Философия микросервисов

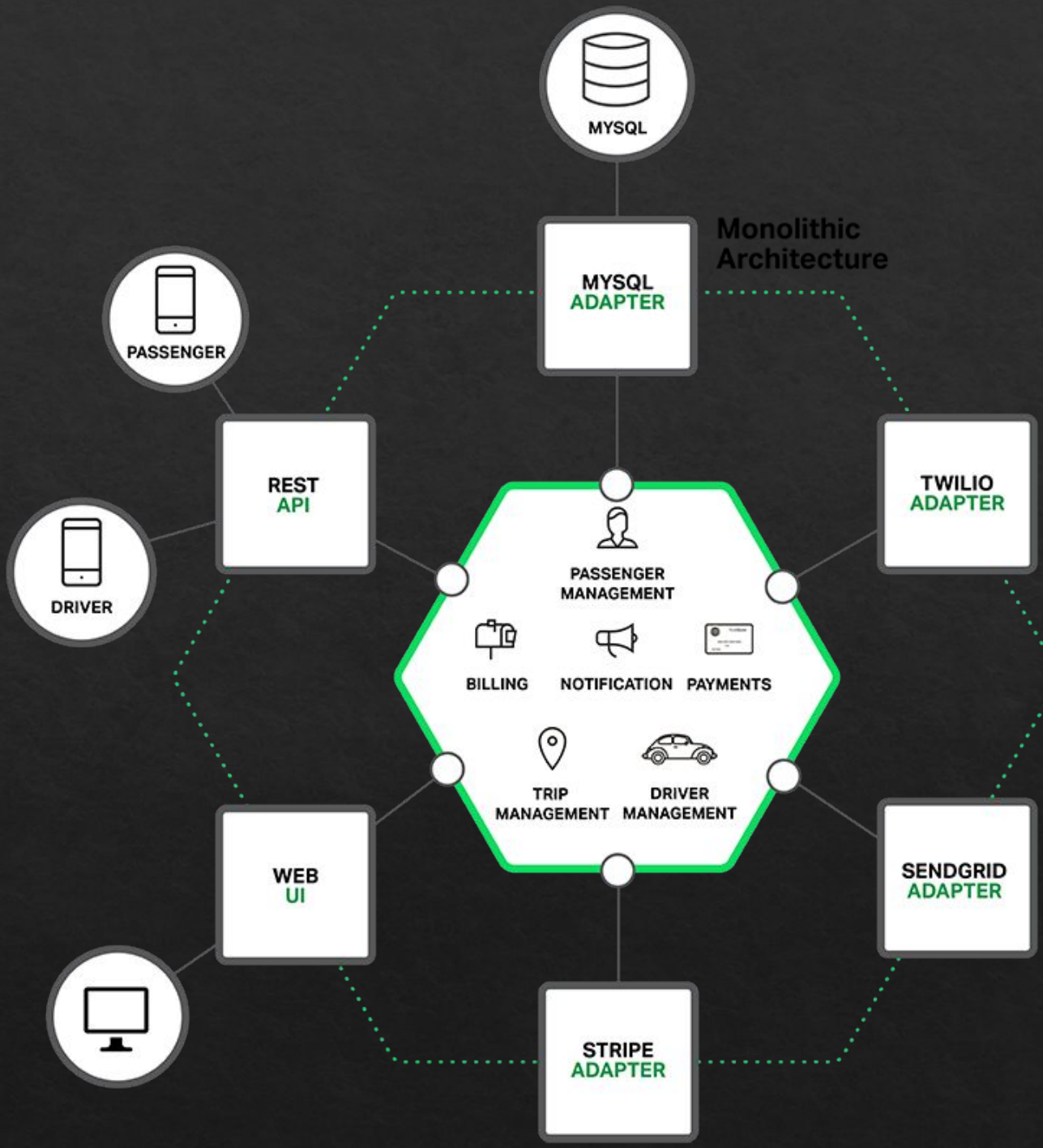
vk.com/javaenjoy

artplastika.ru

technovillage.ru

Виктор Глушенков,

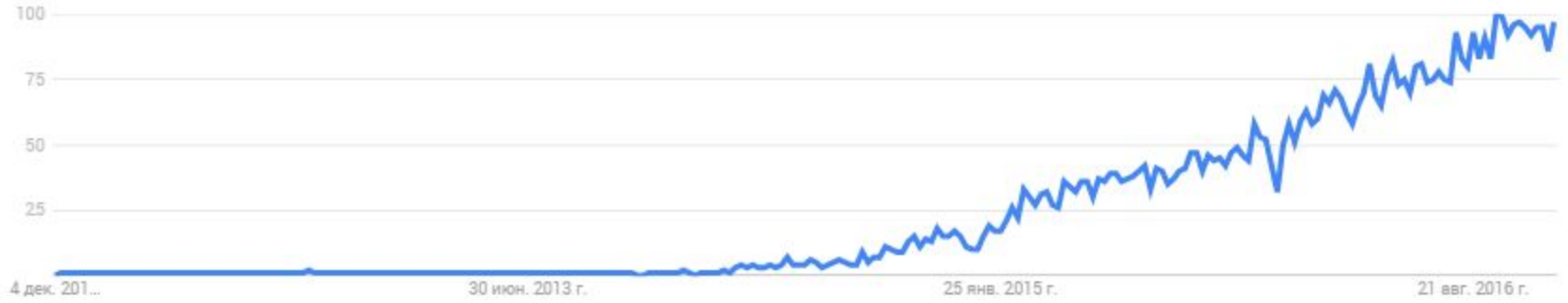
независимый разработчик, программист-прагматик с 2001 года


MYSQL
Monolithic Architecture
MYSQL ADAPTER
PASSENGER
REST API
DRIVER
TWILIO ADAPTER
PASSENGER MANAGEMENT
BILLING
NOTIFICATION
PAYMENTS
TRIP MANAGEMENT
DRIVER MANAGEMENT
WEB UI
SENDGRID ADAPTER
STRIPE ADAPTER

Недостатки

- Долгосрочная привязка к технологиям
- Сложность внесения изменений
- Сложность тестирования
- Медленное развёртывание
- Ресурсоёмкое масштабирование

Динамика популярности



Популярность по регионам

Регион

1	Индия	100
2	Австралия	74
3	Великобритания	66
4	Германия	63
5	Соединенные Штаты	60

SmallTalk

- OOP to me means only messaging, local retention and protection and hiding of state-process, and extreme late-binding of all things.
- I thought of objects being like biological cells and/or individual computers on a network, only able to communicate with messages
- However, doing encapsulation right is a commitment not just to abstraction of state, but to eliminate state oriented metaphors from programming.

Object-Oriented Design

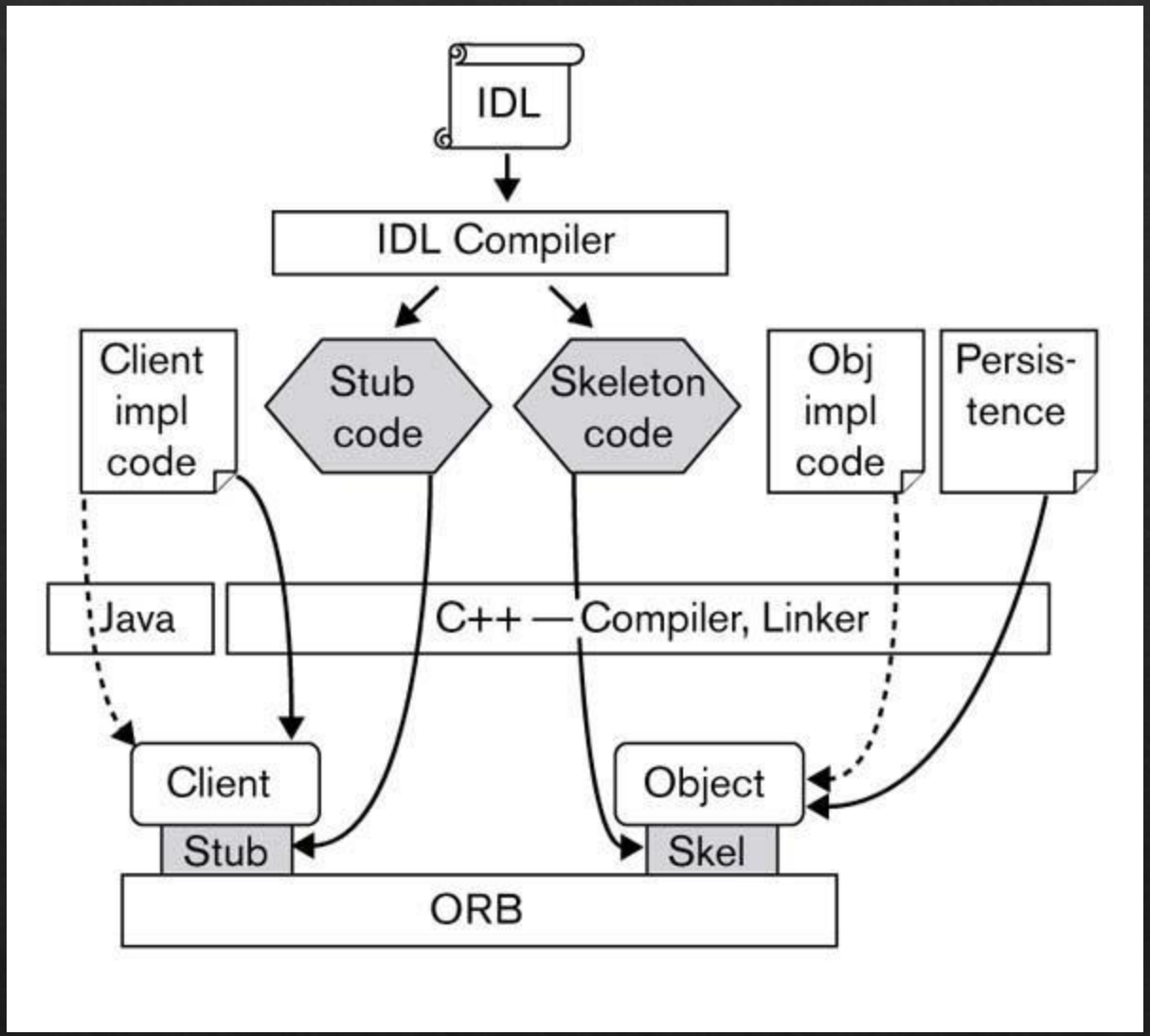
- Decomposition & Abstraction
- Single Responsibility Principle
- Open/Closed Principle
- Low Coupling (связанность)
- High Cohesion (зацепление)
- Encapsulation

SOLID

GRASP

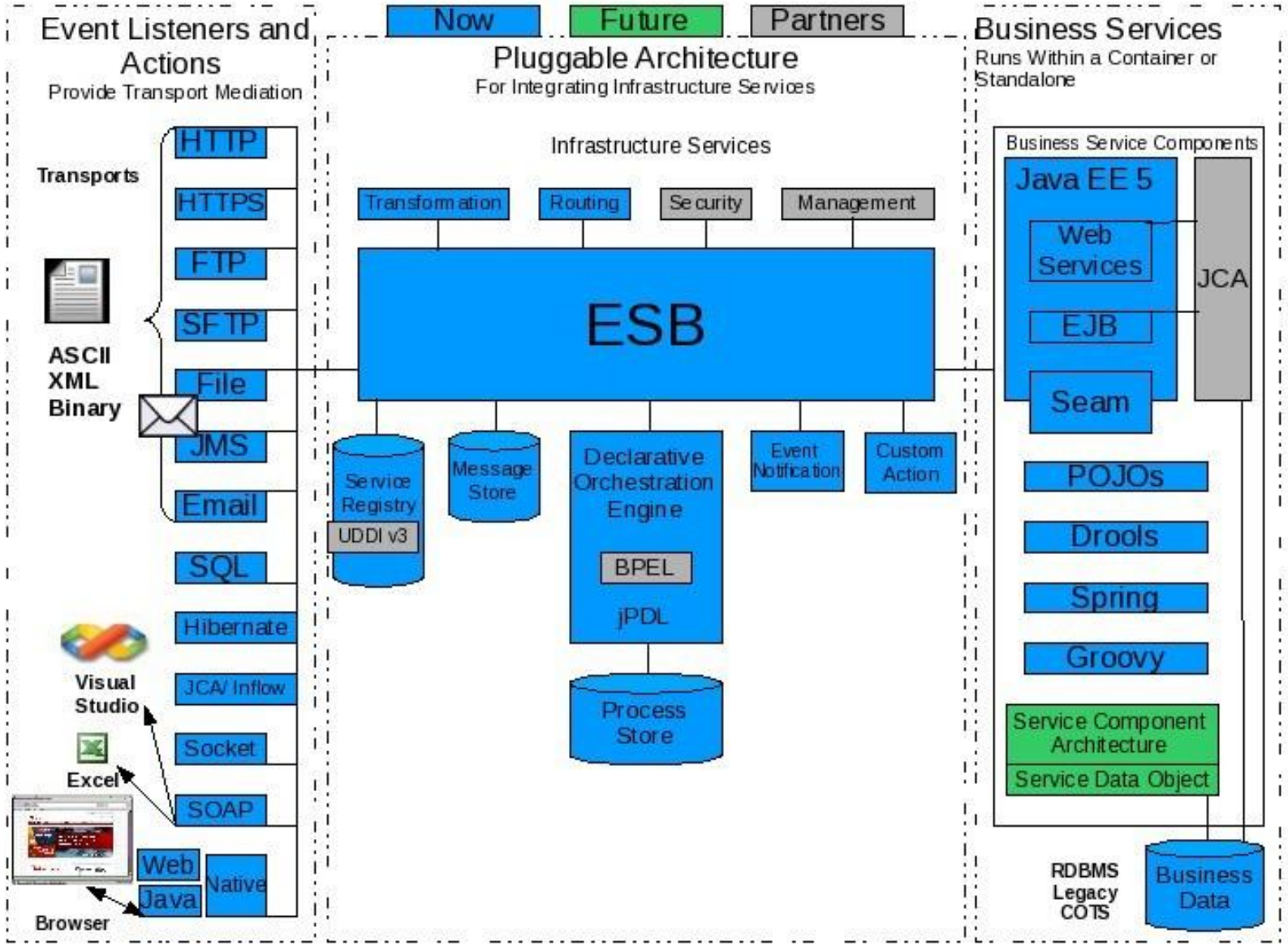
CORBA / RMI

CORBA is particularly important in large organizations, where many systems must interact with each other, and legacy systems can't yet be retired. CORBA provides the connection between one language and platform and another.


IDL
IDL Compiler
Client impl code
Stub code
Skeleton code
Obj impl code
Persis-tence
Java
C++ — Compiler, Linker
Client
Stub
Object
Skel
ORB

SOA / ESB / JBI

- For some SOA is about exposing software through web services
- For some SOA implies an architecture where applications disappear
- For some SOA is about allowing systems to communicate over some form of standard structure (usually XML based) with other applications
- For some SOA is all about using (mostly) asynchronous messaging to transfer documents between different systems.

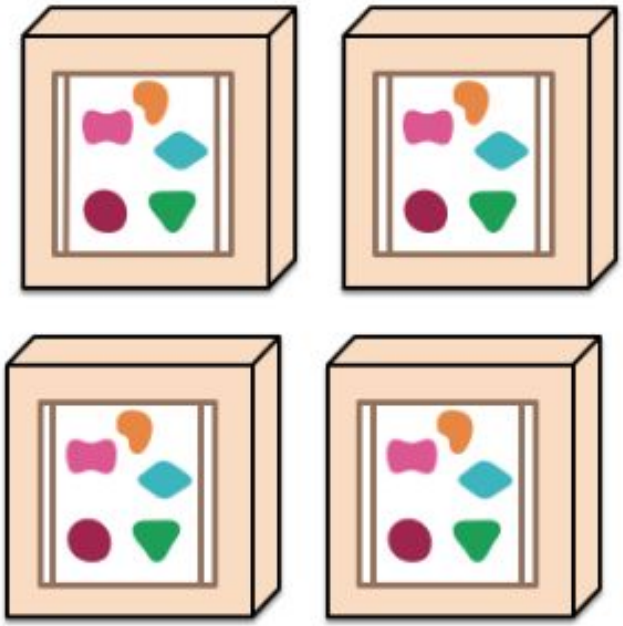

Event Listeners and Actions
Provide Transport Mediation
Transports
HTTP
HTTPS
FTP
SFTP
File
JMS
Email
SQL
Hibernate
JCA/ Inflow
Socket
SOAP
Web
Java
Native
ASCII
XML
Binary
Visual Studio
Excel
Browser
Now
Future
Partners
Pluggable Architecture
For Integrating Infrastructure Services
Infrastructure Services
Transformation
Routing
Security
Management
ESB
Service Registry
UDDI v3
Message Store
Declarative Orchestration Engine
BPEL
jPDL
Process Store
Event Notification
Custom Action
Business Services
Runs Within a Container or Standalone
Business Service Components
Java EE 5
Web Services
EJB
Seam
JCA
POJOs
Drools
Spring
Groovy
Service Component Architecture
Service Data Object
RDBMS
Legacy
COTS
Business Data

OSGi

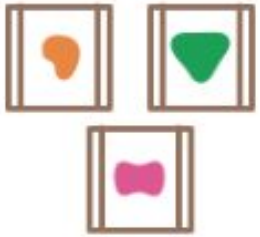
- Development model where applications are (dynamically) composed of many different (reusable) components.
- Components hide their implementations from other components while communicating through services, which are objects that are specifically shared between components.

A monolithic application puts all its functionality into a single process...

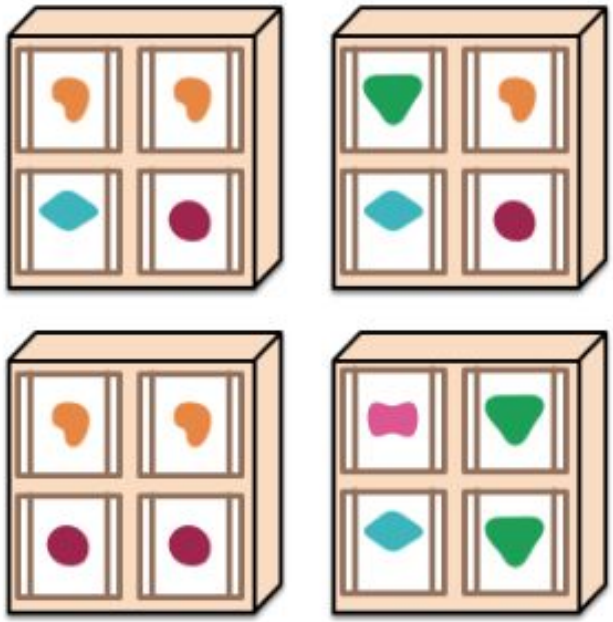
... and scales by replicating the monolith on multiple servers



A microservices architecture puts each element of functionality into a separate service...



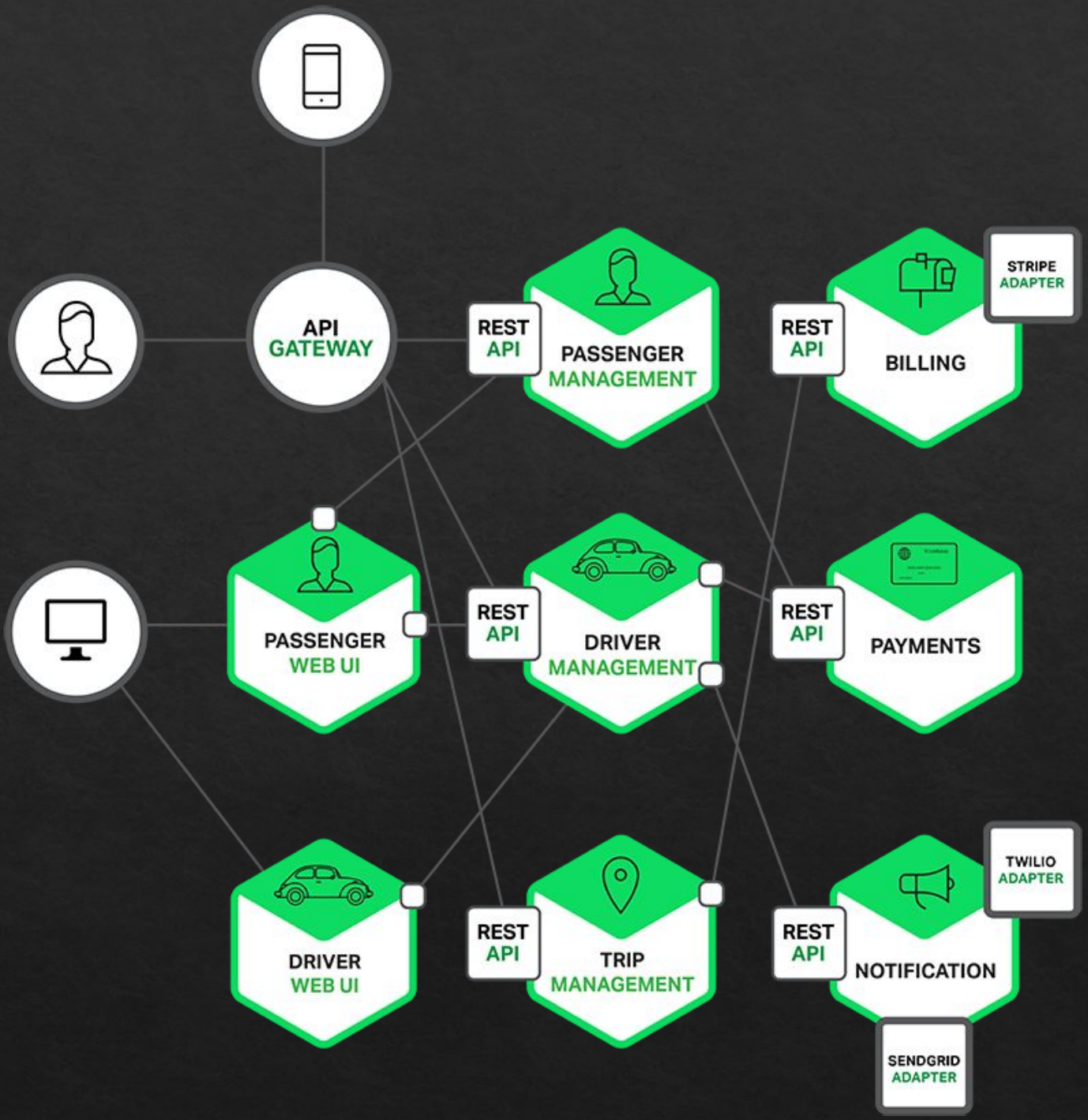
... and scales by distributing these services across servers, replicating as needed.



Микросервисы — это небольшие, автономные, совместно работающие сервисы

Свойства архитектуры

1. Разбиение на компоненты через сервисы
2. Организация вокруг потребностей бизнеса
3. Продукты, а не проекты
4. Умные приёмники и глупые каналы передачи
5. Децентрализованное руководство
6. Децентрализованное управление данными
7. Автоматизация инфраструктуры
8. Проектирование под отказ
9. Эволюционный дизайн

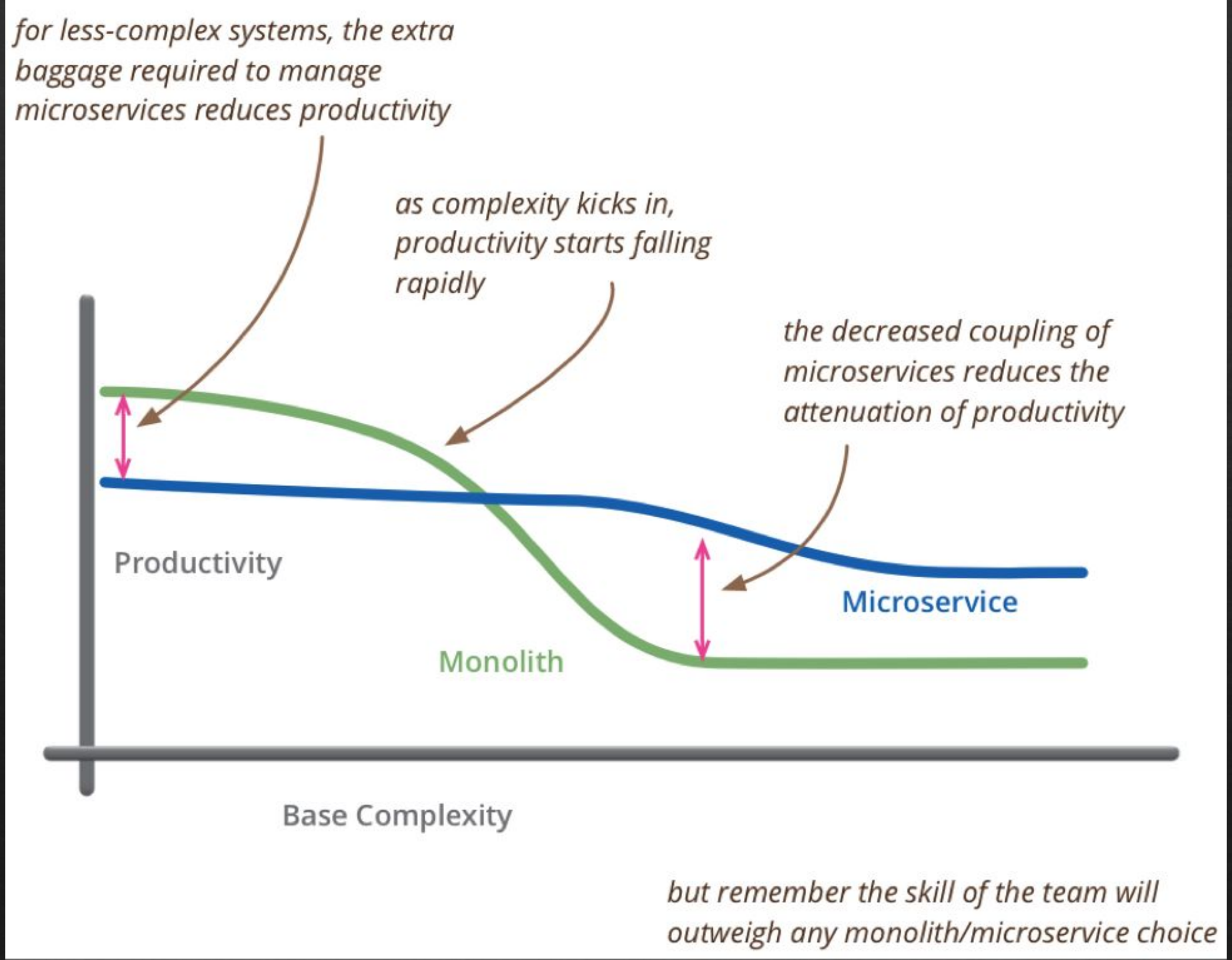

API
GATEWAY
REST
API
PASSENGER
MANAGEMENT
REST
API
BILLING
STRIPE
ADAPTER
PASSENGER
WEB UI
REST
API
DRIVER
MANAGEMENT
REST
API
PAYMENTS
DRIVER
WEB UI
REST
API
TRIP
MANAGEMENT
REST
API
NOTIFICATION
TWILIO
ADAPTER
SENDGRID
ADAPTER

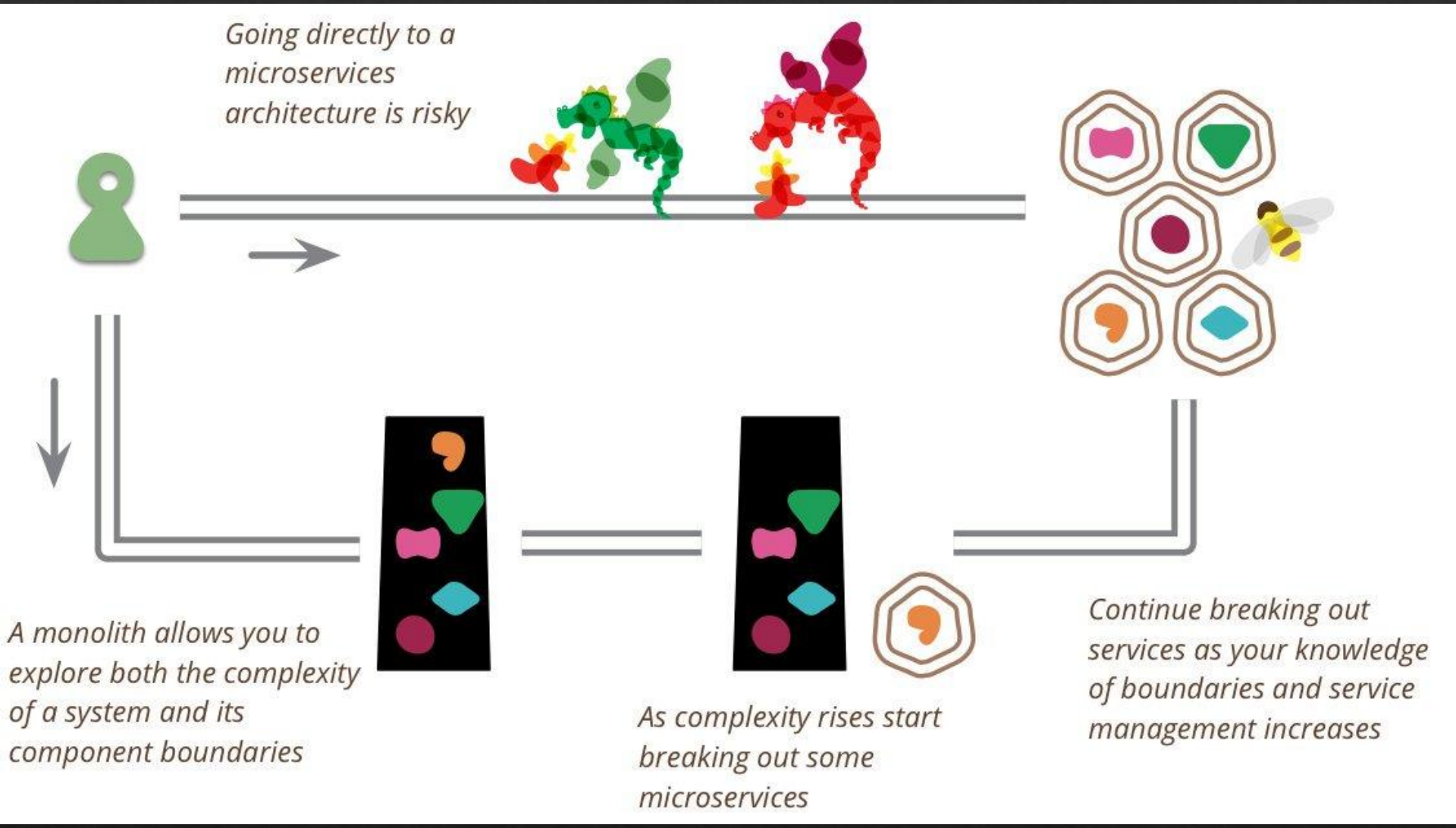
Вселяет оптимизм (Pros)

- Чёткие границы сервисов
- Простота добавления и переделки функционала
- Простота тестирования
- Решение организационных вопросов
- Отказоустойчивость
- Масштабирование
- Повторное использование
- Простота развёртывания
- Технологическая разнородность

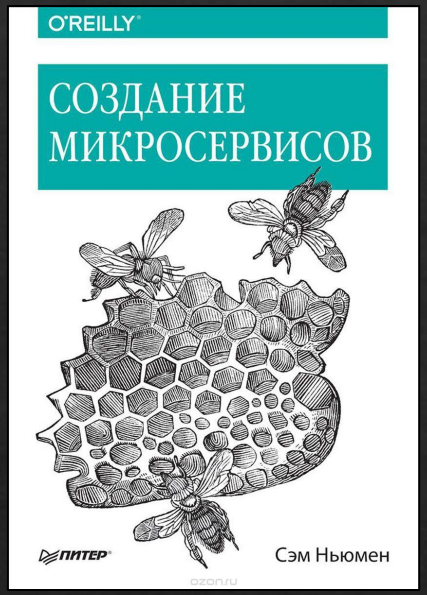
Вызывает скептицизм (Cons)

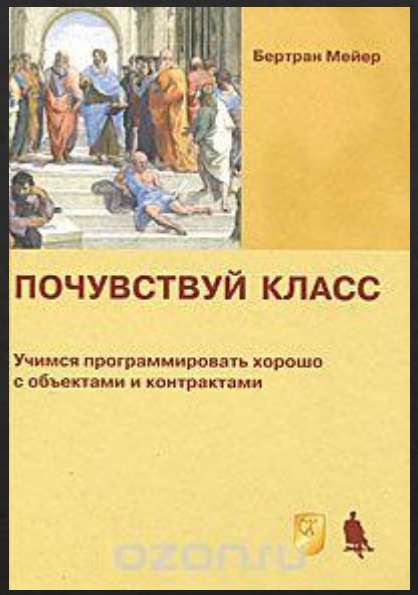
- Сложность разработки распределённых систем
- Производительность при взаимодействии
- Версионность API сервисов
- Распределённые транзакции, авторизация
- Сложность рефакторинга (границы модулей)
- Сложность сопровождения (логирование, отладка)
- Необходим высокий уровень автоматизации
- Требуются DevOps-навыки и специалисты


for less-complex systems, the extra baggage required to manage microservices reduces productivity
as complexity kicks in, productivity starts falling rapidly
the decreased coupling of microservices reduces the attenuation of productivity
Productivity
Microservice
Monolith
Base Complexity
but remember the skill of the team will outweigh any monolith/microservice choice


Going directly to a microservices architecture is risky
A monolith allows you to explore both the complexity of a system and its component boundaries
As complexity rises start breaking out some microservices
Continue breaking out services as your knowledge of boundaries and service management increases

Литература





- Microservices Resource Guide by Martin Fowler & James Lewis
- Microservices: From Design to Deployment by Chris Richardson
- «Народная библиотека» сообщества TechnoVillage — можно взять почитать книги

Что дальше?

- — Работайте, братья!
- Spring Boot
- RxJava
- Continuous Delivery
- WebPurple MeetUp: «Level-Up By Community Growth-Hacking»