------------------	----------------------------	------------------	---------------------------	-------------------------	-------------------------	------------------------------	--

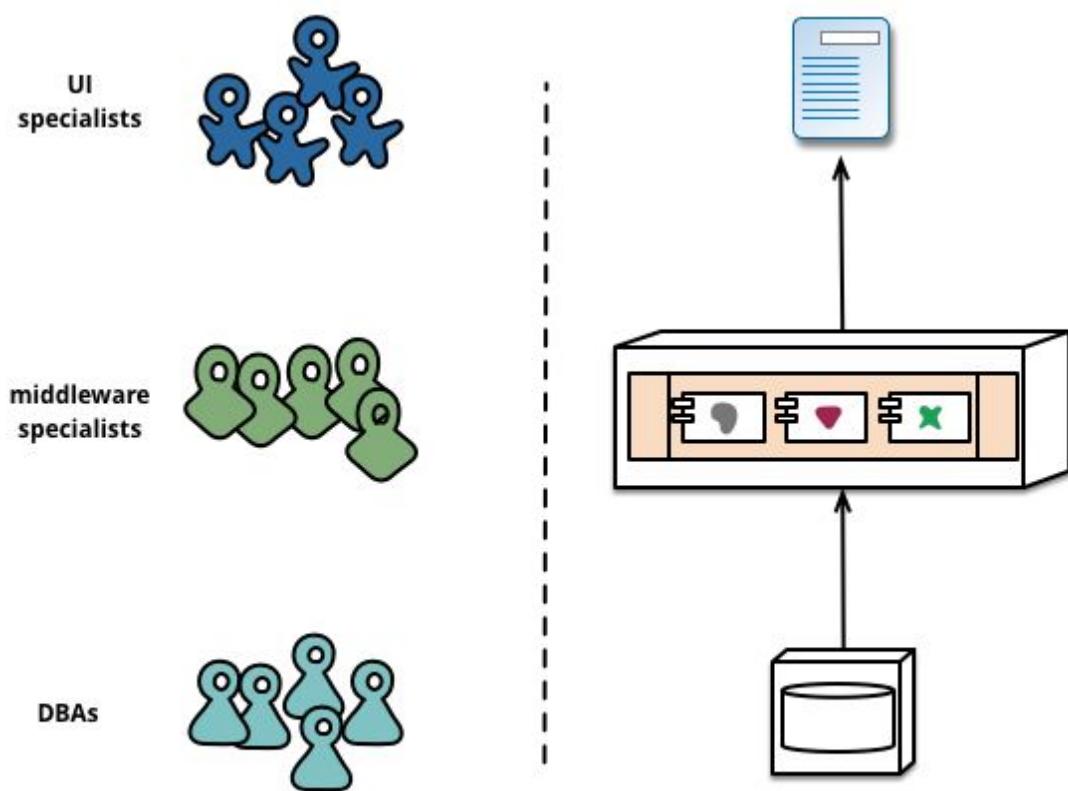
Yo yoman									Zn zipkin
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	Sl sleuth	
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes	
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos	

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing

Yo yoman							Zn zipkin	
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	Sl sleuth
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing
------------------	-----------------------------	-------------------------------------	----------------------------	------------------	---------------------------	-------------------------	-------------------------	------------------------------

Закон Конвея





Вход в интернет-банк

Логин

Пароль

Запомнить логин

[Регистрация](#)

[Забыли пароль?](#)



Впервые в интернет-банке? [Пройдите регистрацию](#)

Возможности

- [Возможности интернет-банка](#)
- [Видео-презентации](#)
- [Часто задаваемые вопросы](#)

Безопасность

- [Меры предосторожности](#)
- [Защита от мошенников](#)
- [Использование Альфа-Ключа](#)



Мобильный банк



Альфа-Мобайл

Простой и удобный банк в вашем телефоне всегда под рукой

- iOS Windows Phone
 Android Другие системы

Новости

2014-12-29

[Пополнить счет клиентов "Альфа-Форекс" через Интернет-банк "Альфа-Клик" стало проще и удобнее](#)

2014-12-24

[Жители Волгограда теперь могут оплачивать коммунальные услуги через Интернет-банк "Альфа-Клик"](#)

2014-11-28

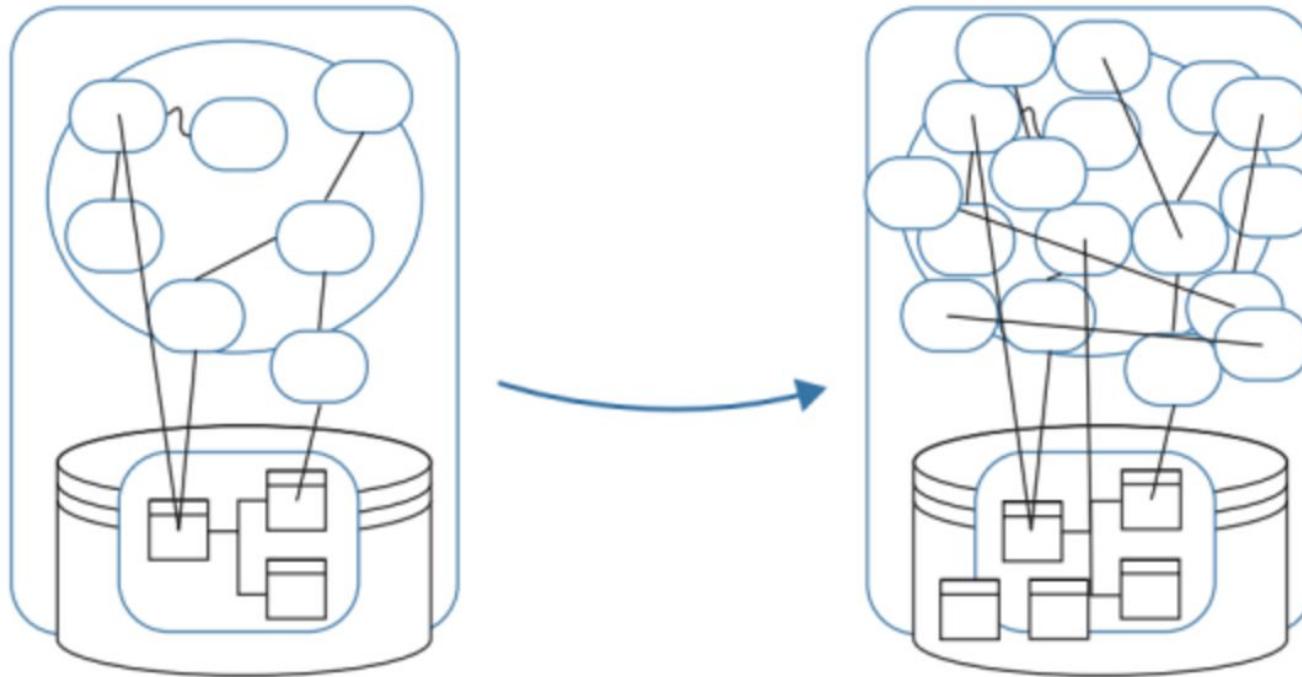
[Погашать кредиты в Альфа-Банке просто и удобно](#)

2014-11-18

[Оплата услуг ЖКУ без комиссии в Интернет-банке "Альфа-Клик"](#)

[Архив новостей](#)

Big Ball of Mud



Проблемы архитектуры

- Сильная связанность между модулями
- Слабое тестовое прикрытие
- Регрессионная спираль смерти
 - частично решалась Selenium-тестами
 - но это дорого

Counter-Strike Cats

SOLID



Counter-Strike Cats

GOF



SOLID



TDD





Some Big App

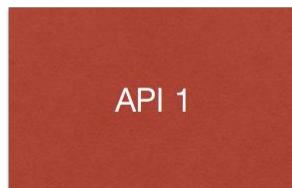
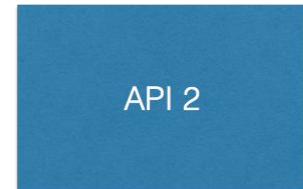
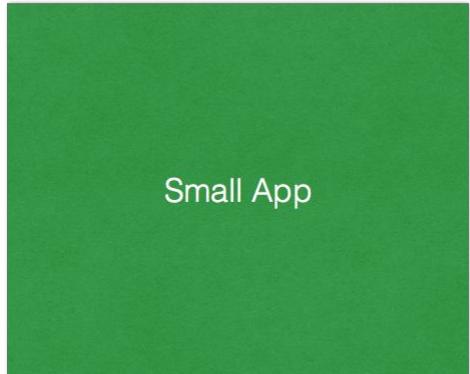
Some Big App

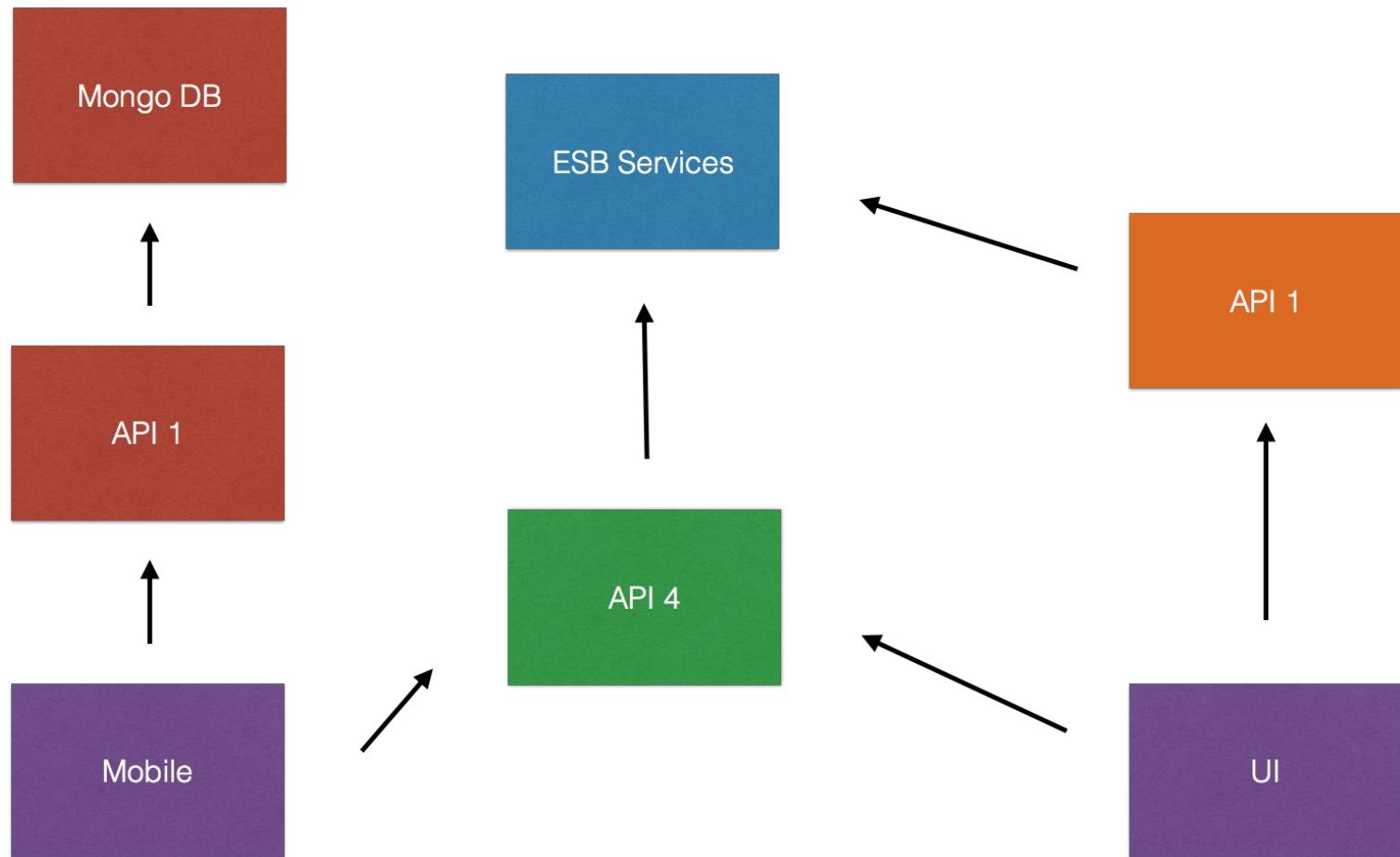


Small App 1

Small App 2

Small App 3







12 April 1996

Первое упоминание SOA

Принципы SOA

1. Standardized service contract
2. Loose coupling
3. Encapsulation
4. Reusability
5. Autonomy
6. Statelessness
7. Discoverability

Принципы SOA

1. Standardized service contract

2. Loose coupling

3. Encapsulation

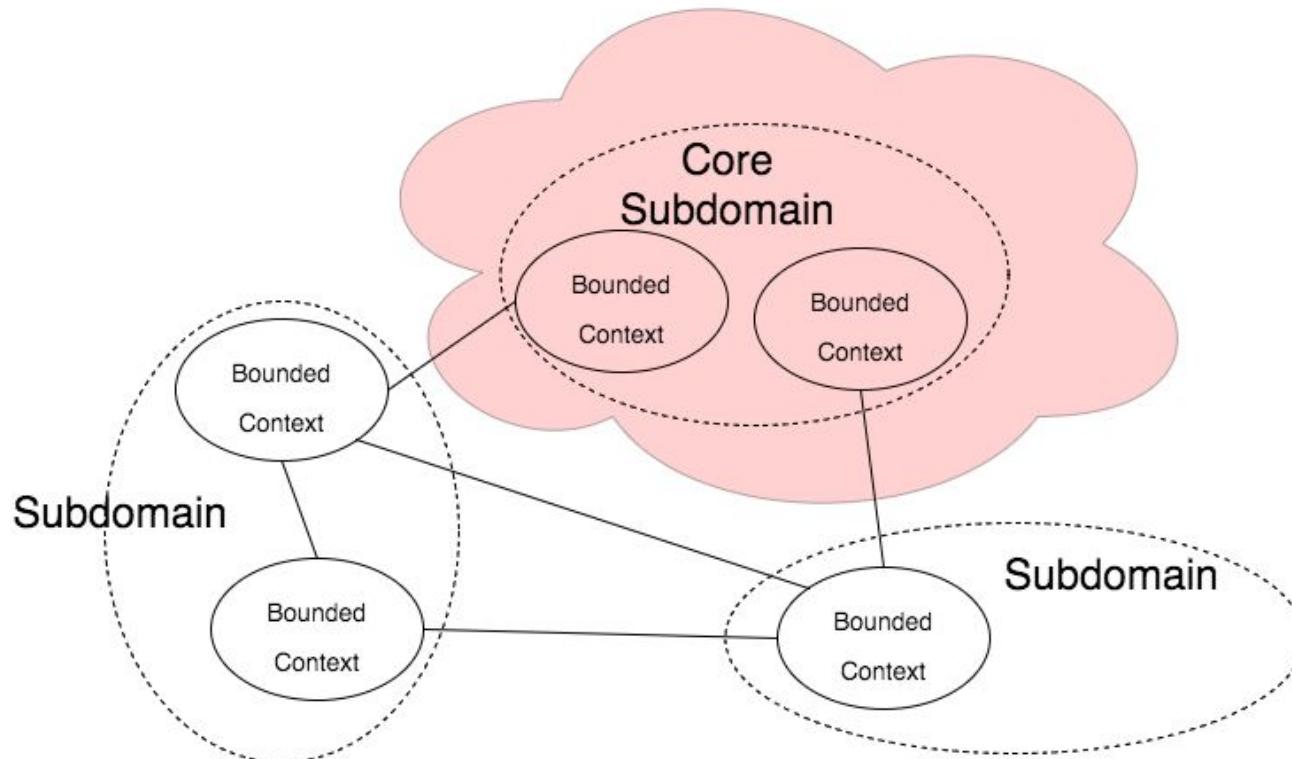
4. Reusability

5. Autonomy

6. Statelessness

7. Discoverability

Domain-Driven Design





5 January 2009 SOA is Dead

<http://apsblog.burtongroup.com/2009/01/soa-is-dead-long-live-services.html>

SOA != SOAP (WS-*)



So

SOA

Dd

DDD

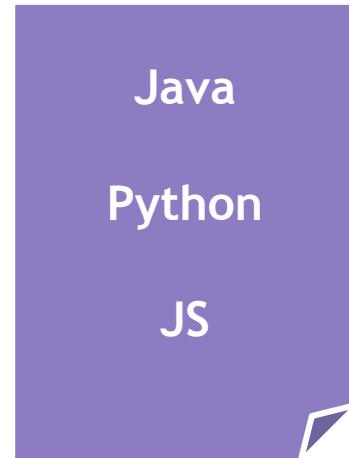
Co

Conway's Law

Проблема эволюции

- Начинали с одного проекта
 - одна команда
 - ui + три сервиса
- Более 10-ти однотипных проектов
 - несколько команд
 - десятки сервисов
 - технологически одинаковые



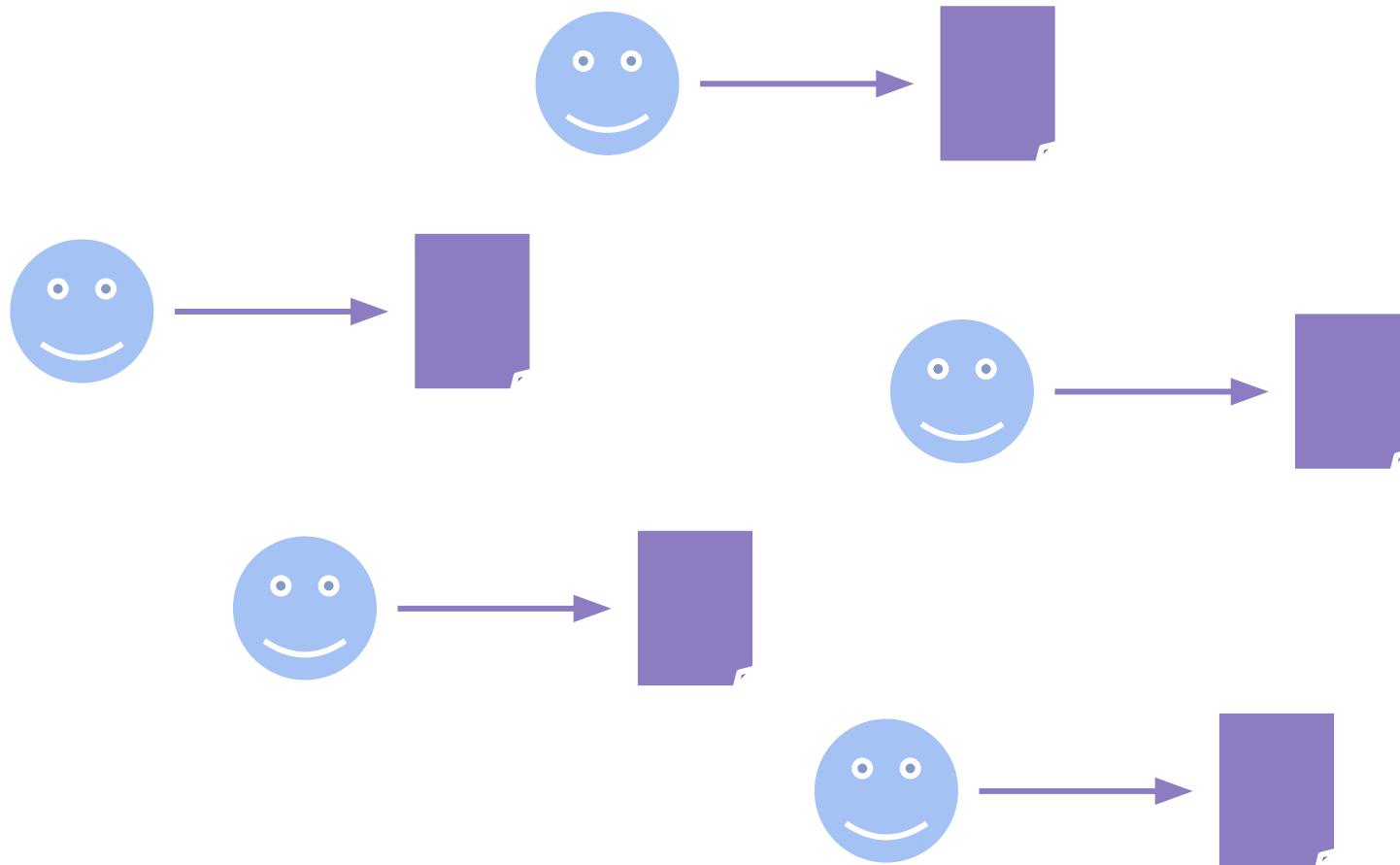


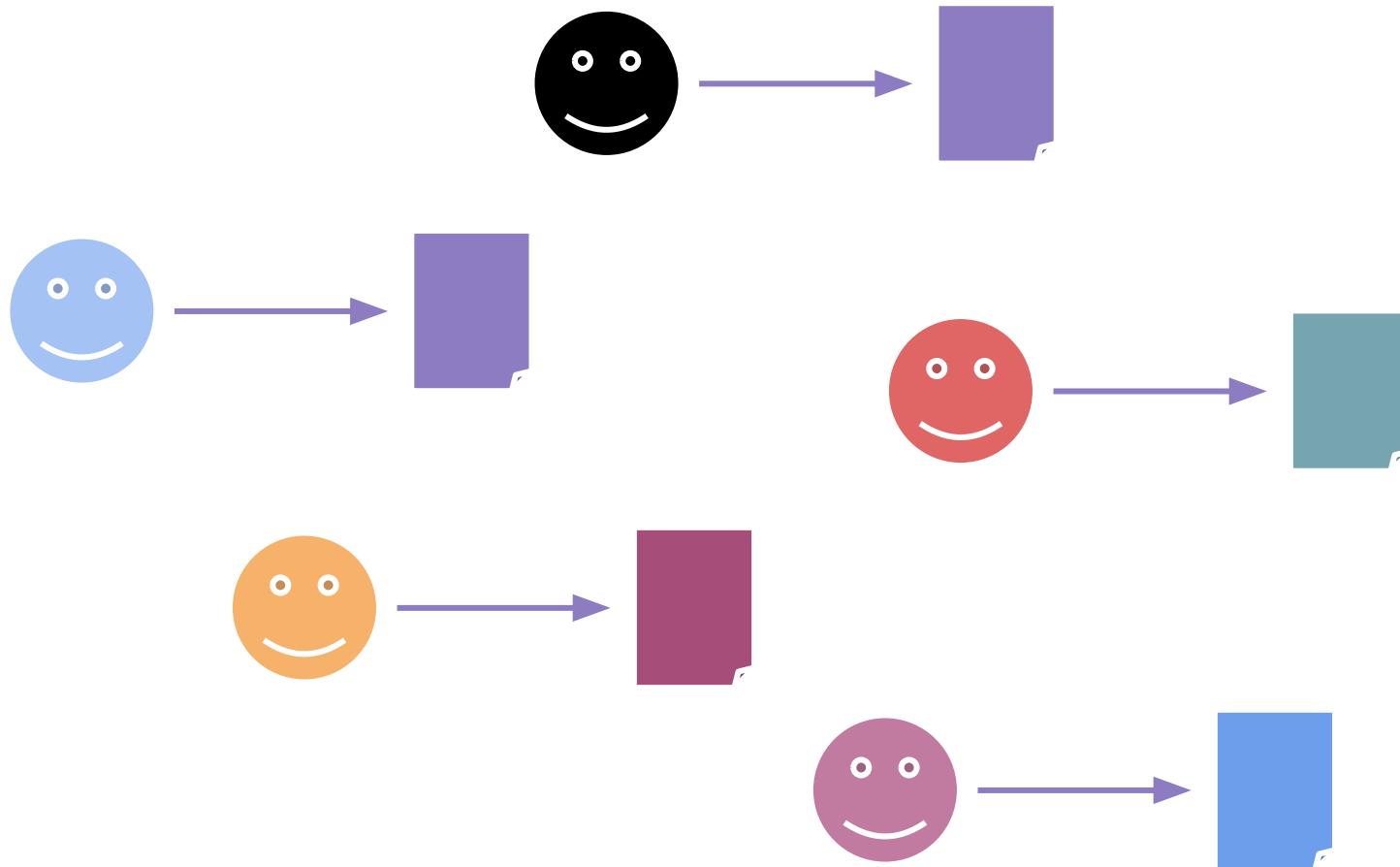
- Spring Boot/Spring Cloud
- Ratpack
- Dropwizard
- Vert.x
- Restlet
- Spark
- KumuluzEE



- Spring Boot/Spring Cloud
- Ratpack
- Dropwizard
- Vert.x
- Restlet
- Spark
- KumuluzEE

Выбирайте то,
что больше
нравится
/
в чем есть
экспертиза





Принцип LSD

- L языков программирования
- S среднее число фреймворков на язык
- D типов источников данных

complexity = L * S * D

Немного LSD для вас

- **три** языка программирования
- **два** в среднем фреймворка на язык
- **семь** типов источников данных
 - legacy WS, mongo db
 - OLTP, OLAP
 - elasticsearch, neo4j
 - Мишкина база %)

complexity = 3 * 2 * 7 = 42 (!)



Чем нельзя пожертвовать?

$\min (L * S * D) \rightarrow ?$



$\min(L * S * D) \rightarrow \max(D)$



L = Java 18+

S = Spring Boot

D = ∞

Мы сами нарушаєм правила

L = Java

L += Python

L += JavaScript

Ja

java

Sb

spring boot

So

SOA

Ls

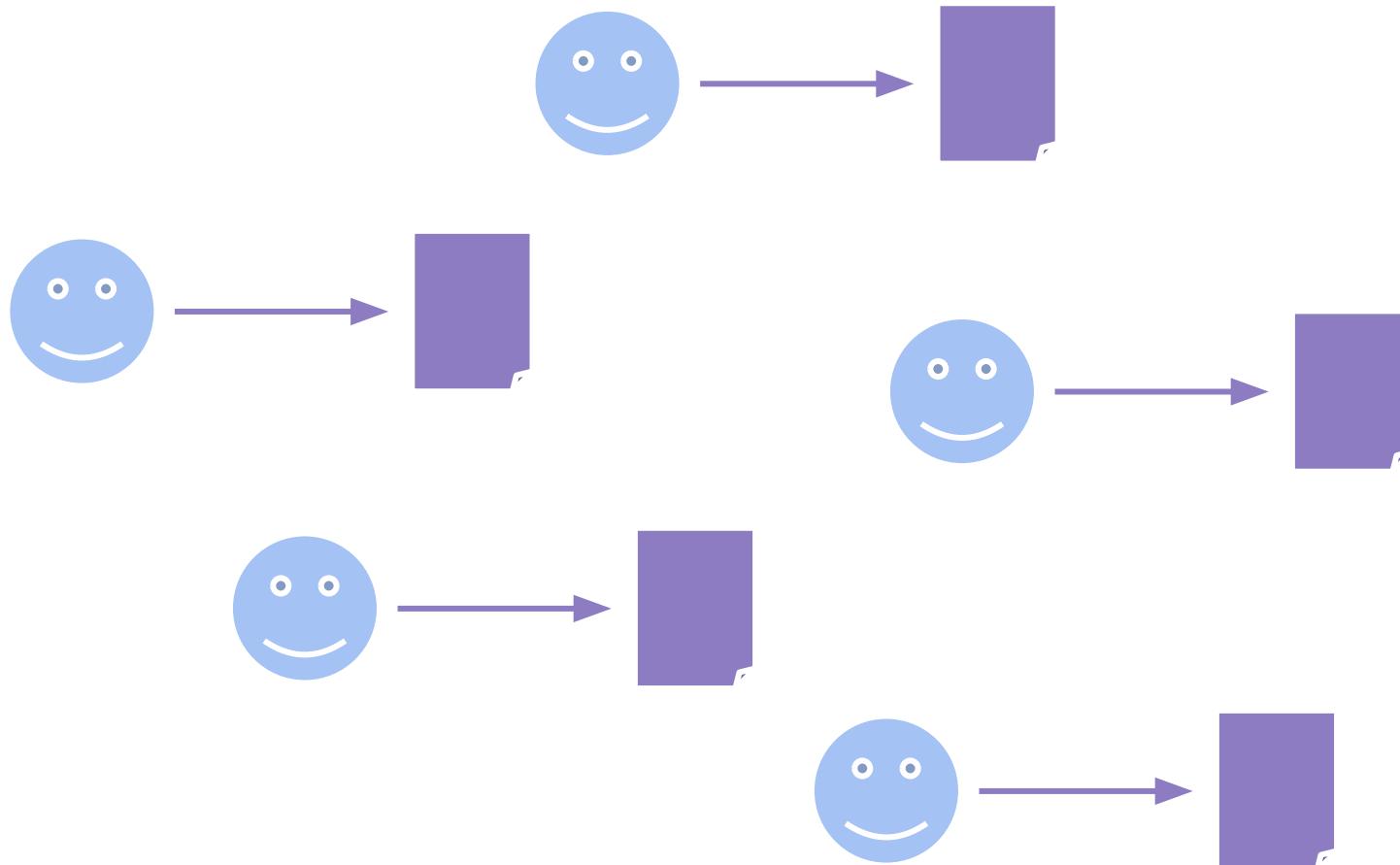
LSD principle

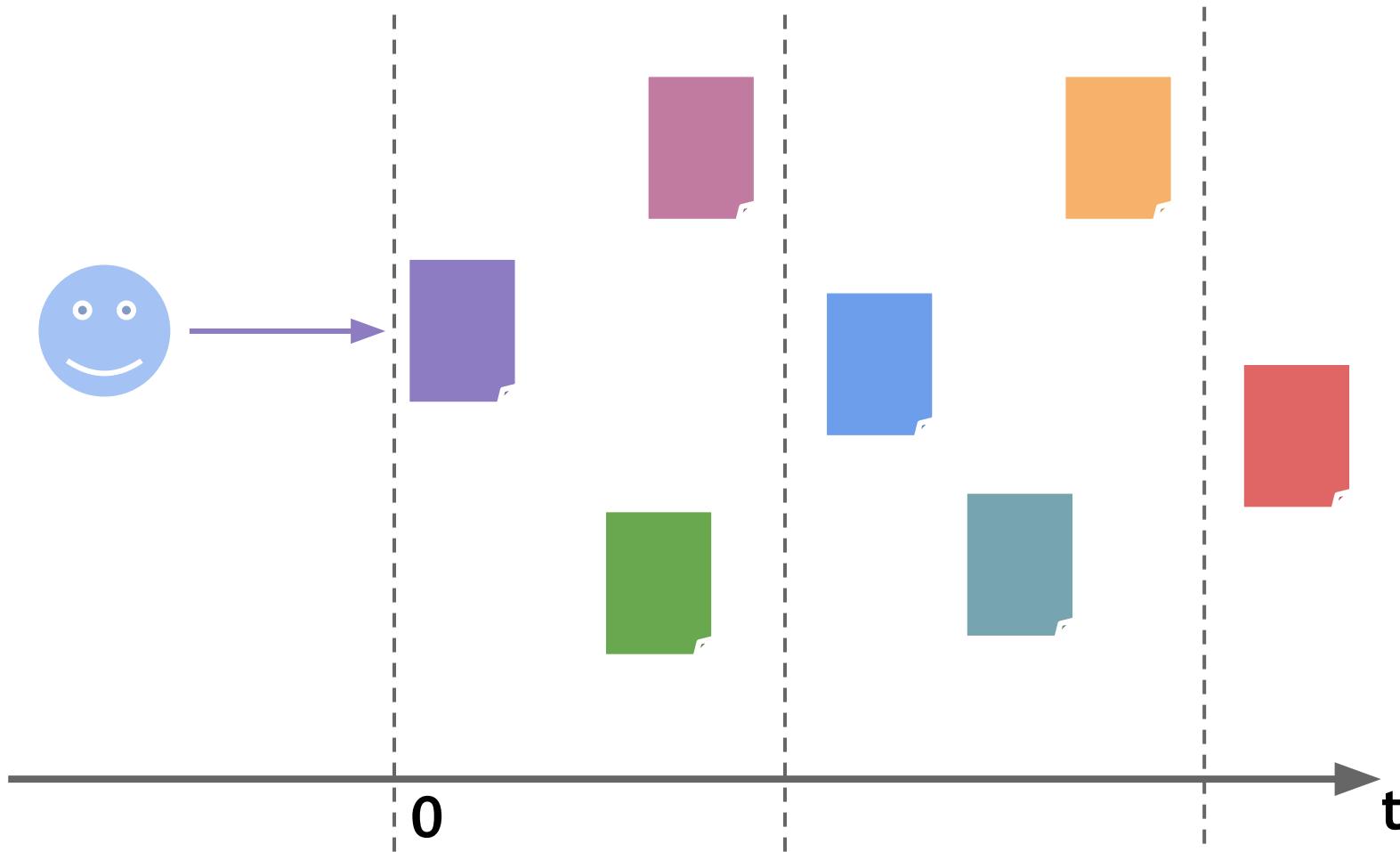
Dd

DDD

Co

Conway's Law

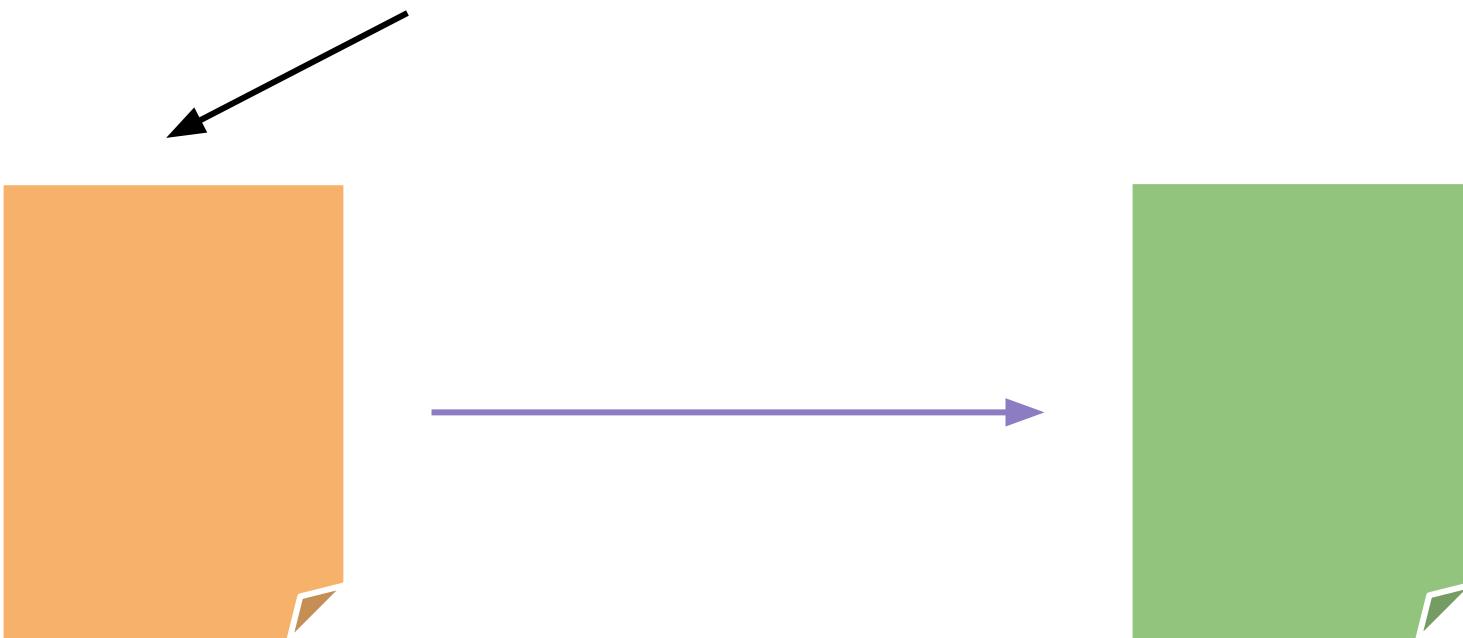






git clone <golden_service_repo>

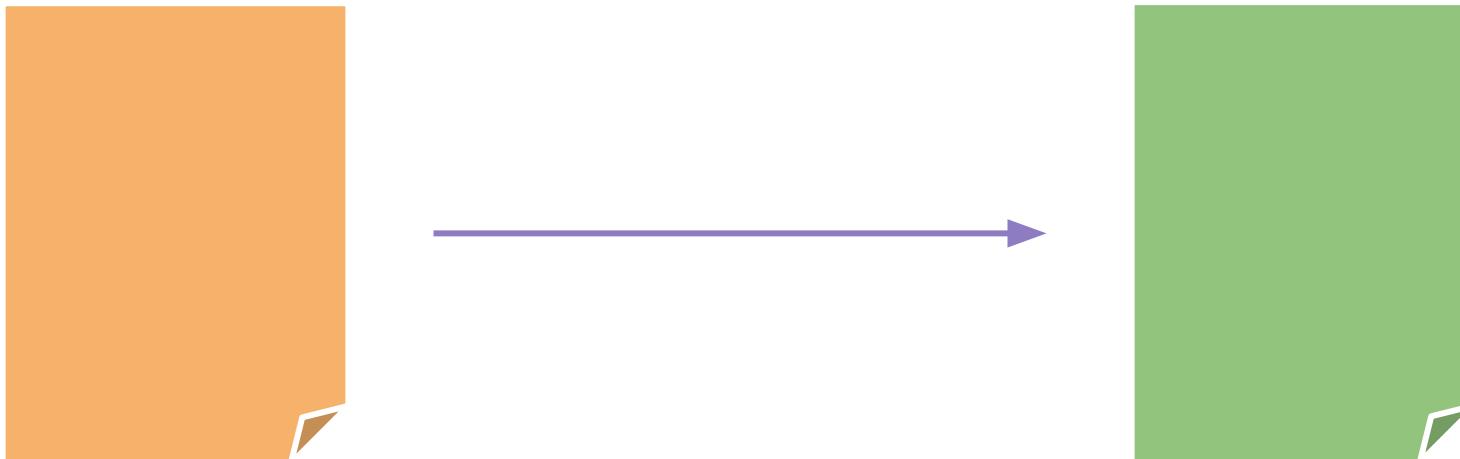
Код устарел



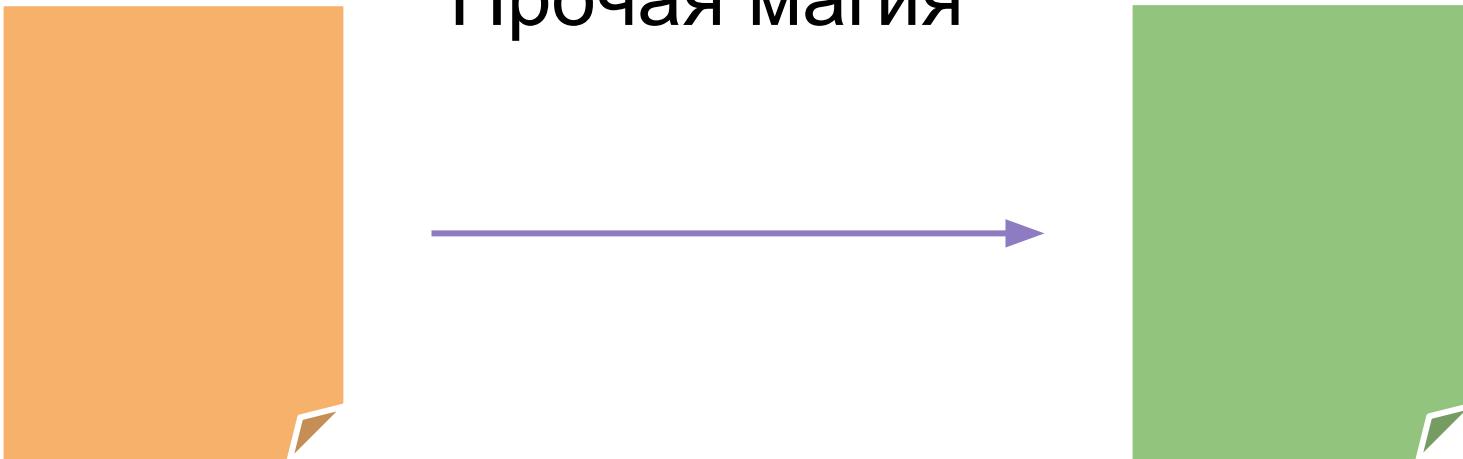
Код устарел
Нужно переименовать классы



Код устарел
Нужно переименовать классы
Нужно удалить неиспользуемое



Код устарел
Нужно переименовать классы
Нужно удалить неиспользуемое
Прочая магия

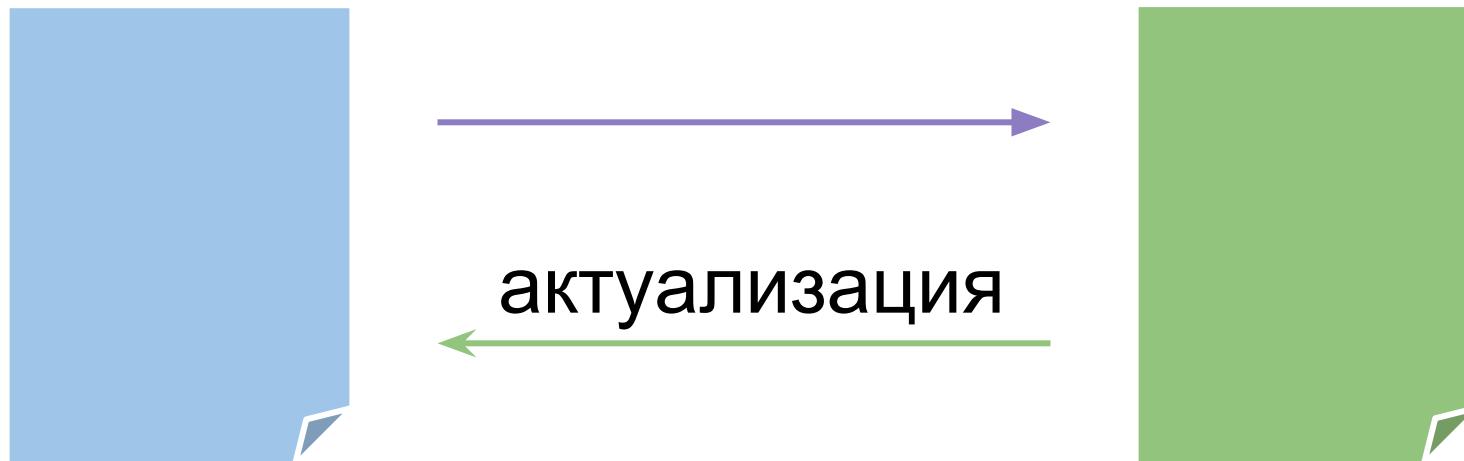


Время, ошибки, фрагментация



git clone <template_repo>

Нужно переименовать классы
Нужно кастомизировать руками



Время



lazybones / start.spring.io / yeoman

```
lazybones create api 1.0.1 service_name
```

```
~home > lazybones create api 0.0.1 rent-service
```

```
~home > lazybones create api 0.0.1 rent-service
Creating project from template api 0.0.1 in 'rent-service'
Define value for 'group' [ru.joker]:
```

```
~home > lazybones create api 0.0.1 rent-service
Creating project from template api 0.0.1 in 'rent-service'
Define value for 'group' [ru.joker]:
Define value for 'version' [0.0.1]:
```

```
~home > lazybones create api 0.0.1 rent-service
Creating project from template api 0.0.1 in 'rent-service'
Define value for 'group' [ru.joker]:
Define value for 'version' [0.0.1]:
```

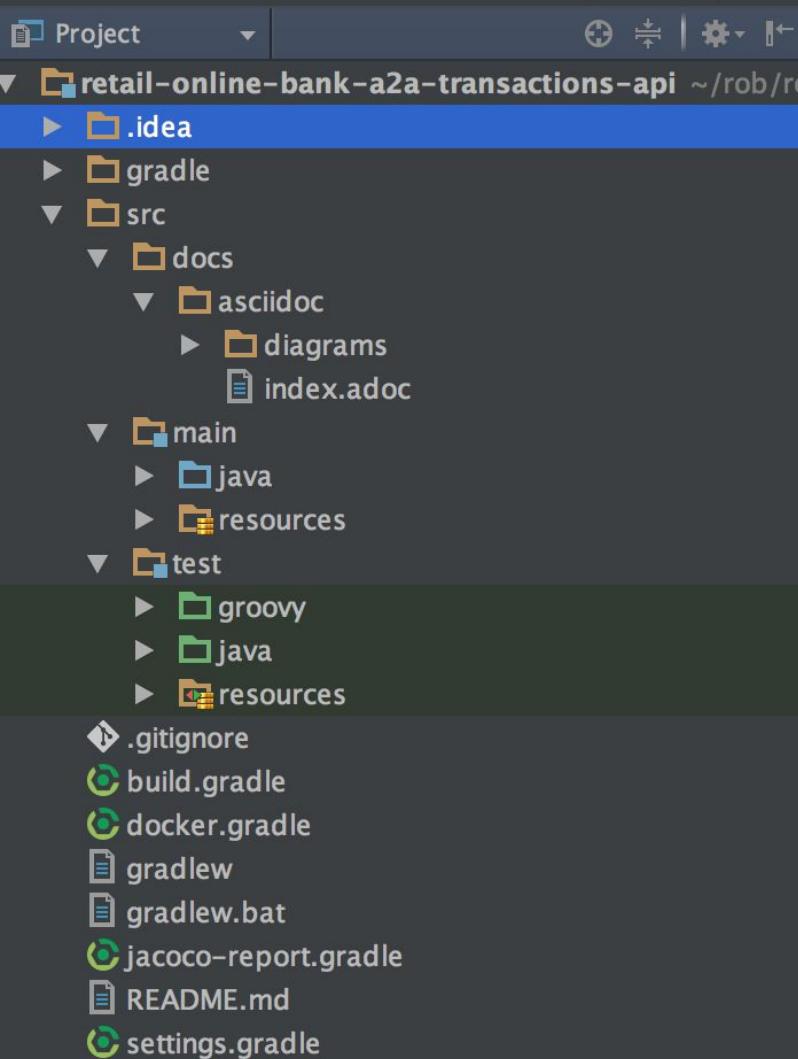
```
srv1
├── srv2
└── srv3
logging
sleuth
```

```
Define value for 'dependencies' [logging,sleuth]:
```

```
~home > lazybones create api 0.0.1 rent-service
Creating project from template api 0.0.1 in 'rent-service'
Define value for 'group' [ru.joker]:
Define value for 'version' [0.0.1]:
```

```
srv1
├─ srv2
└─ srv3
logging
sleuth
```

```
Define value for 'dependencies' [logging,sleuth]:
Project created for rent-service!
```



Search Everywhere Double ↑

Go to File ⇧⌘O

Recent Files ⌘E

Navigation Bar ⌘↑

Drop files here to open

Yo		
yeoman		
Lz	Ss	Ja
lazybones	start.spring.io	java

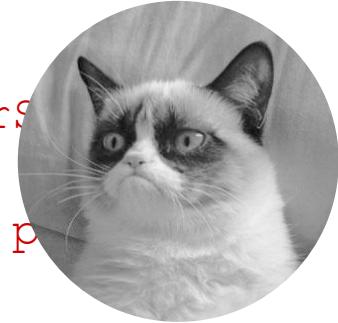
So	Ls	Dd	Co
SOA	LSD principle	DDD	Conway's Law

DRY off



```
TServerTransport serverTransport = new TServerS
        new
InetSocketAddress(InetAddress.getLocalHost(), p
TProcessor processor = new
TInsuranceService.Processor<>(
    //business value here
);

server = new TSimpleServer(
    new
TServer.Args(serverTransport).processor(processor));
server.serve();
```





```
TSocket transport = new TSocket(host, port);
transport.open();

TBinaryProtocol tBinaryProtocol = new
TBinaryProtocol(transport);
TInsuranceService.Client client =
    new TInsuranceService.Client(tBinaryProtocol);

perform(client); //business value here

transport.close();
```



@ThriftController

```
public class InsuranceController implements  
TInsuranceService.Iface {  
    @Override  
    public void someMethod(TInsurance insurance) {  
        //business value here  
    }  
}
```



```
@ThriftClient(serviceId = "insurance-service")
TInsuranceService.Client client;

...
perform(client); //business value here
...
```

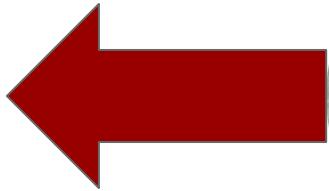


shared libs
utils
helpers



smart libs
starters
launchers

Старт приложения



Рабочее приложение



```
@ThriftClient(serviceId = "insurance-service")
TInsuranceService.Client client;

TInsuranceService.Client client =
    Creator.create(TInsuranceService.Client.class,
                  "insurance-service");

...
perform(client); //business value here
...
```

```
@Getter // generate getters
@Setter // generate setters
@Aspect // we are an aspect
@ToString // generate toString()
@EnableWs // SOAP is so enterprisy, we definitely need it
@Endpoint // Seriously, just read above
@EnableWebMvc // we want MVC
@EnableCaching // and we want to cache stuff
@Configuration // this class can configure itself
@RestController // we want some REST
@XmlRootElement // this component is marshallable
@EnableWebSocket // we want web socket, it's so new-generation
@RedisHash("cat") // this class is an entity saved in redis
@EnableScheduling // we want scheduled tasks
@EnableWebSecurity // and some built-in security
@NoArgsConstructorConstructor // generate no args constructor
@ContextConfiguration // we want context configuration for unit testing
@SpringBootApplication // this is a Sprint Boot application
@Accessors(chain = true) // getters/setters are chained (ala jQuery)
@EnableAspectJAutoProxy // we want AspectJ auto proxy
@EnableAutoConfiguration // and auto configuration
@EnableRedisRepositories // since it is an entity we want to enable spring data repositories for redis
@EnableWebSocketMessageBroker // we want a broker for web socket messages
```



smart libs/~~shared~~ libs

documentation → smart documentation

Not smart

= This is main documentation

This document describes how to be the most fundamental and important document in the world of documents

...

COPY-PASTE documentation from another document

...



Not so smart

= This is main documentation

This document describes how to be the most fundamental and important document in the world of documents

`include::https://raw.github.com/asciidoc/asciidoc/master/Gemfile[]`

`include::::/other.adoc[]`

`include:::/home/tolkv/git/docs-0/superdoc.adoc[]`



Really smart

= This is main documentation

This document describes how to be the most fundamental and important document in the world of documents

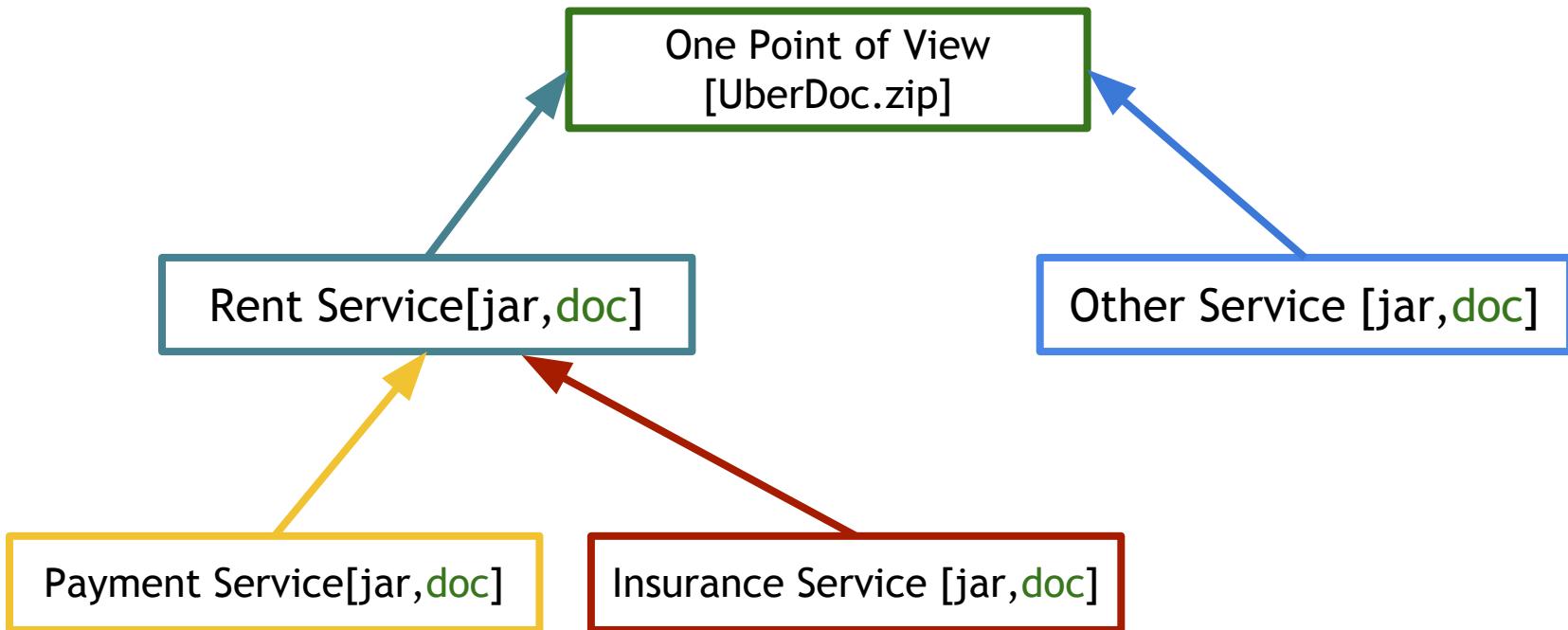
```
include::https://raw.github.com/asciidoc/asciidoc/master/Gemfile[]
```

```
include::gradle://gradle-advanced:service-with-deps:1.0/deps.adoc[]
```

```
include::gradle://:service/doc.adoc[]
```



Агрегация информации



Парадокс централизации

Чтобы эффективно разрабатывать **распределённые** приложения, нам нужны очень хорошие **централизованные** библиотеки и инструменты

Например: логирование, health-checking, метрики, обработка типовых ошибок, автодокументирование

Парадокс централизации

Чтобы эффективно разрабатывать **распределённые** приложения, нам нужны очень хорошие **централизованные** библиотеки и инструменты

Но: не выносите бизнес-логику или доменные объекты!
Не размывайте бизнес-контекст вашего API

Yo
yoman

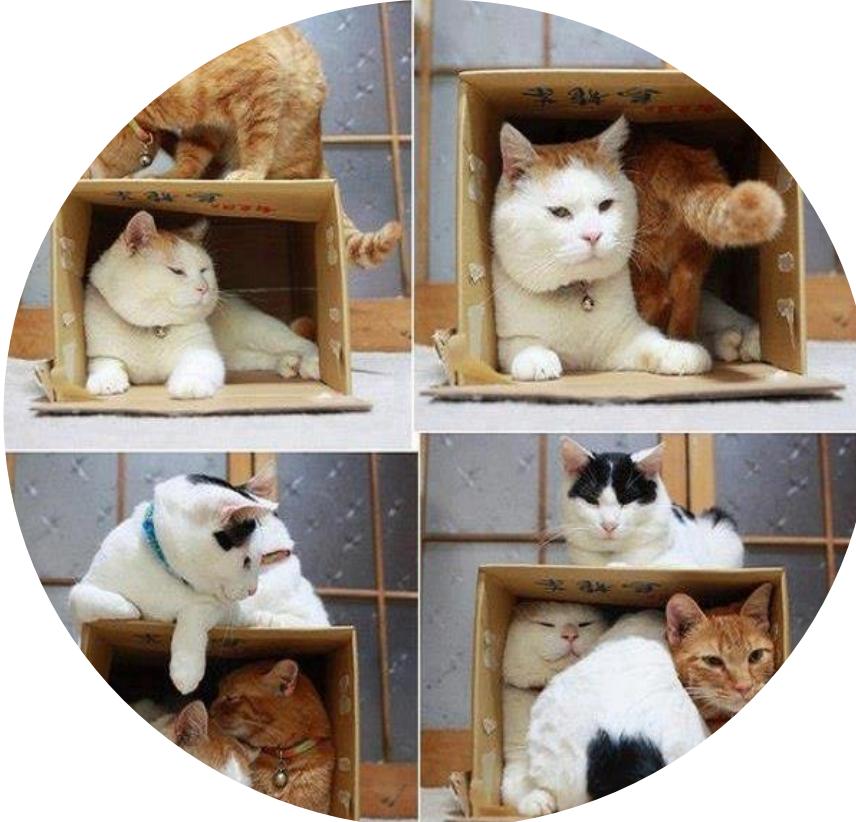
Lz	Ss	Ja
lazybones	start.spring.io	java

Sb	Gy
spring boot	groovy

Th	Gr	Ad
thrift	gradle	asciidoc

So	Cp	Ls	Dd	Co	Sd	Sl
SOA	Centralization paradox	LSD principle	DDD	Conway's Law	Smart Docs	Smart Libs

Изоляция данных



Всё просто



Одно приложение - одна БД



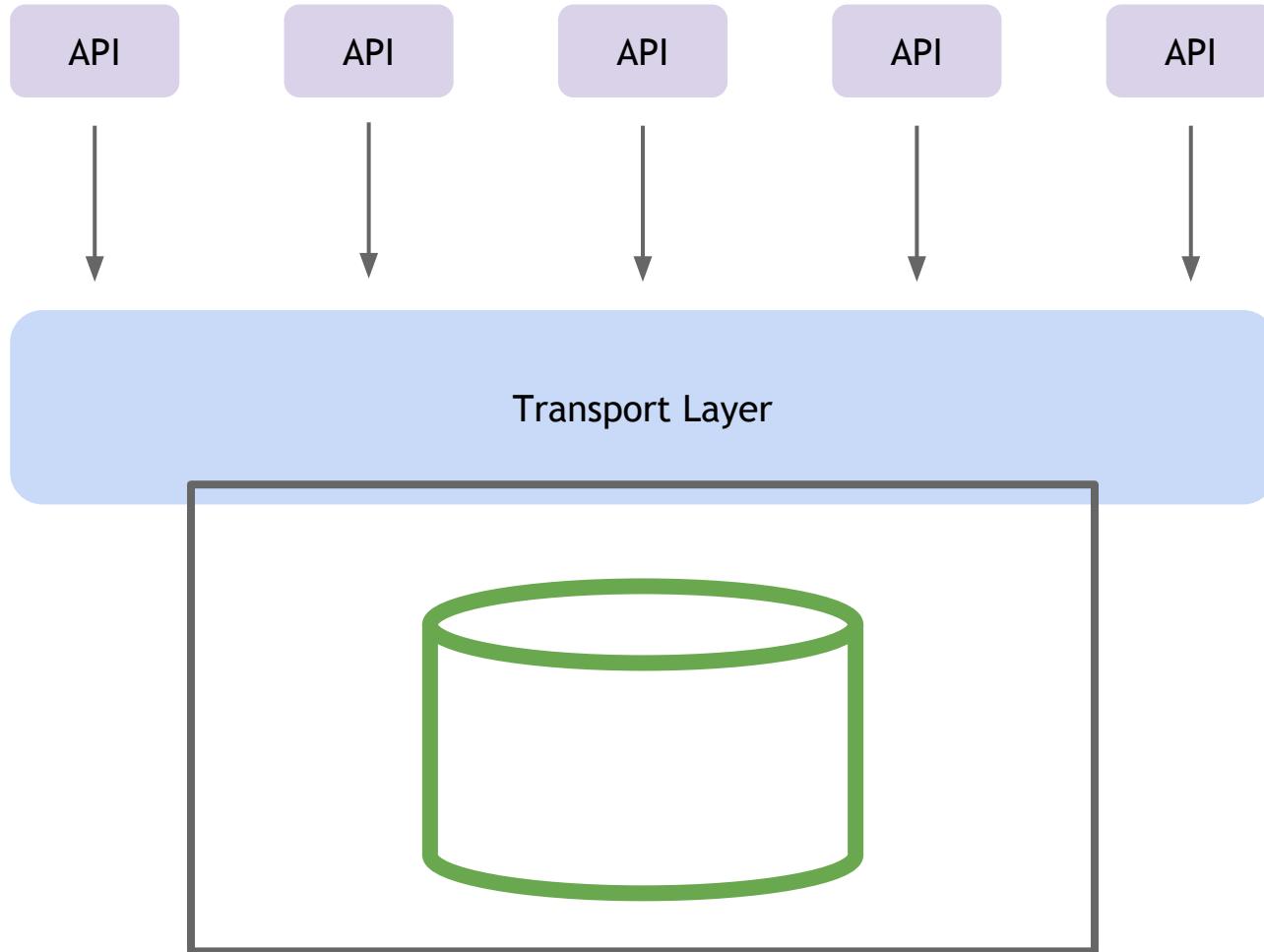
N сервисов → 1 БД

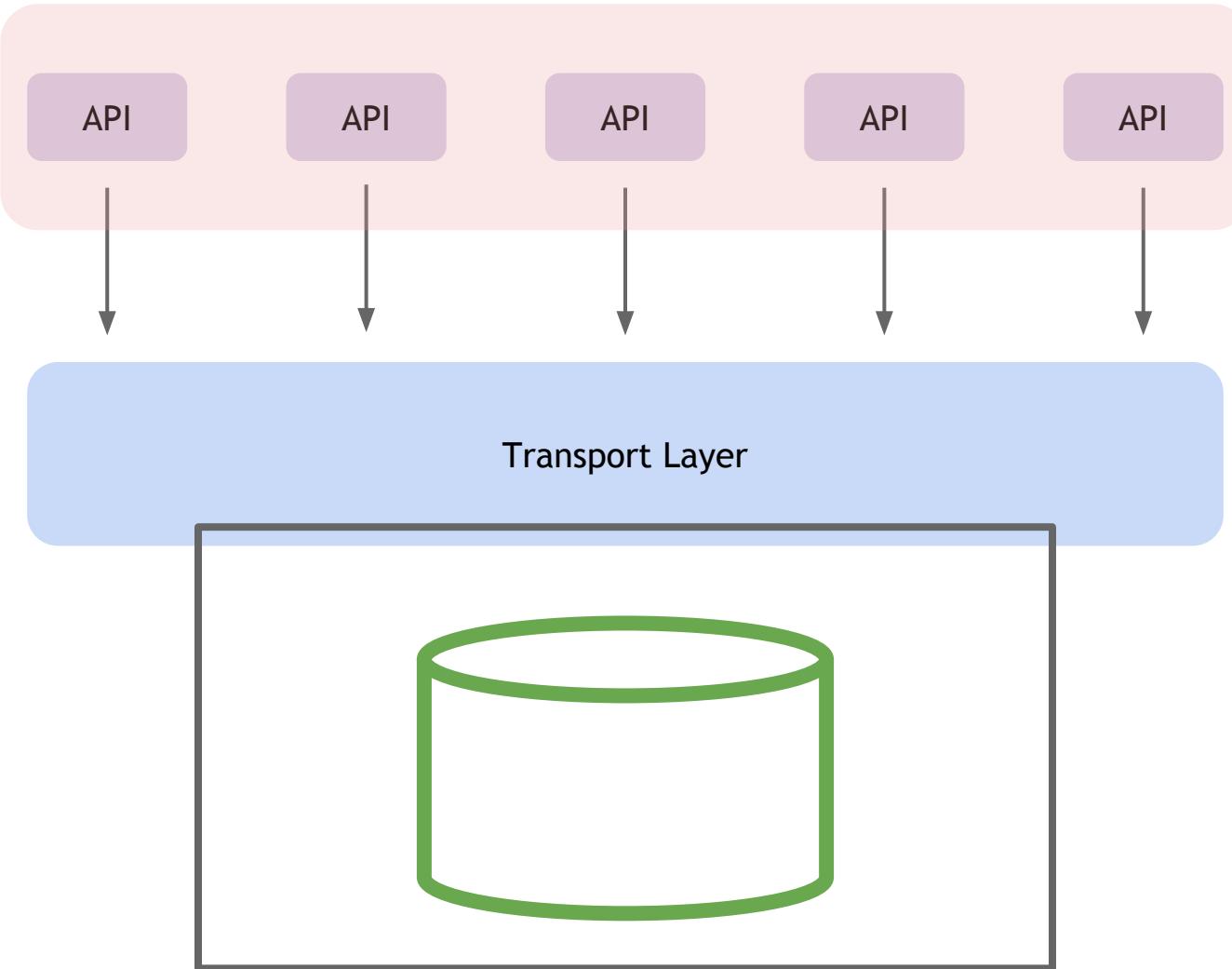
Изоляция на уровне таблиц или схем



N сервисов → 1 БД

Изоляция через хранимые
процедуры







N сервисов → 1 БД

Распределённый монолит

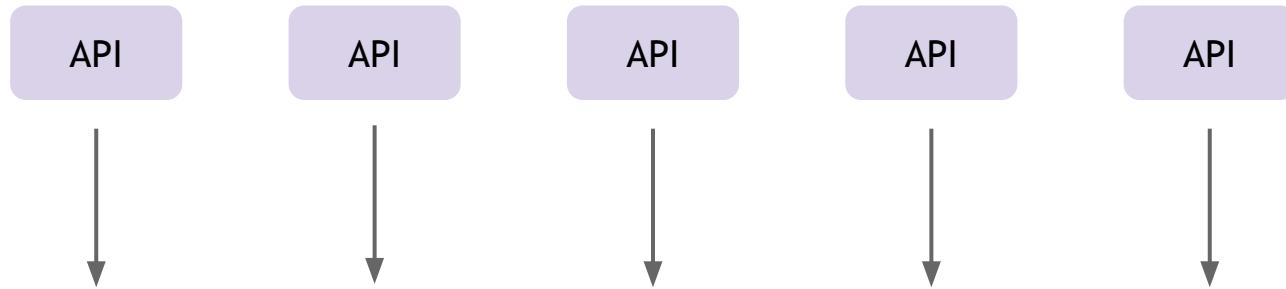


N сервисов → N Бд

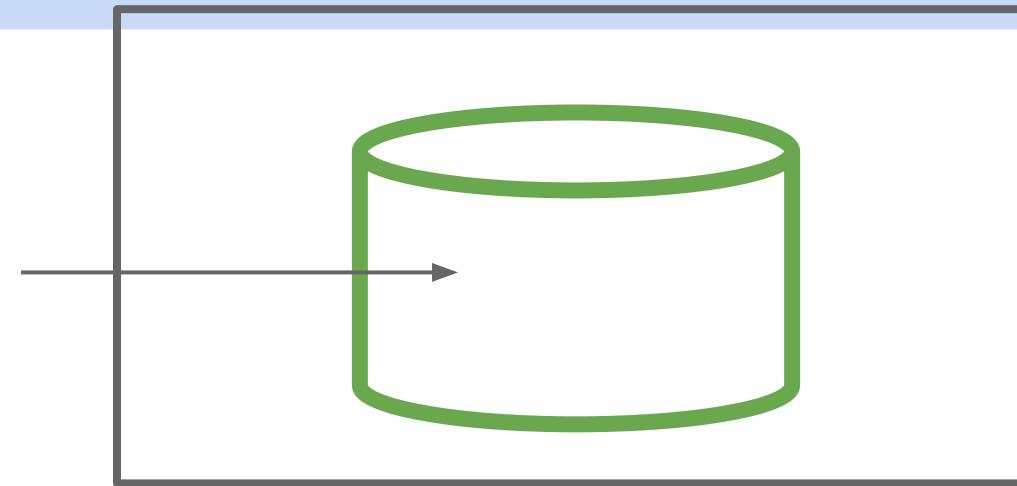


N сервисов → N БД

Распределённые транзакции –
это больно



Очень
много
кода





N сервисов → N БД

Ваше legacy тянет вас на дно

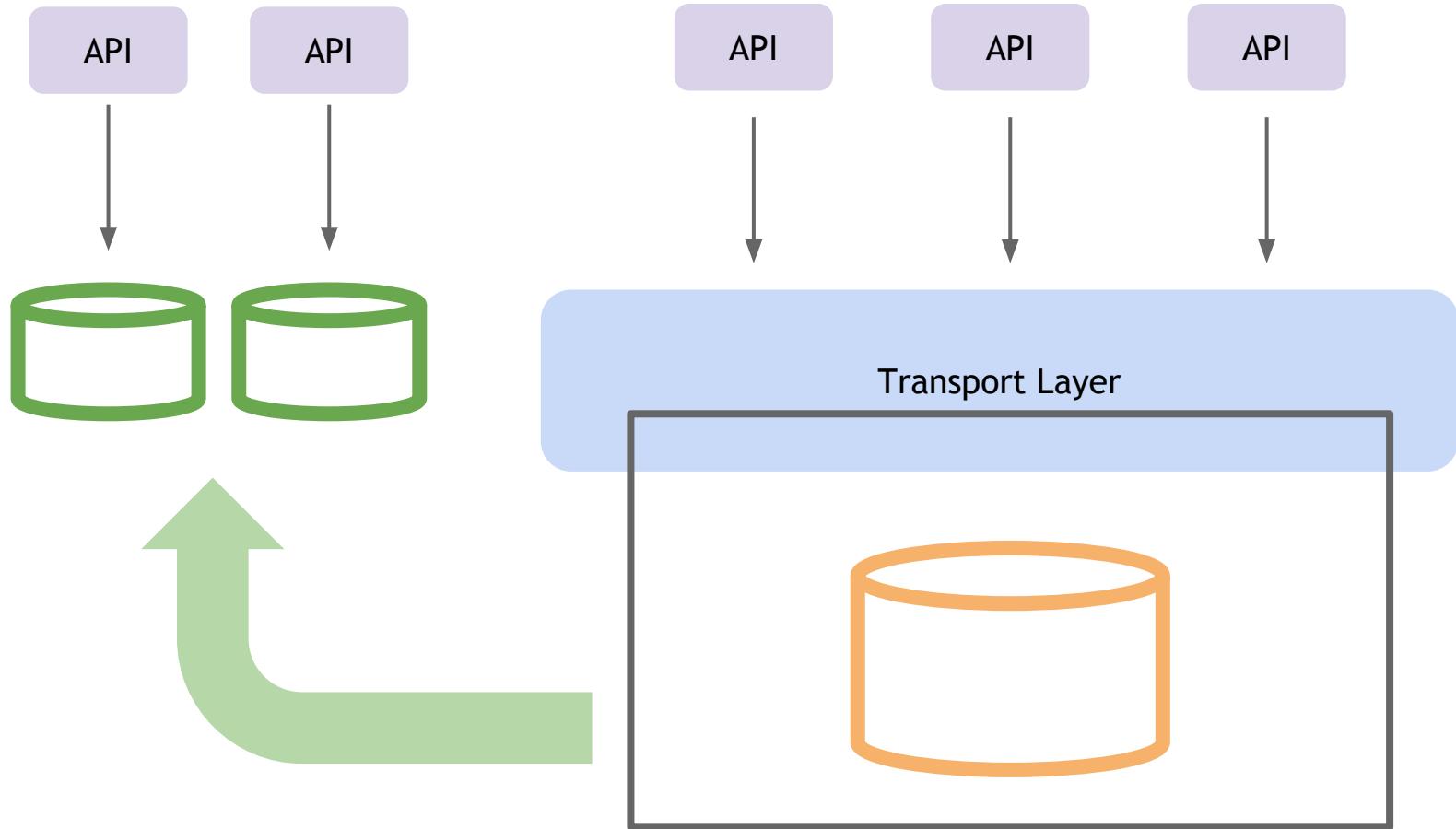
ИСТИНА ГДЕ-ТО ПОСЕРЕДИНЕ





М сервисов → 1 БД
L сервисов → L БД

М + L = N, M << L





М сервисов → 1 БД
L сервисов → L БД

М + L = N, М << L

Yo		
yoman		
Lz	Ss	Ja
lazybones	start.spring.io	java
Sb		Gy
spring boot		groovy
Th		Gr Ad
thrift		gradle asciidoc

So	Di	Cp	Ls	Dd	Co	Sd	Sl
SOA	Data Isolation	Centralization paradox	LSD principle	DDD	Conway's law	Smart Docs	Smart Libs

Доставка





war/ear

Зависимость от сервера приложений



executable fatJar

Зависимость от системных
библиотек



executable fatJar/npm-пакет

Разный менеджмент для разных
форматов дистрибуции



docker/rkt/packer

war < fatJar < docker



scp fat.jar root@prod101:/home/app/



scp fat.jar root@prod101:/home/app/

Что делать с консистентностью?
Как быть с доступностью?



```
./gradlew deployToArtifactory  
ansible -i [stage,dev,test] -t deploy
```

“Прибитая молотком” конфигурация

```
- hosts: dc1-rest-api
  roles:
    - { role: api, name: transactions-api, service_port: 9081 }
    - { role: api, name: cards-api, port: 8081 }
    - { role: api, name: customers-api, port: 9091 }
```

dc1-rest-api

<ip_address_1>

<ip_address_2>

...

Как быстро масштабироваться?



Нужна оркестрация

```
{  
  "id": "/retail-online-bank/middle/transactions-api",  
  "cpus": 1,  
  "mem": 1024,  
  "instances": 5,  
  "container": {  
    "docker": {  
      "image": "docker/retail-online-bank-transactions-api:1.17.0",  
      "portMappings": [  
        {  
          "containerPort": 8080,  
          "servicePort": 0  
        }  
      ]  
    }  
  }  
}
```



mesos / kubernetes / ∞

Как использовать ресурсы



Bare Metal



**Bare
Metal**

**Virtual
Machines**



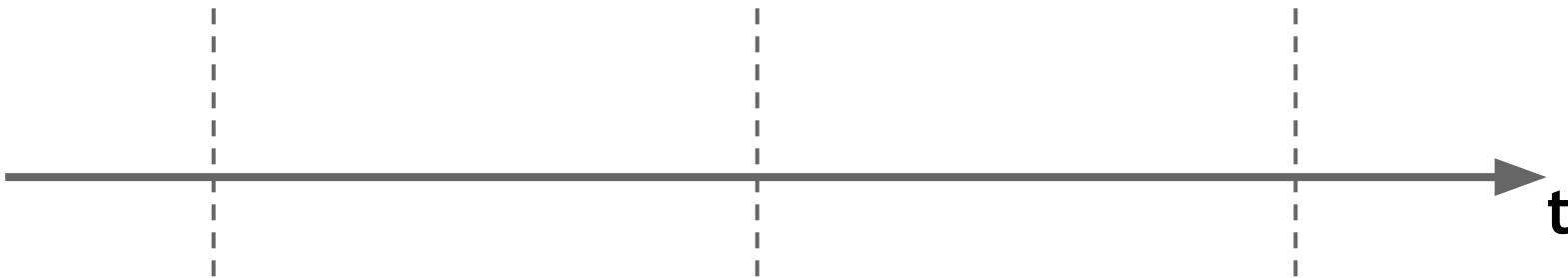
Bare Metal



Virtual Machines



Lightweight Containers



Bare Metal



Virtual Machines



Lightweight Containers



a2a-transactions-api

Deploying (7 of 7 instances)

3 Healthy (43%) 4 Unhealthy (57%) 0 Unknown

Scale Application

Restart

⚙️

Instances

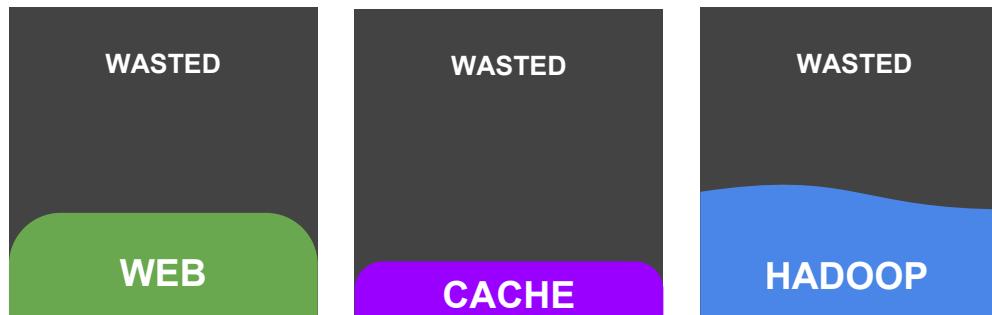
Configuration

Debug

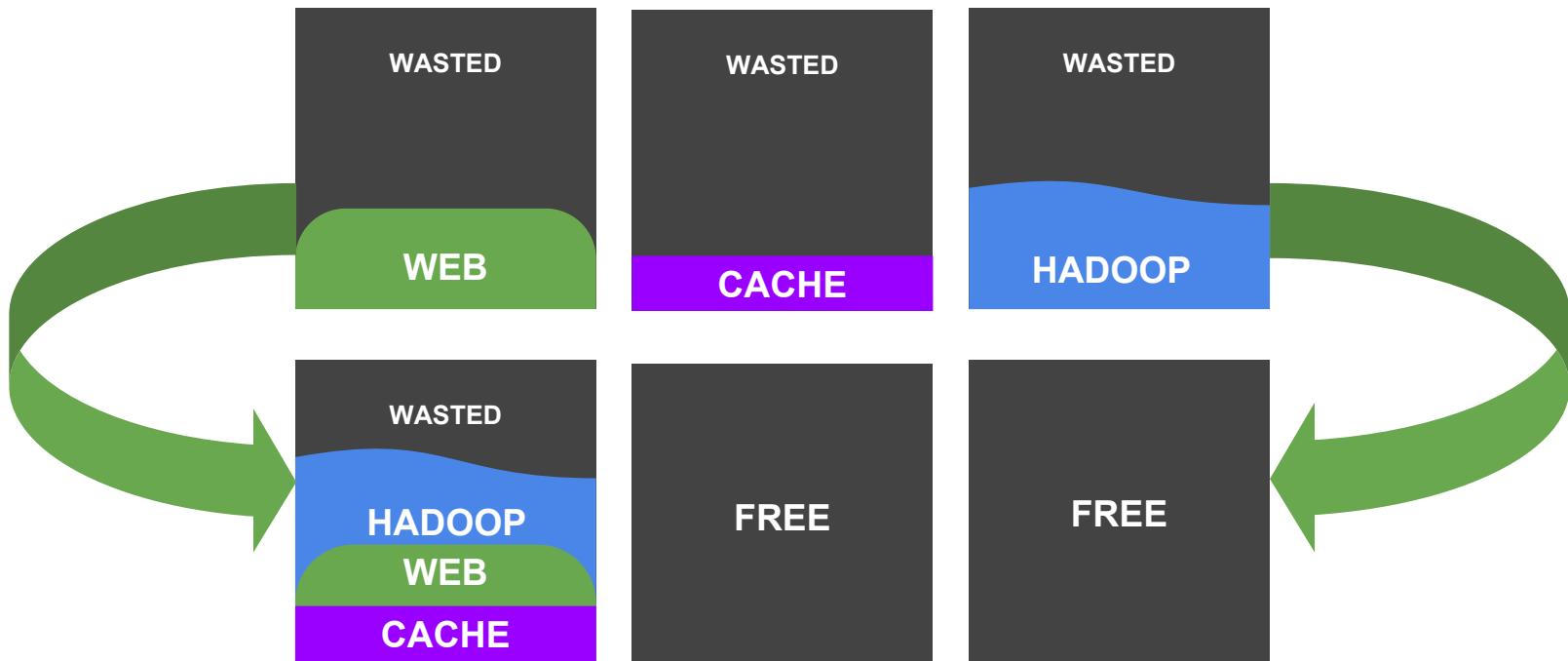
⟳ Refresh

ID	Health	Status	stderr	stdout	Version	Update
retail-online-bank_middle_a2a-transactions-api.7f7bcb9-892a-11e6-900f-e248e8be00cd abmdev:20757	Healthy	Started	stderr	stdout	17 days ago	03.10.2016, 8:30:3
retail-online-bank_middle_a2a-transactions-api.6a0d09b5-8cb4-11e6-900f-e248e8be00cd abmdev:20204	Healthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:2
retail-online-bank_middle_a2a-transactions-api.6eafb6ca-8cb4-11e6-900f-e248e8be00cd abmdev:20099	Unhealthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:2
retail-online-bank_middle_a2a-transactions-api.6eafb6c9-8cb4-11e6-900f-e248e8be00cd abmdev:20697	Unhealthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:3
retail-online-bank_middle_a2a-transactions-api.6eafb6c8-8cb4-11e6-900f-e248e8be00cd abmdev:20688	Healthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:3
retail-online-bank_middle_a2a-transactions-api.6eaf8fb7-8cb4-11e6-900f-e248e8be00cd abmdev:20175	Unhealthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:3
retail-online-bank_middle_a2a-transactions-api.6eaf8fb6-8cb4-11e6-900f-e248e8be00cd abmdev:20494	Unhealthy	Started	stderr	stdout	a few seconds ago	07.10.2016, 20:35:3

Использование группы машин как одной



Представьте, что кластер это просто одна большая машина



Yo								
yoman								
Lz	Ss	Ja						
lazybones	start.spring.io	java						
Sb		Gy	Py				Ms	Kb
spring boot		groovy	python				mesos	Kubernetes
Th	Ar	An	Gr	Ad	Dr	Au	Ma	Ch
thrift	Artifactory	ansible	gradle	asciidoc	docker	aurora	marathon	chronos
So	Di	Cp	Ls	Dd	Sa	Sd	Sl	Ds
SOA	Data Isolation	Centralization paradox	LSD principle	DDD	SAGAS	Smart Docs	Smart Libs	Dynamic Sharing

Найти нужный экземпляр





```
insert into settings_table values (key, endpoint)
```

“Прибитая молотком” конфигурация



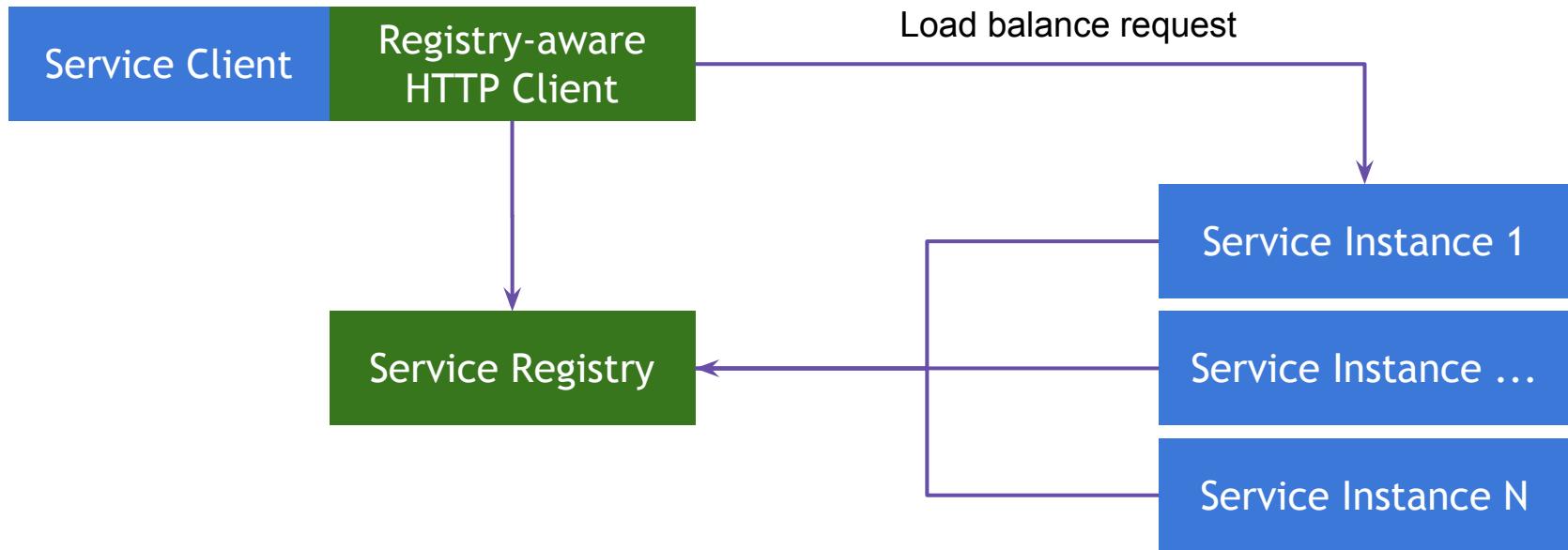
update template → services.location
kill -s HUP nginx

Остаётся статическая привязка
ресурсов

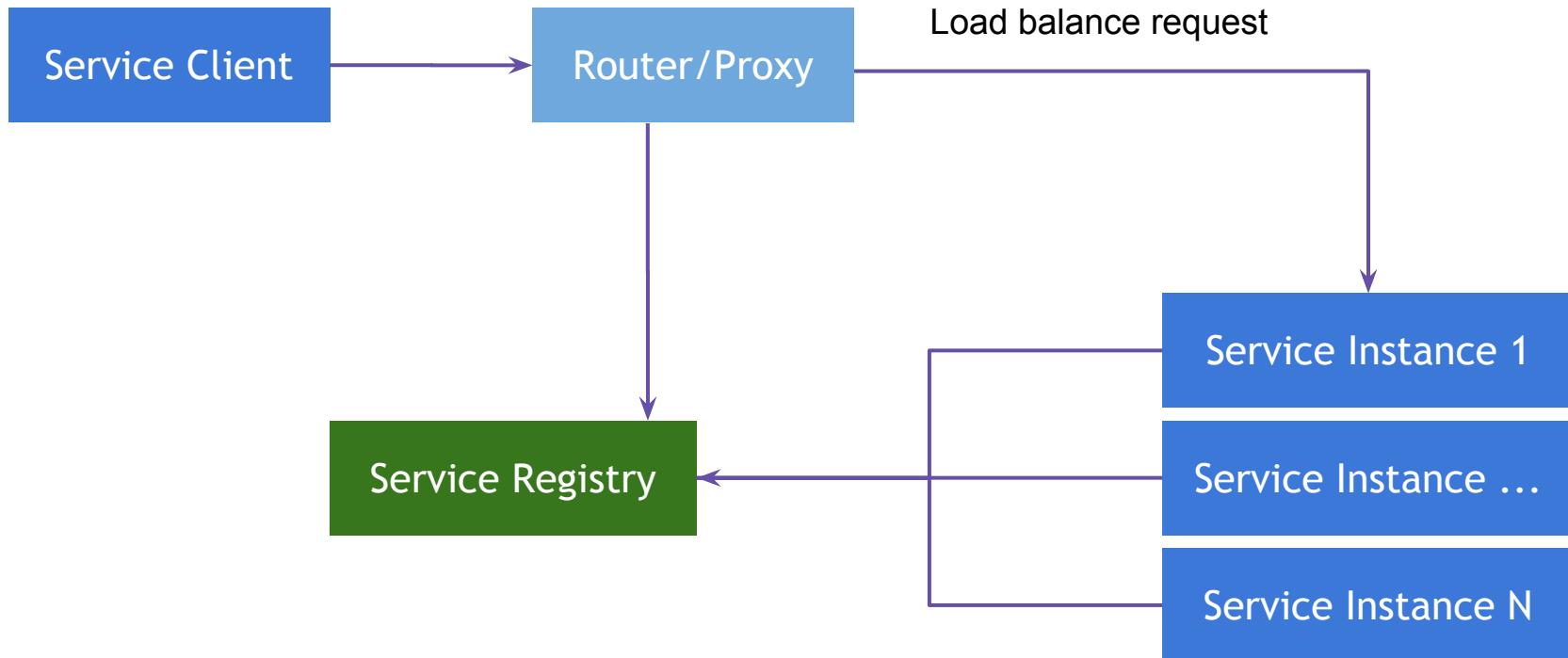


eureka/consul/etc/zookeeper

Client side discovery



Server side discovery

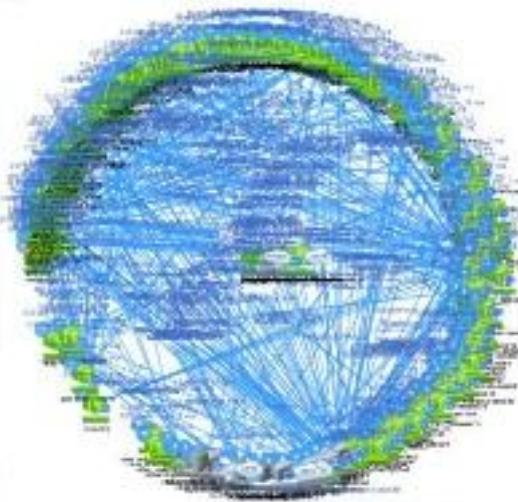


Yo yoman								
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	
Sb spring boot		Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos
So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Sa SAGAS	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing

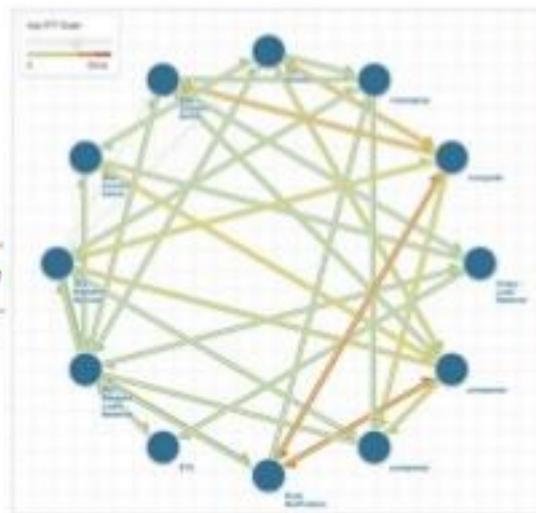
Защити себя сам



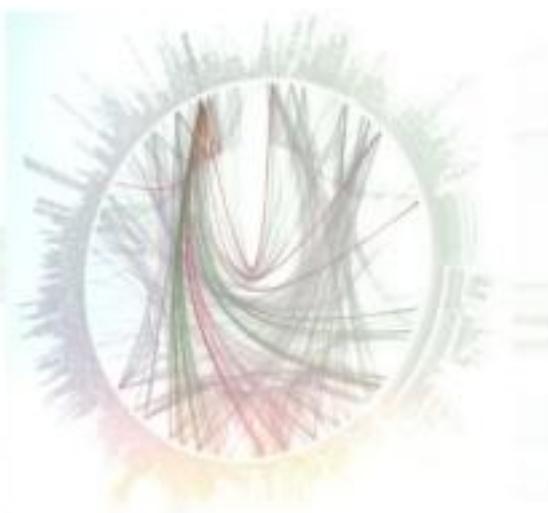
“Death Star” Architecture Diagrams



Netflix



Gilt Groupe (12 of 450)

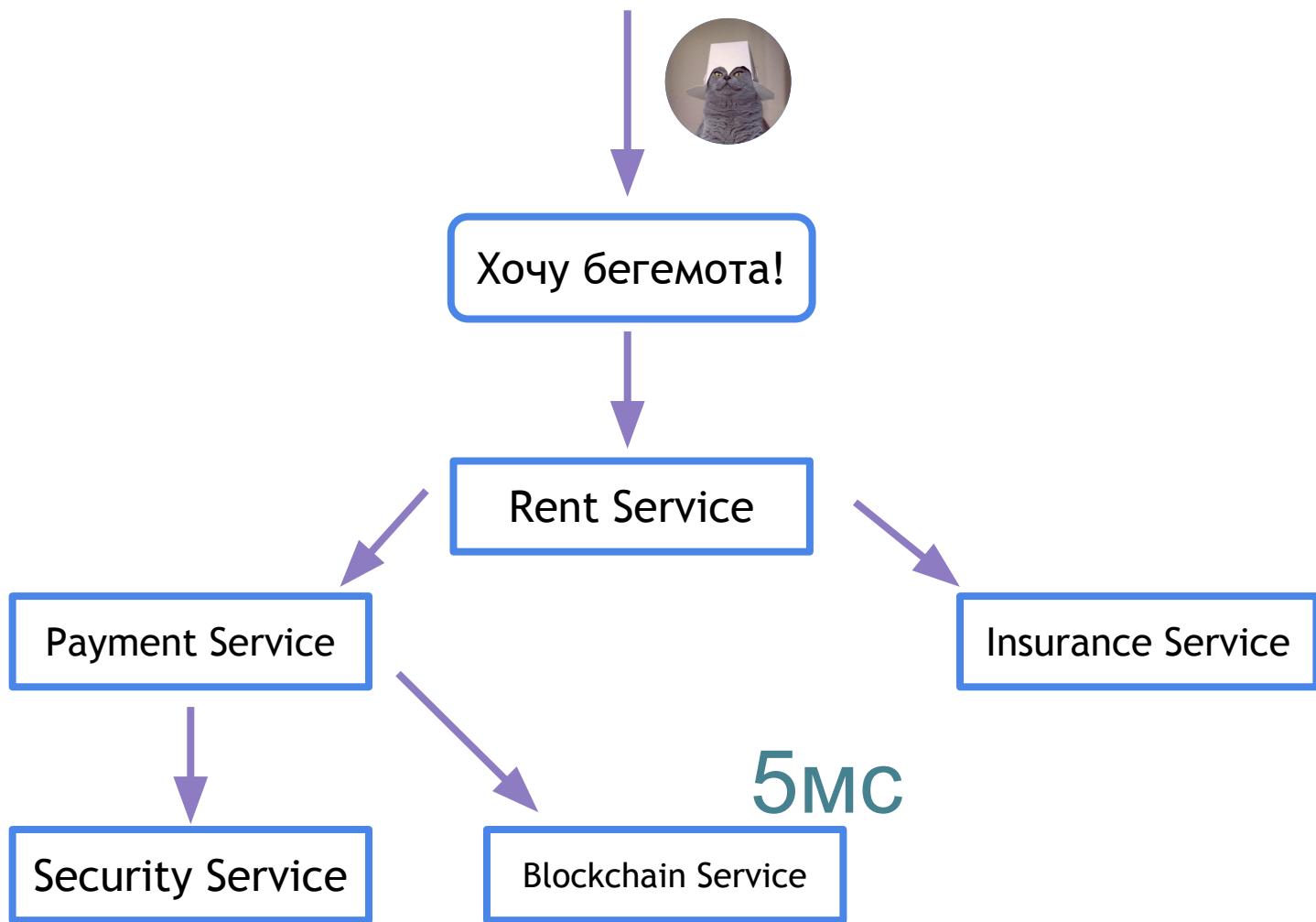


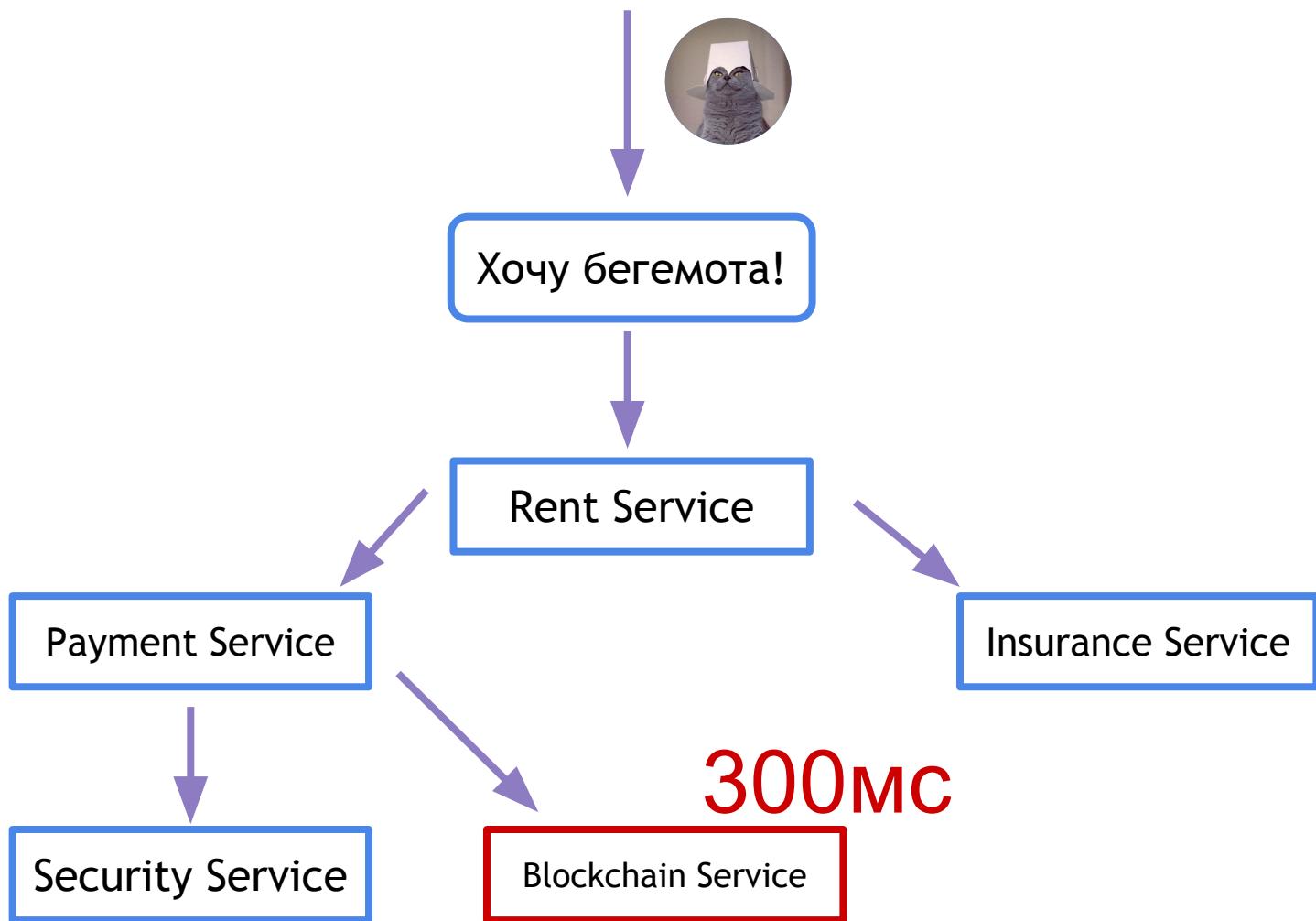
Twitter

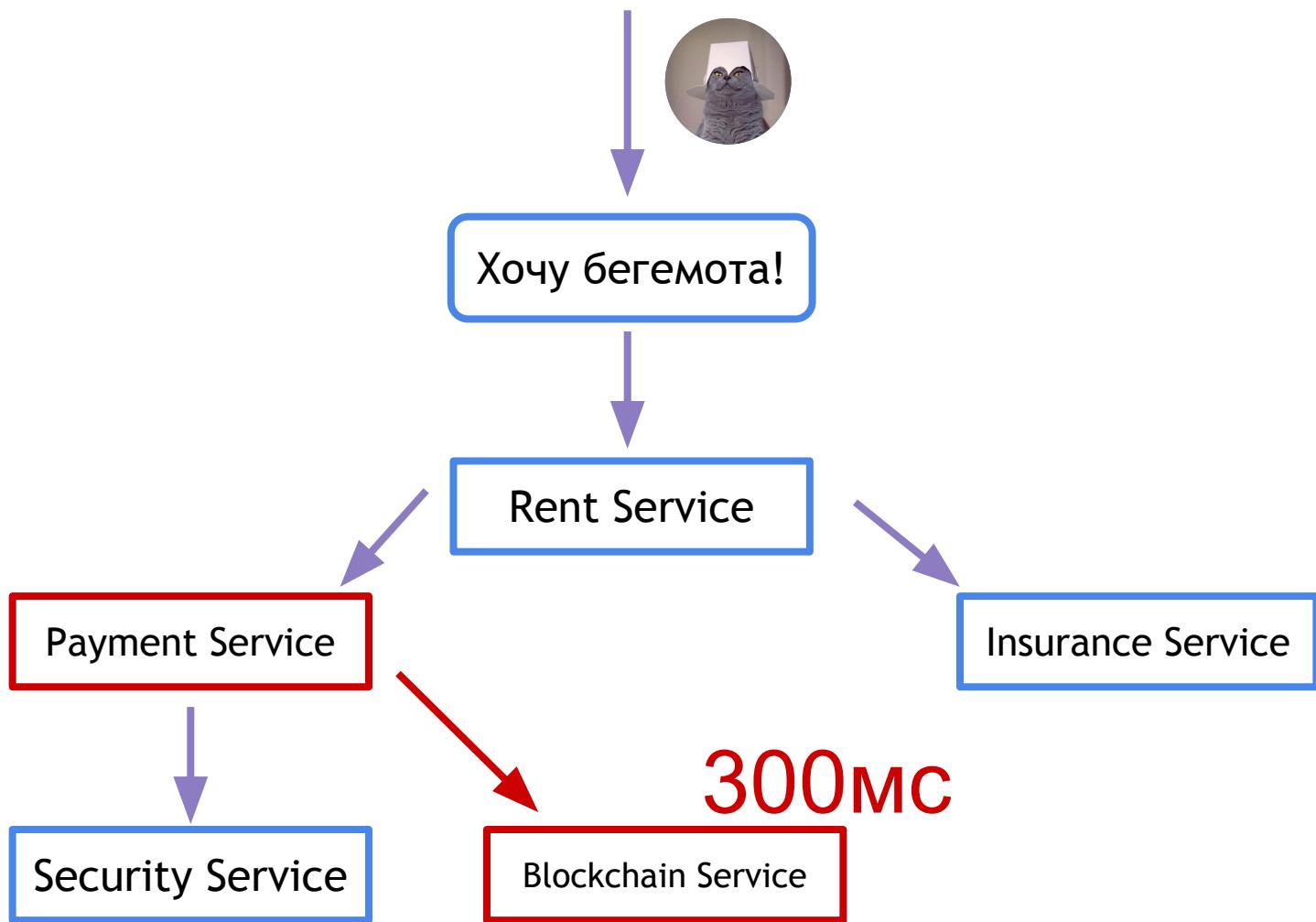


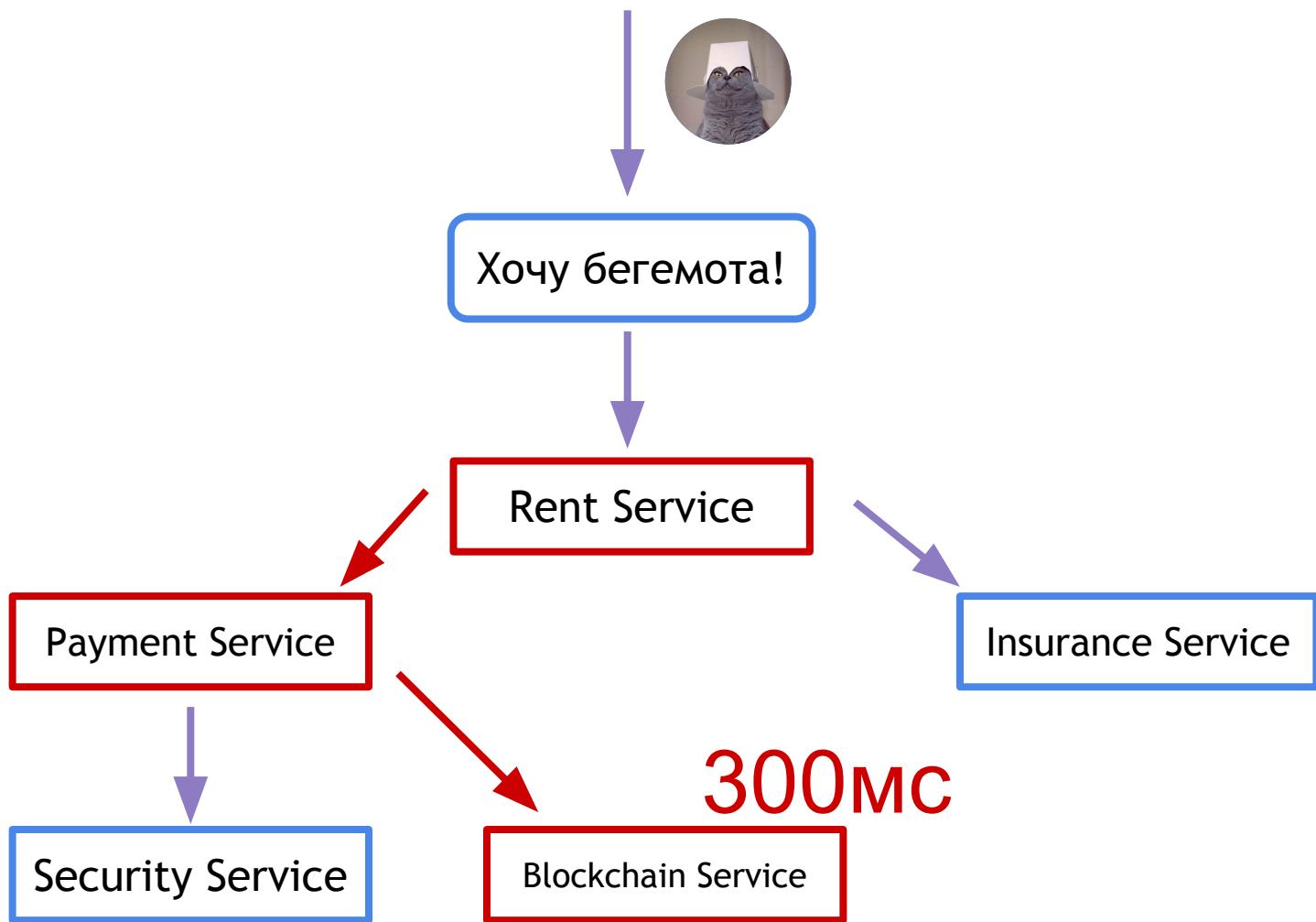
As visualized by Appdynamics, Boundary.com and Twitter internal tools

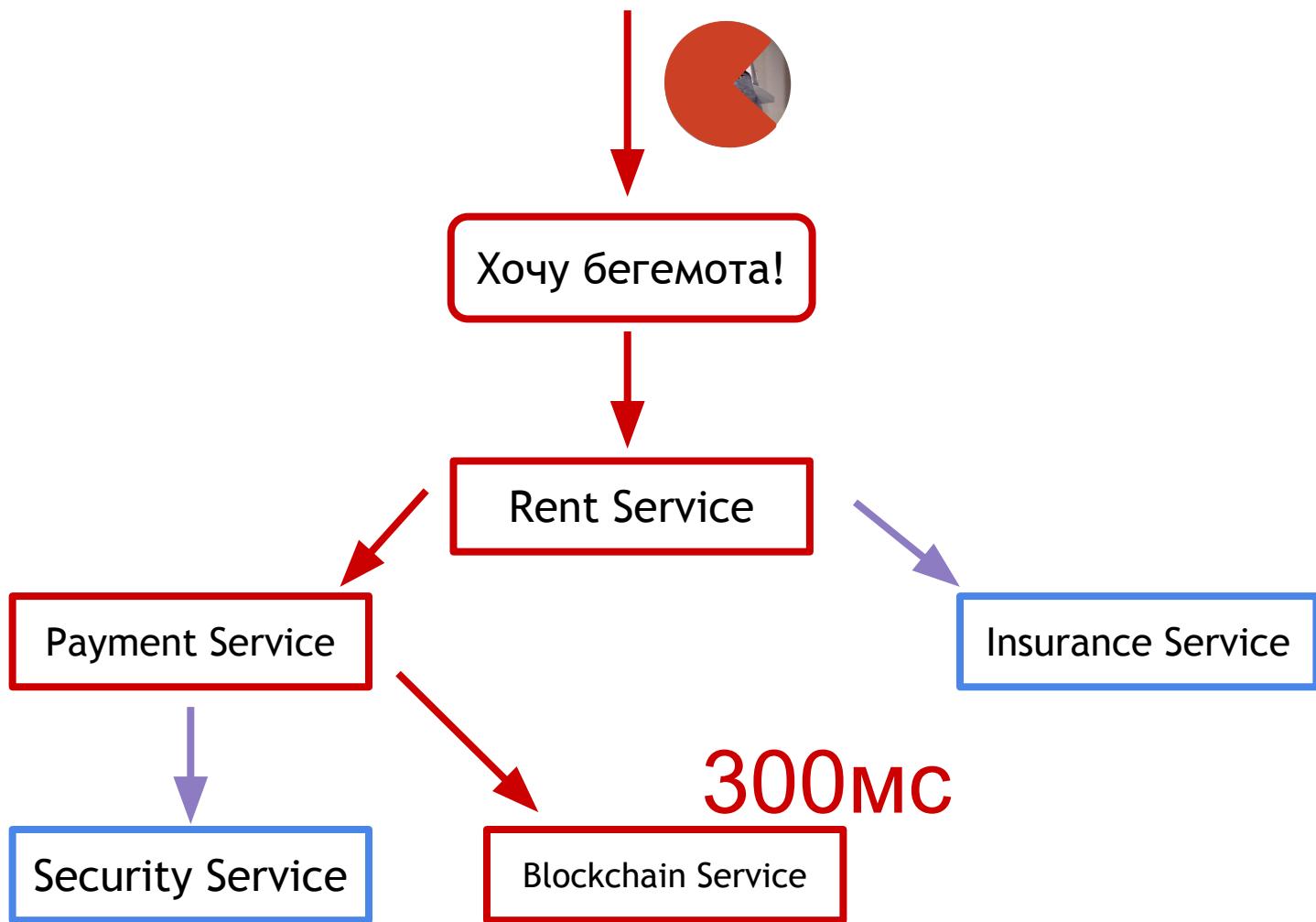
BV | Battery Ventures



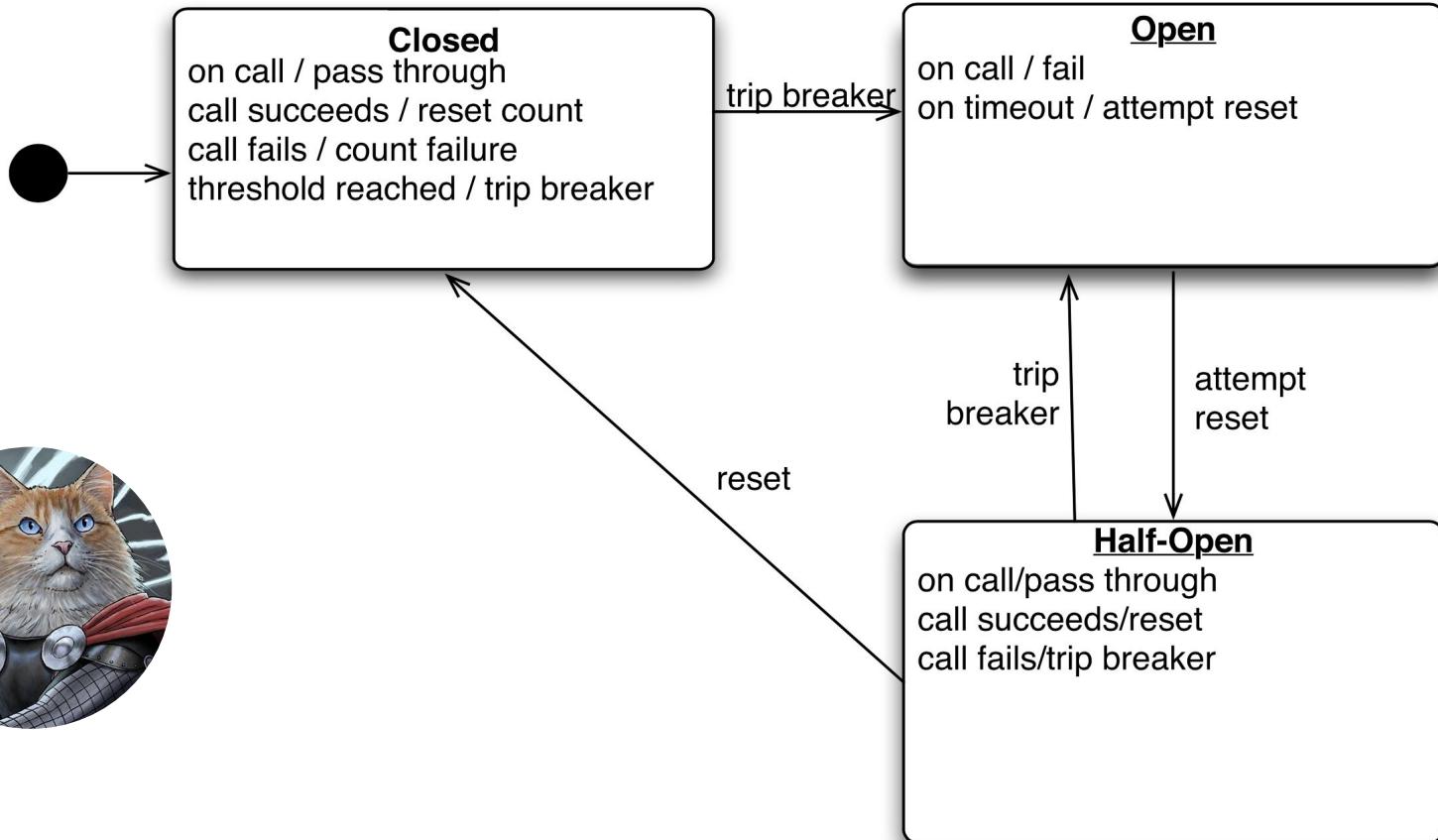


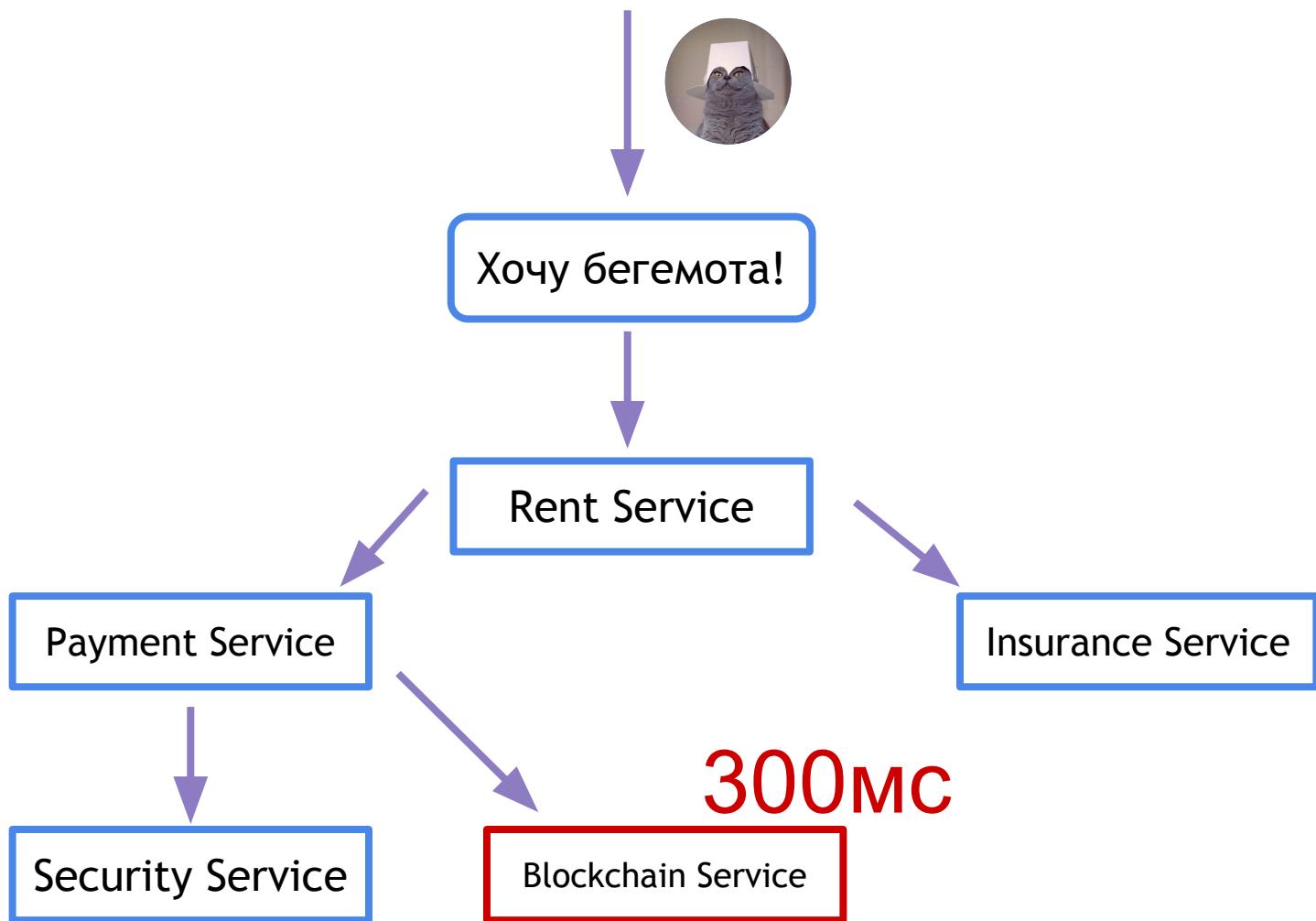


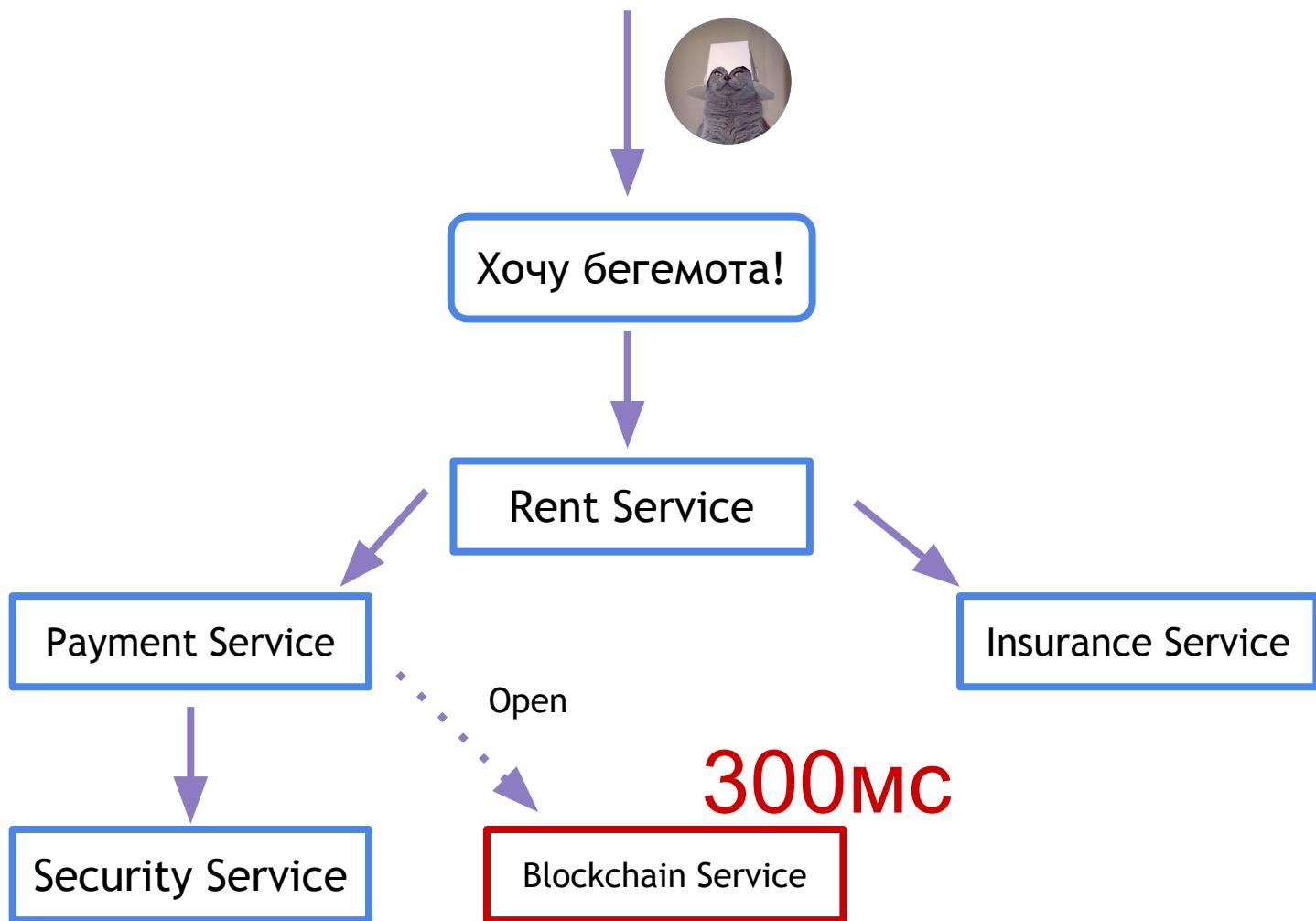


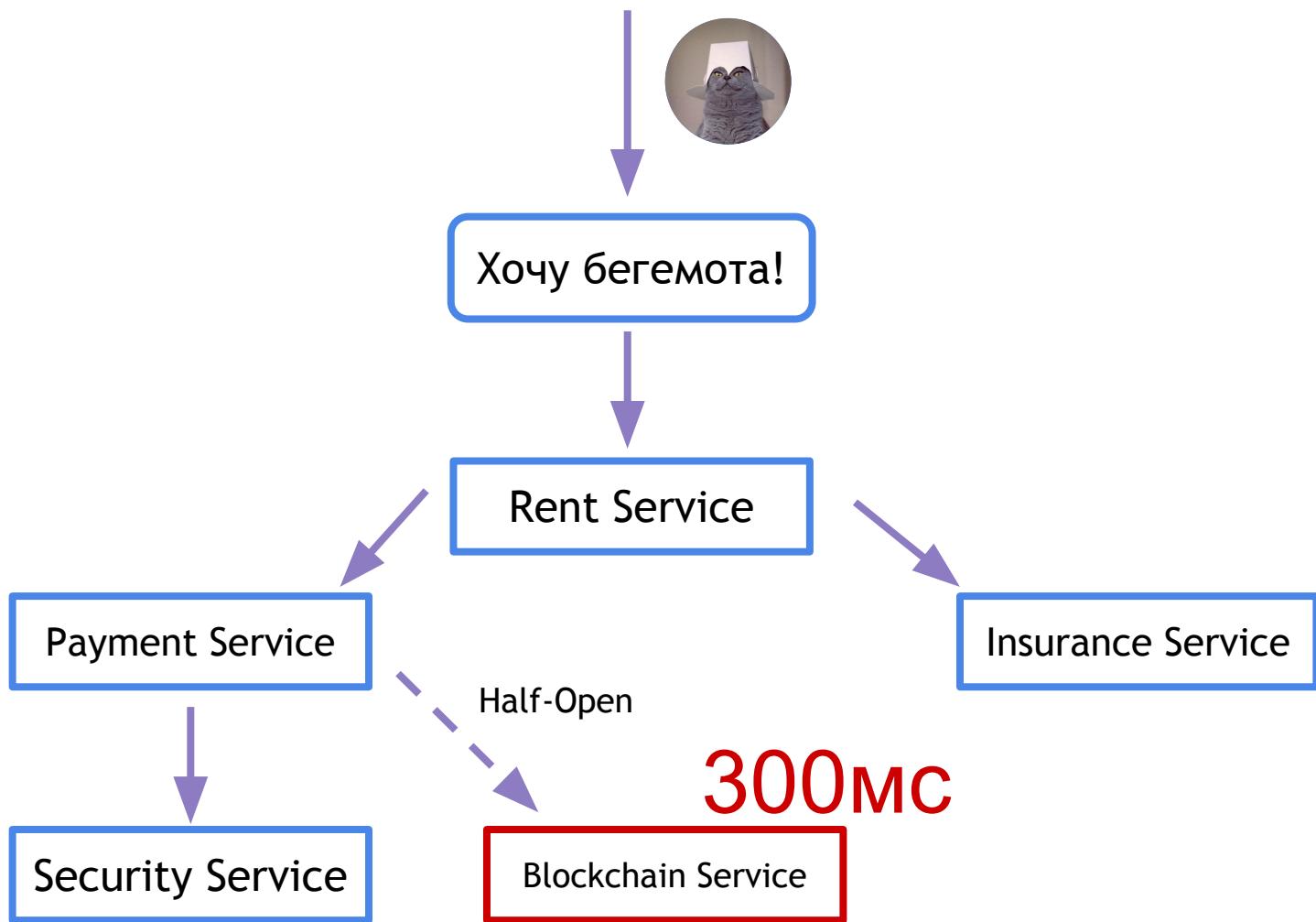


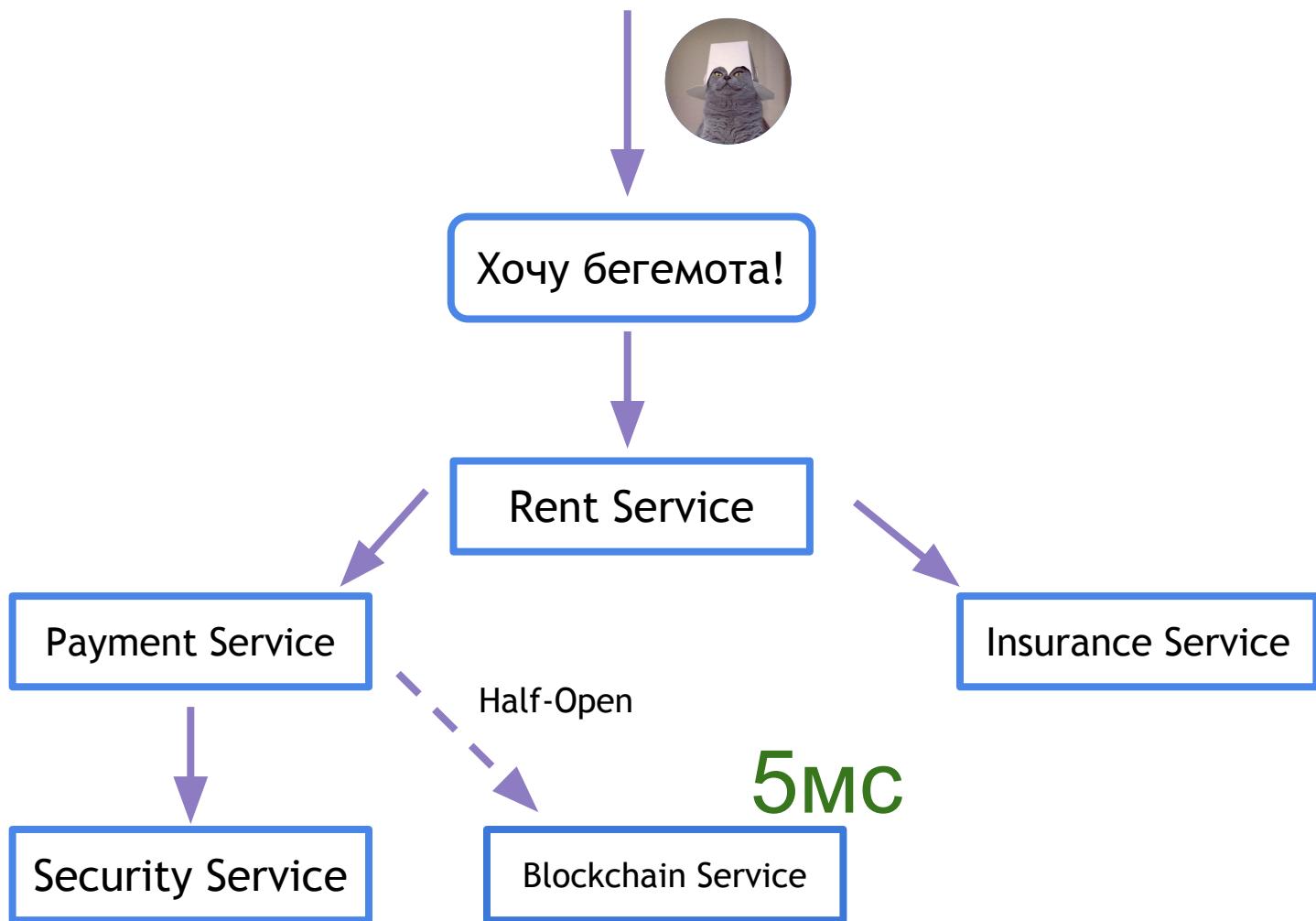
Circuit Breaker



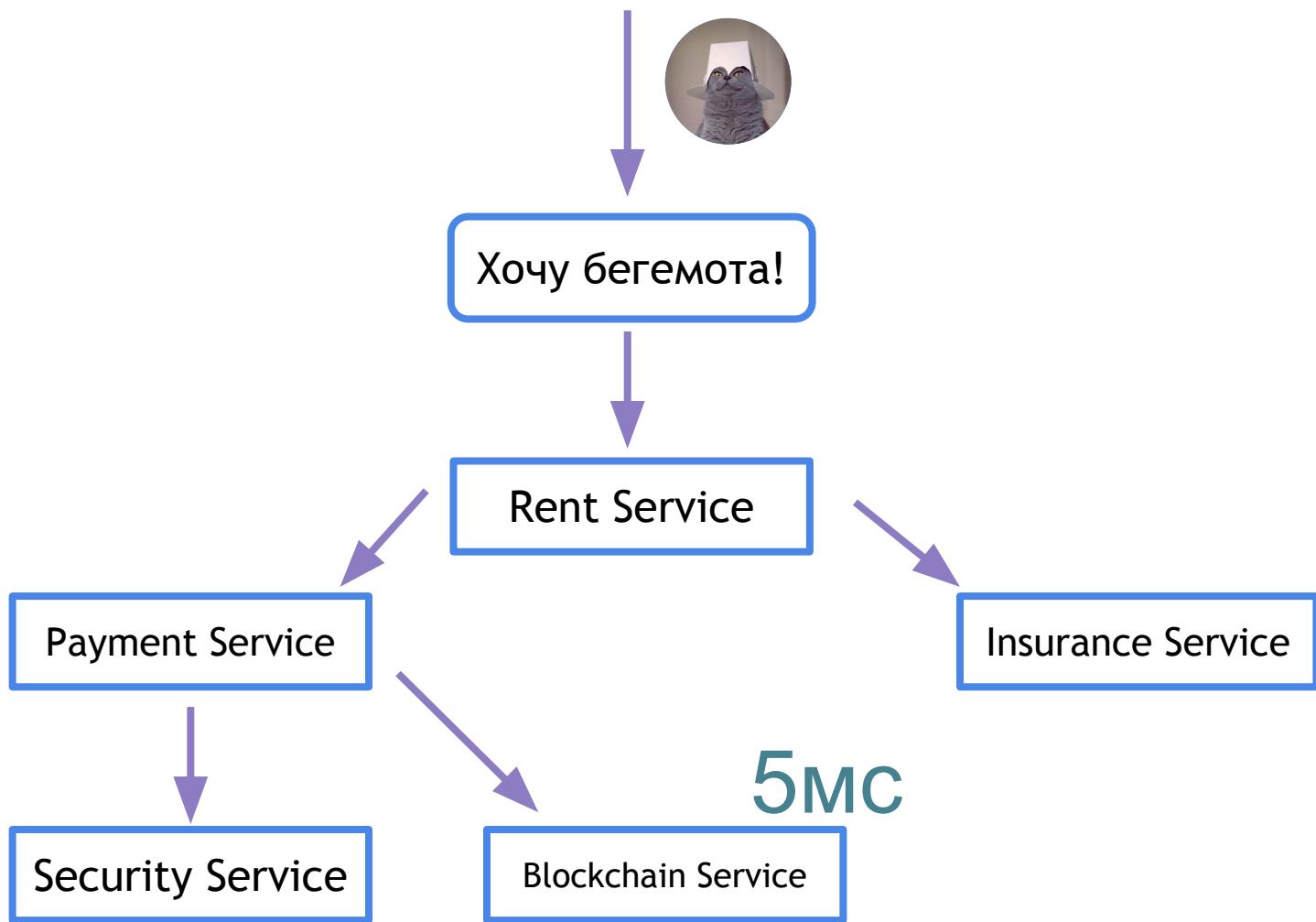








5МС



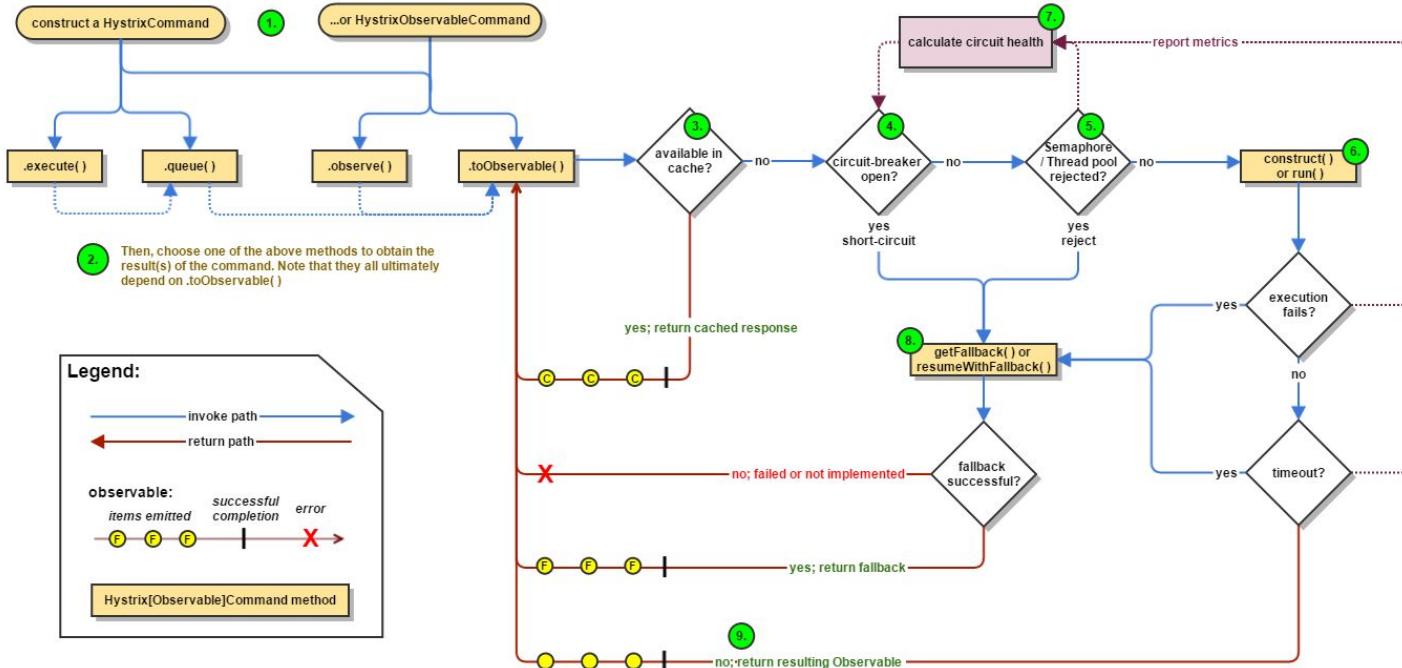


hystrix/apache camel/akka



HYSTRIX

DEFEND YOUR APP



Yo yoman								
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing
------------------	-----------------------------	-------------------------------------	----------------------------	------------------	---------------------------	-------------------------	-------------------------	------------------------------

Следи за своим ПО





Нет трассировки - нет проблем? :)



X-Request-Id = X-Request-Id ?: new ID

Простой вариант с ServletFilter



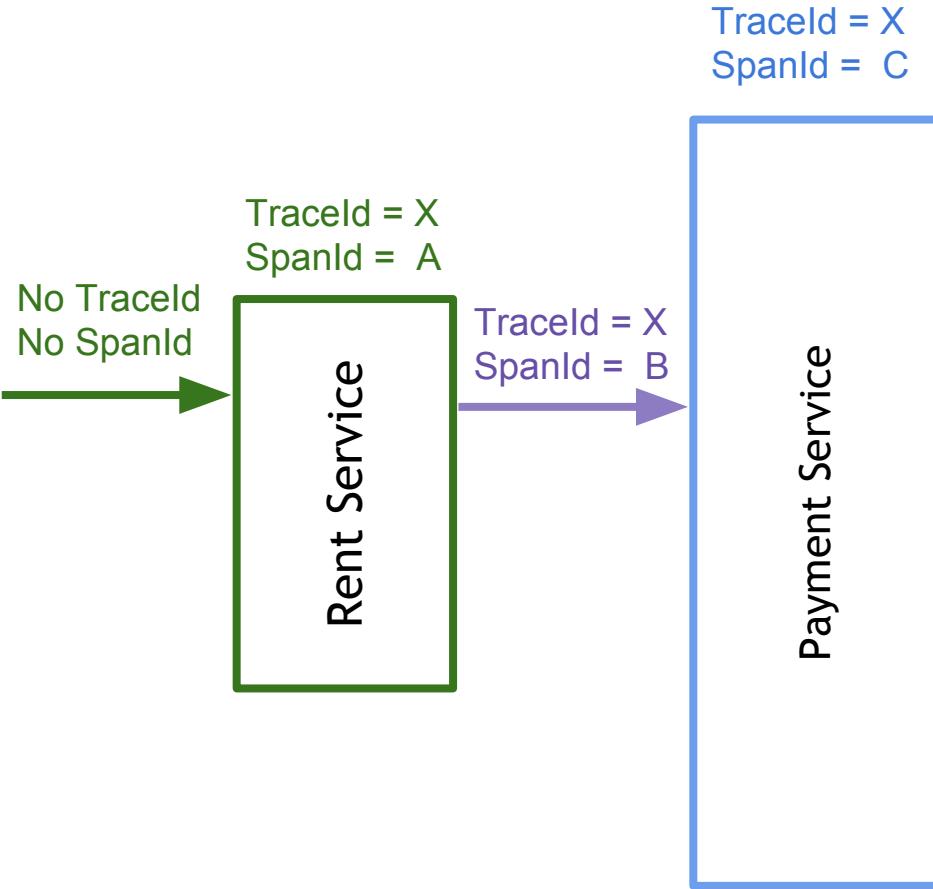
spring-cloud-sleuth/open zipkin

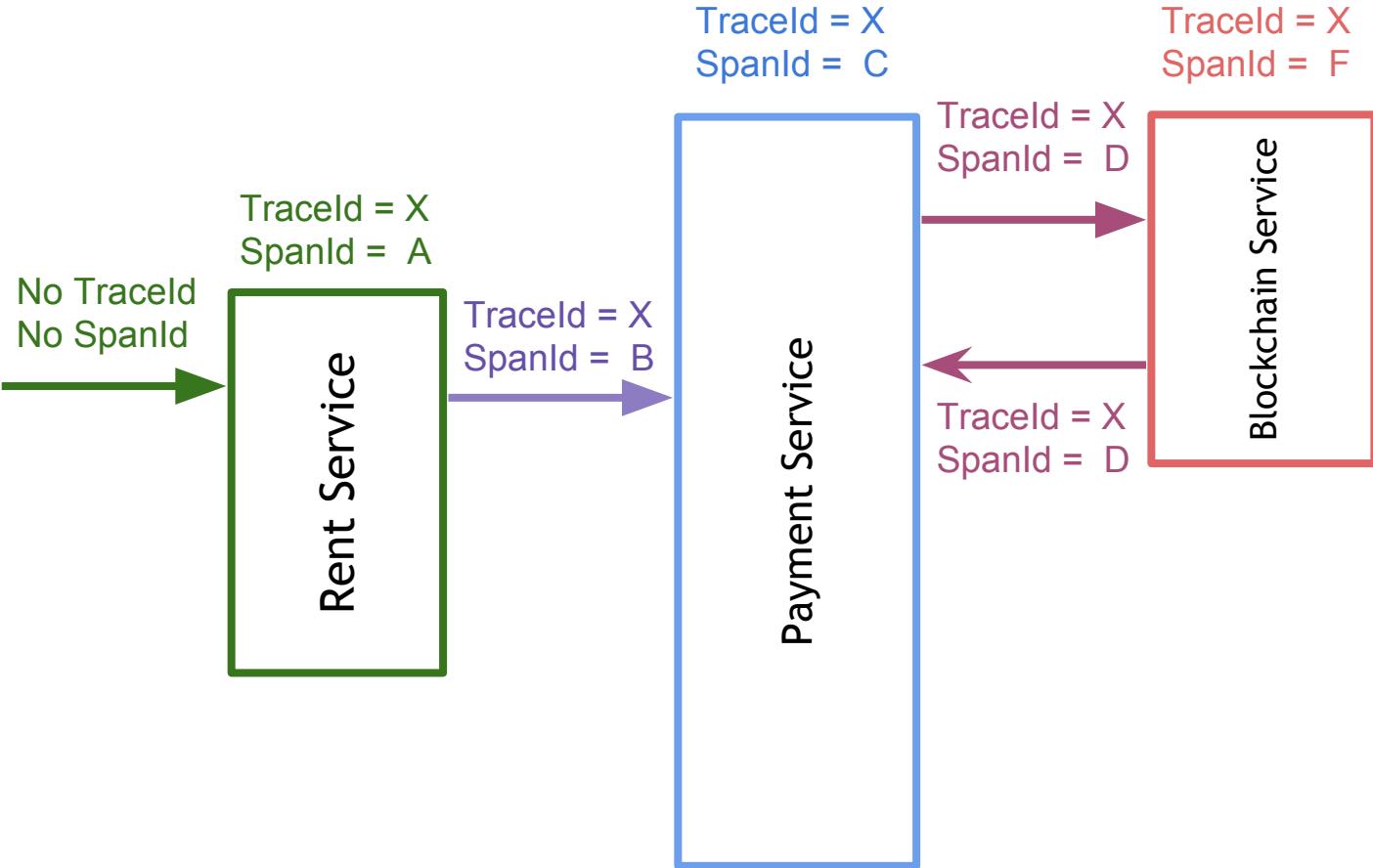


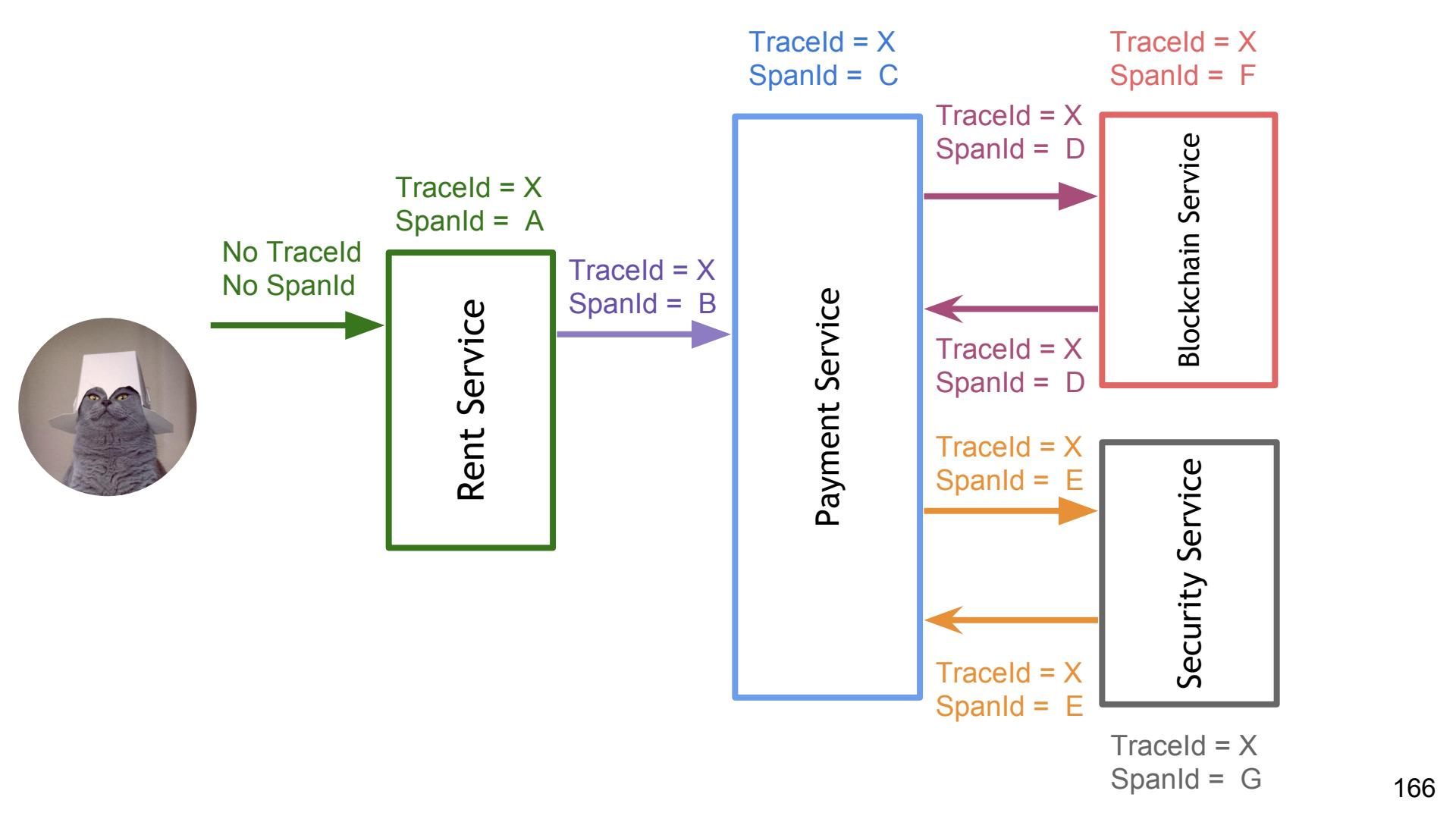
No TraceId
No SpanId

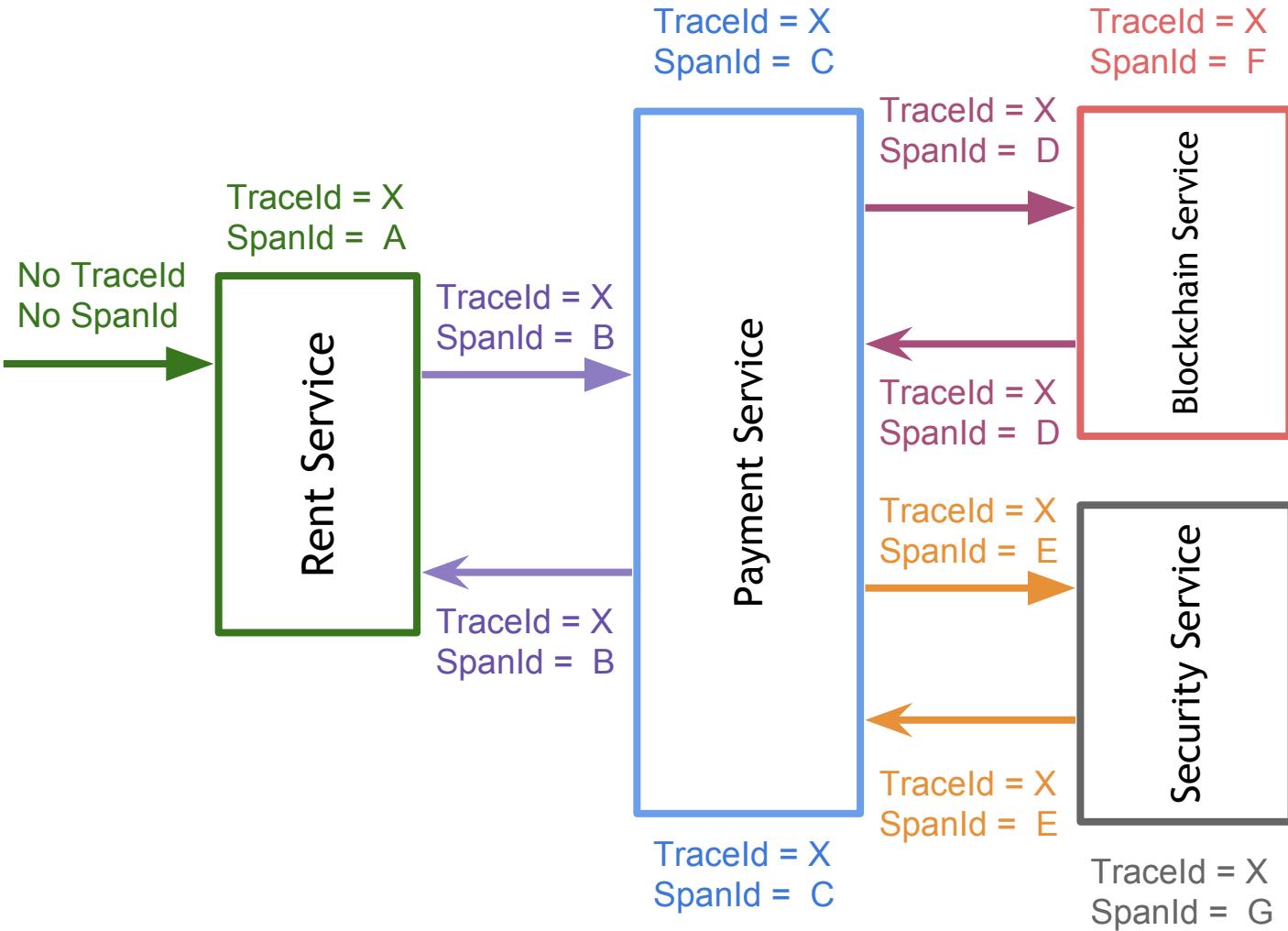
TraceId = X
SpanId = A

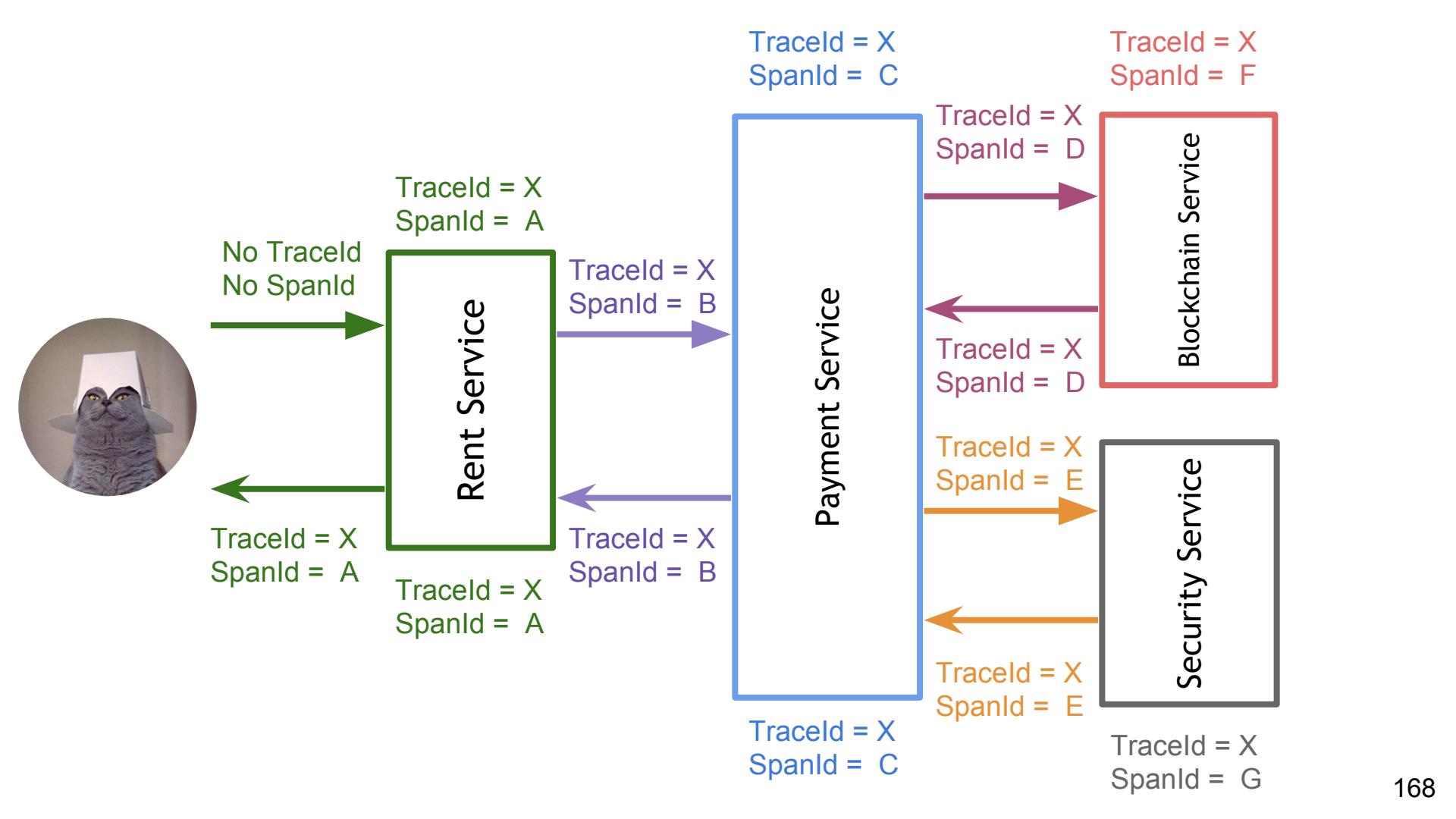












Zipkin - Index

localhost:9412/?serviceName=message-client&spanName=all&endTs=1455553899906&minDuration=&limit=10&annotationQuery=

Apps to read cf-mvp.sh

Zipkin Investigate system behavior Find a trace Dependencies Go to trace

message-client all End time 02-15-2016 11:31 Duration (μs) >=

Limit 10 Find Traces ?

Annotations Query (e.g. "finagle.timeout", "http.path=/foo/bar/ and cluster=foo and cache.miss")

Showing: 9 of 9 Sort: Longest First

Services: message-client

1.163s 5 spans
message-client 100%
message-client x2 1163ms | message-service x3 1163ms 2 minutes ago

869.000ms 5 spans
message-client 100%
message-client x2 869ms | message-service x3 869ms 2 minutes ago

699.000ms 5 spans
message-client 100%
message-client x2 699ms | message-service x3 699ms 2 minutes ago

629.000ms 5 spans
message-client 100%

localhost:9412/traces/bb772152c661d80a?serviceName=message-client 2 minutes ago

Duration: 134.038s

Services: 2

Depth: 3

Total Spans: 7

Expand All

Collapse All

Filter Service ...

hippo-service x2

parrot-service-8081 x6

Services

- hippo-service

1.250s : http:/rent

26.808s

53.615s

- hippo-service

131.152s : httpOf feign.retry

.

parrot-service-8081

19.238s : http:/fee

.

parrot-service-8081

7.017s : http:/fee

.

parrot-service-8081

20.526s : http:/fee

.

parrot-service-8081

6.140s : http:/fee

.

parrot-service-8081

109.033s : http:/fee

.



Rent Service

Payment Service

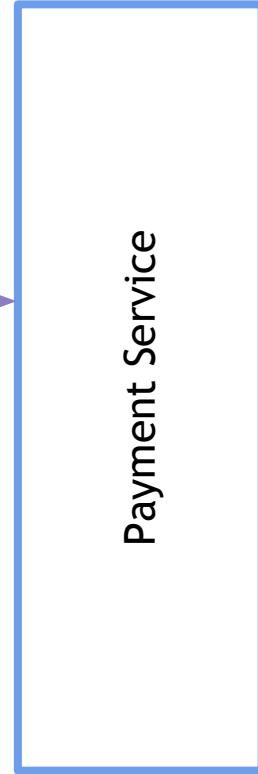
TraceId = X
SpanId = A

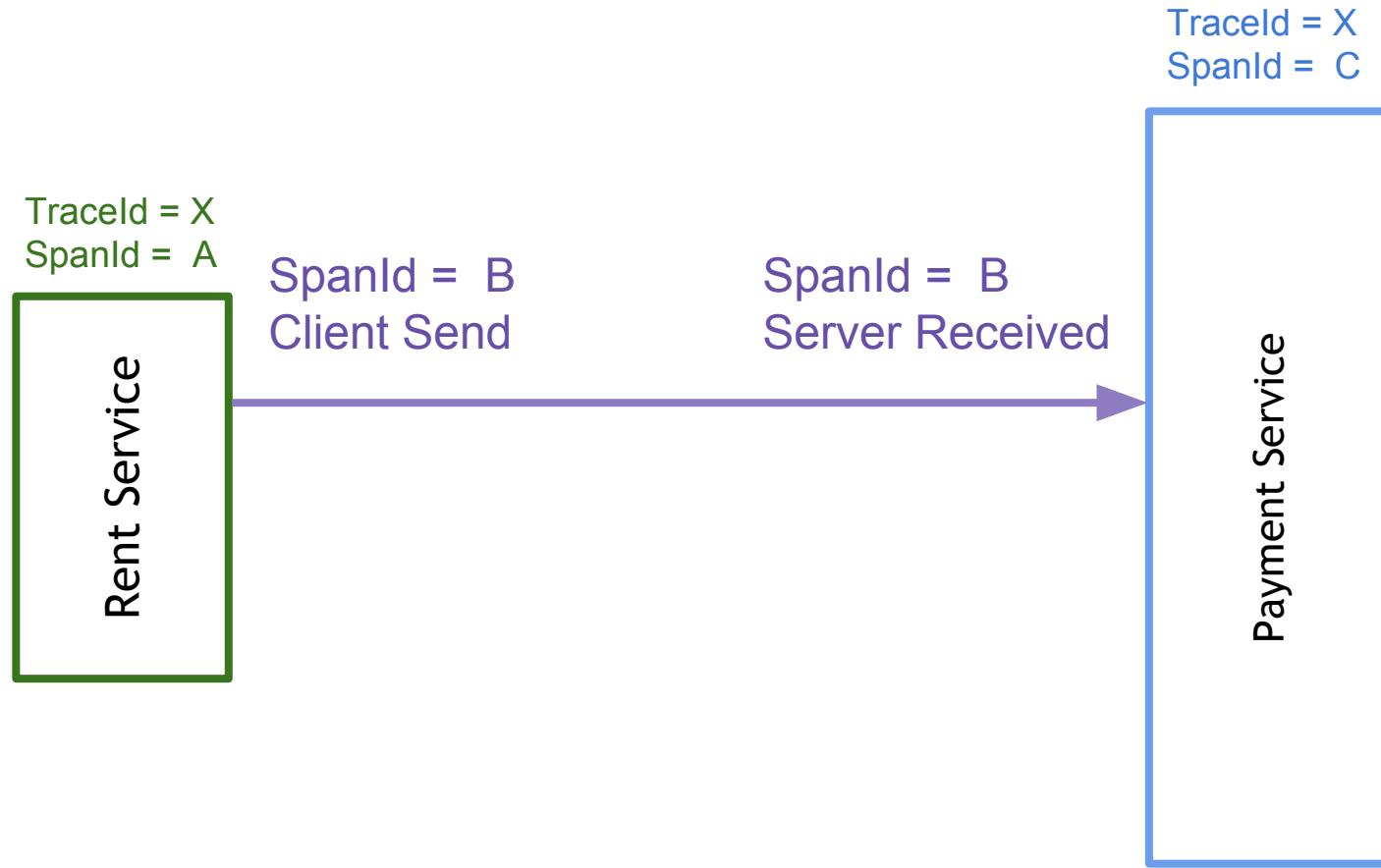


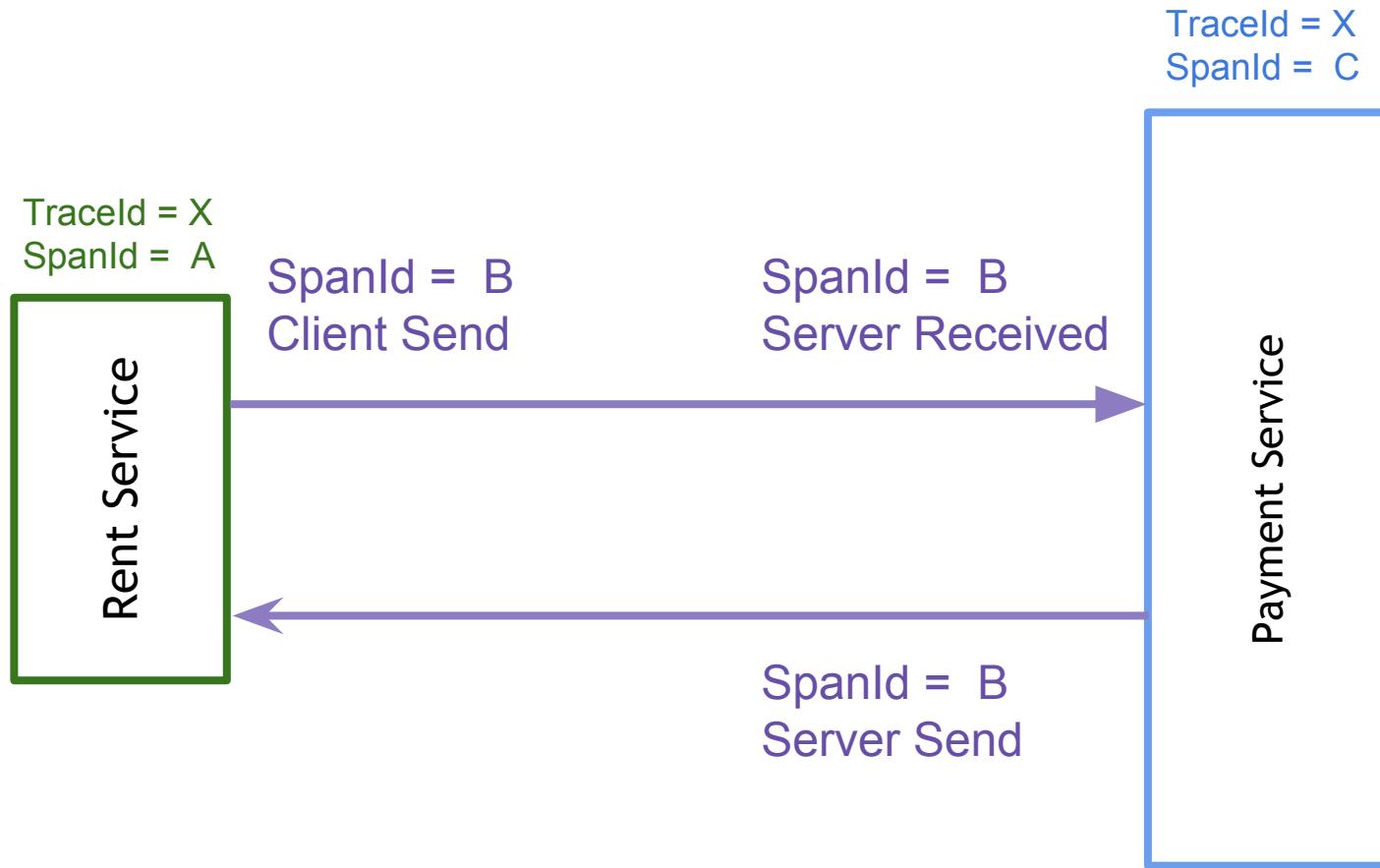
SpanId = B
Client Send

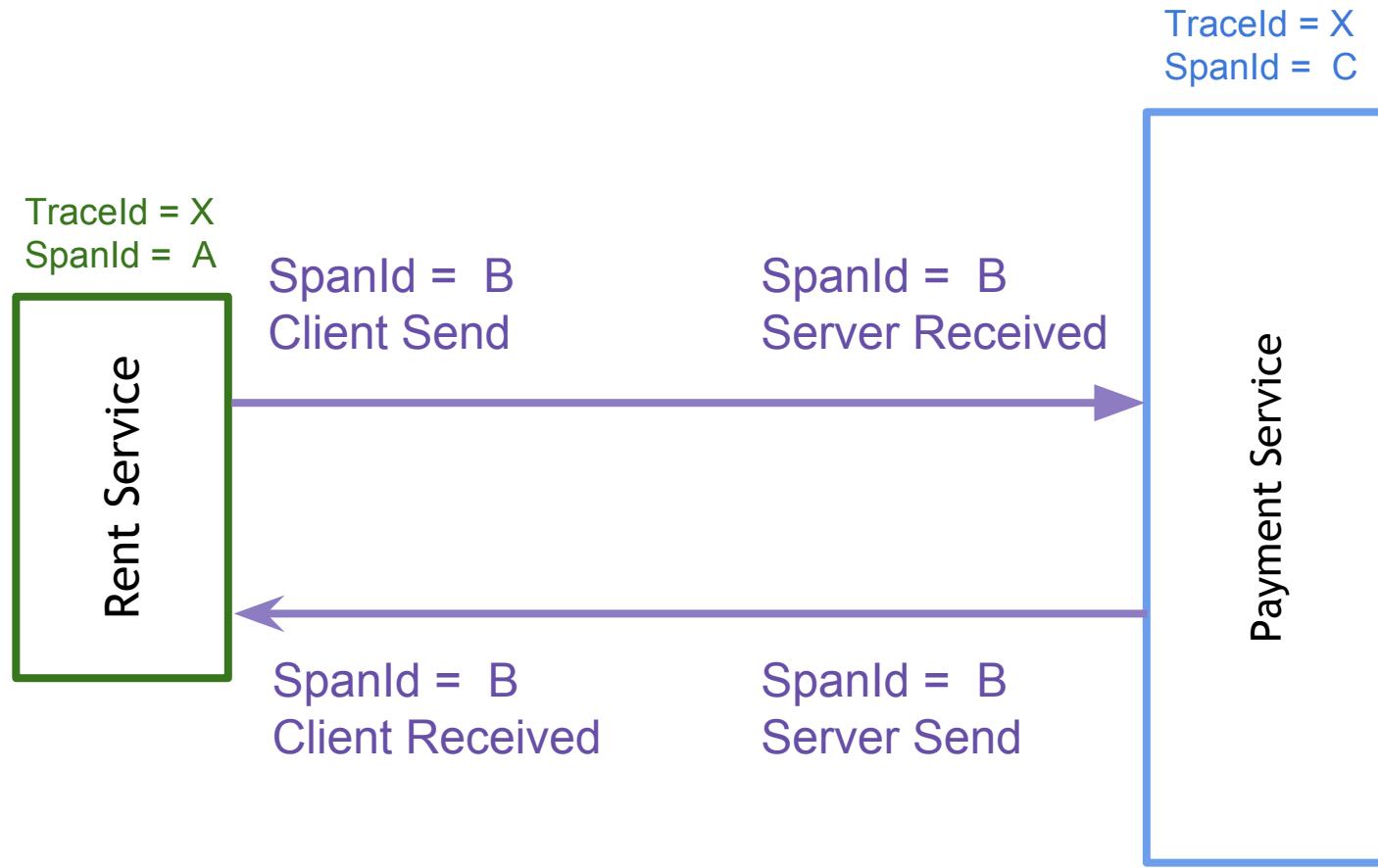


Payment Service









Yo yoman							Zn zipkin	
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	Sl sleuth
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing
------------------	-----------------------------	-------------------------------------	----------------------------	------------------	---------------------------	-------------------------	-------------------------	------------------------------

Yo yoman							Zn zipkin	
Lz lazybones	Ss start.spring.io	Ja java				Eu eureka	Ed etcd	Sl sleuth
Sb spring boot	Hx hystrix	Gy groovy	Py python		Cl consul	Zk zookeeper	Ms mesos	Kb Kubernetes
Th thrift	Ar Artifactory	An ansible	Gr gradle	Ad asciidoc	Dr docker	Au aurora	Ma marathon	Ch chronos

So SOA	Di Data Isolation	Cp Centralization paradox	Ls LSD principle	Dd DDD	Co Conway's Law	Sd Smart Docs	Sl Smart Libs	Ds Dynamic Sharing
------------------	-----------------------------	-------------------------------------	----------------------------	------------------	---------------------------	-------------------------	-------------------------	------------------------------

Yo							Zn	
yoman							zipkin	
Lz	Ss	Ja				Eu	Ed	Sl
lazybones	start.spring.io	java				eureka	etcd	sleuth
Sb	Hx	Gy	Py		?	Zk	Ms	Kb
spring boot	hystrix	groovy	python		consul	zookeeper	mesos	Kubernetes
Th	Ar	An	Gr	A	Dr	Au	Ma	Ch
thrift	Artifactory	ansible	gradle	asciidoc	docker	aurora	marathon	chronos

So	Di	Cp	Ls	Dd	Co	Sd	Sl	Ds
SOA	Data Isolation	Centralization paradox	LSD principle	DDD	Conway's Law	Smart Docs	Smart Libs	Dynamic Sharing

Простить, потому что

Простить, потому что

- 1. Архитектура – функция от множества переменных**

Простить, потому что

1. Архитектура – функция от множества переменных
2. Архитектура – результат эволюции на протяжении времени

Простить, потому что

1. Архитектура – функция от множества переменных
2. Архитектура – результат эволюции на протяжении времени
3. Принципы должны быть “**вечны**”, а инструменты **актуальны и эффективны**

Придерживайтесь принципов

1. SOA принципы живы

Придерживайтесь принципов

1. SOA принципы живы
2. Принцип LSD

Придерживайтесь принципов

1. SOA принципы живы
2. Принцип LSD
3. Изоляция данных делает жизнь приятнее

Придерживайтесь принципов

1. SOA принципы живы
2. Принцип LSD
3. Изоляция данных делает жизнь приятнее
4. Парадокс централизации

Придерживайтесь принципов

1. SOA принципы живы
2. Принцип LSD
3. Изоляция данных делает жизнь приятнее
4. Парадокс централизации
5. Планируй ресурсы динамически

Links

Лекция Жени Кривошеева про архитектуру:

https://www.youtube.com/watch?v=_Kex5hwGE-w

Пример Smart-библиотеки:

<https://github.com/lavcraft/grpc-spring-boot-starter>

Пример реализации “умной документации”:

<https://github.com/aatarasoff/documentation-plugin-demo>

Спасибо! Готовы ответить на ваши вопросы



@tolkv



@lavcraft



@aatarasoff



@aatarasoff