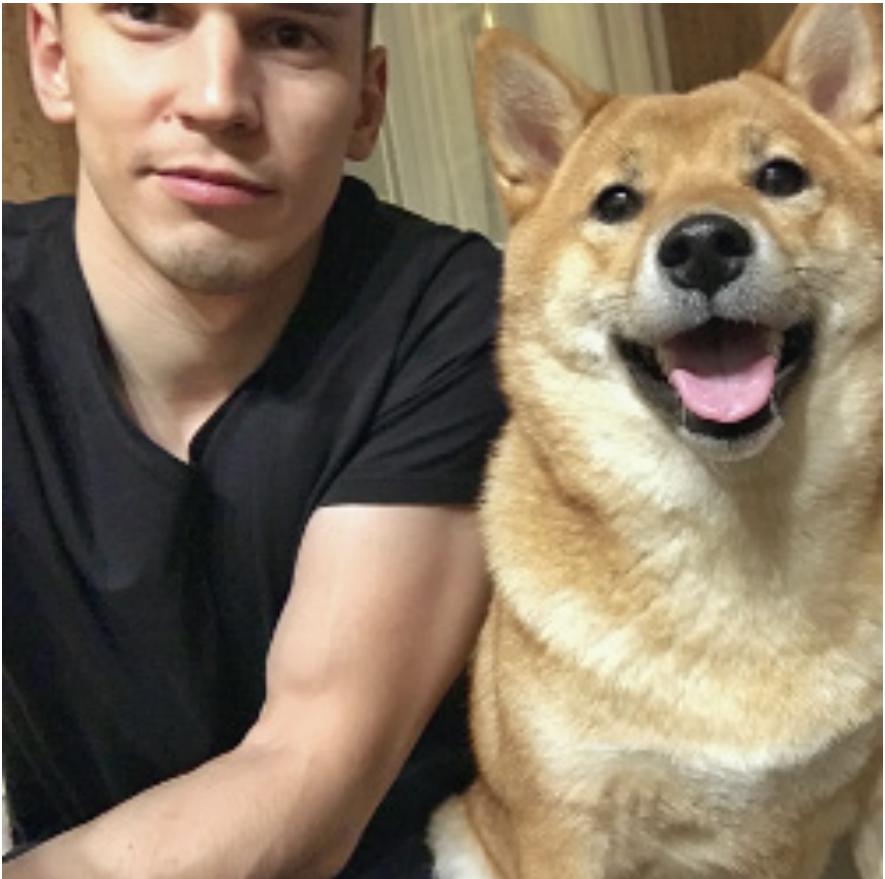




# **Чем SRE в FunBox занимаются.**

## **Часть 1: Теория**

**Алексей Мокров**  
SRE



ВЕЩАЕТ

# Алексей Мокров

Я/Мы SRE

---

СВЯЗАТЬСЯ СО МНОЙ



@a.mokrov



@mstrleo

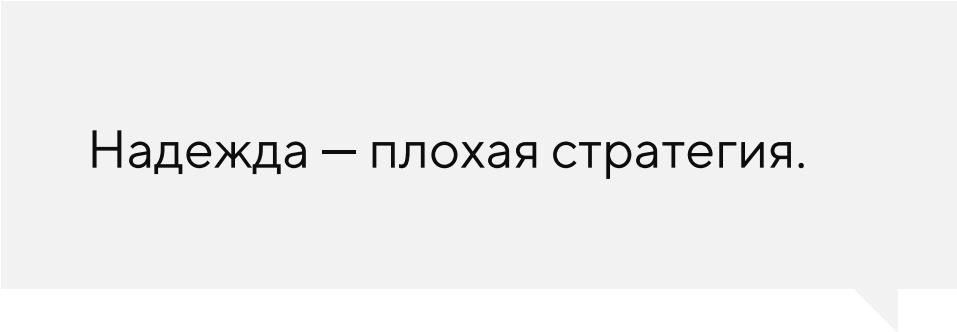


@mstrleo

# **SRE = Site Reliability Engineering**

обеспечение надежности информационных систем

# **SRE = Site Reliability Engineering**



Надежда – плохая стратегия.

Традиционная поговорка SRE

# **SRE = Site Reliability Engineering**

Что такое обеспечение надежности информационных систем?  
Я могу дать простое объяснение: SRE – это то, что происходит,  
когда вы просите программиста спроектировать команду  
службы эксплуатации.

Бенджамин Трейнор Слосс

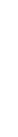
# **SRE = Site Reliability Engineering**

**SRE** – инженерная дисциплина, направленная на устойчивое достижение организациями необходимого уровня надежности в их системах, службах и продуктах.

# Хронология развития SRE



2020



**Июнь 2020**

Онлайн-митап

FunBox

# Хронология развития SRE

**Июль 2018**

"The Site Reliability  
Workbook"

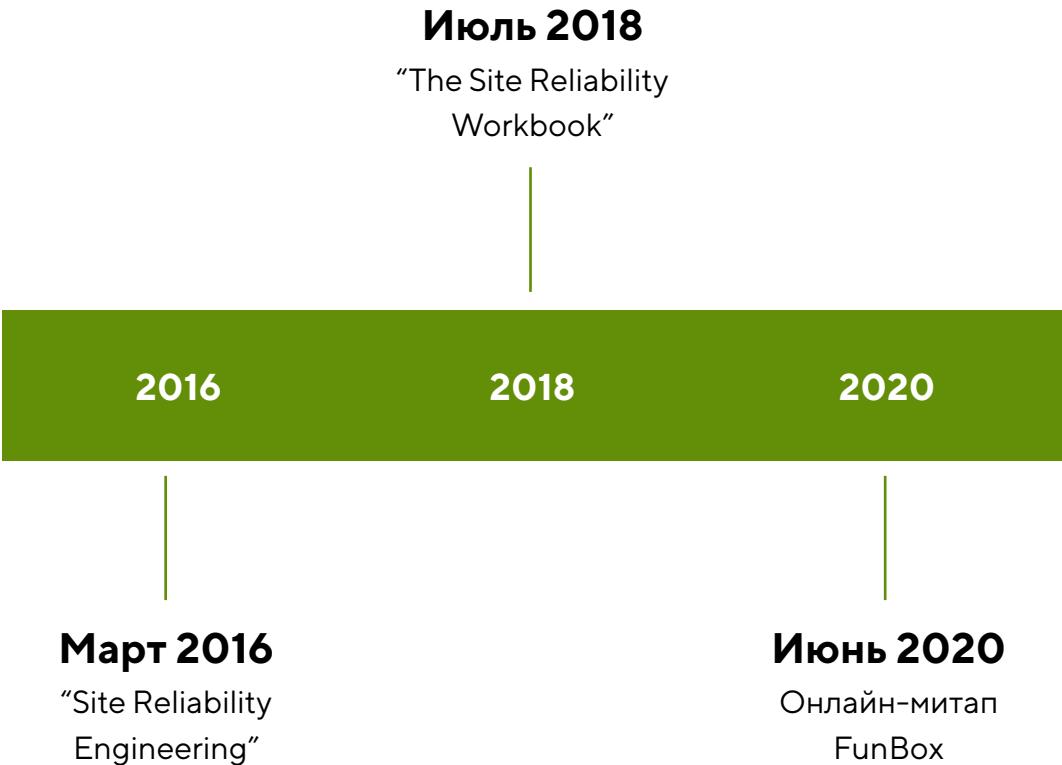
2018

2020

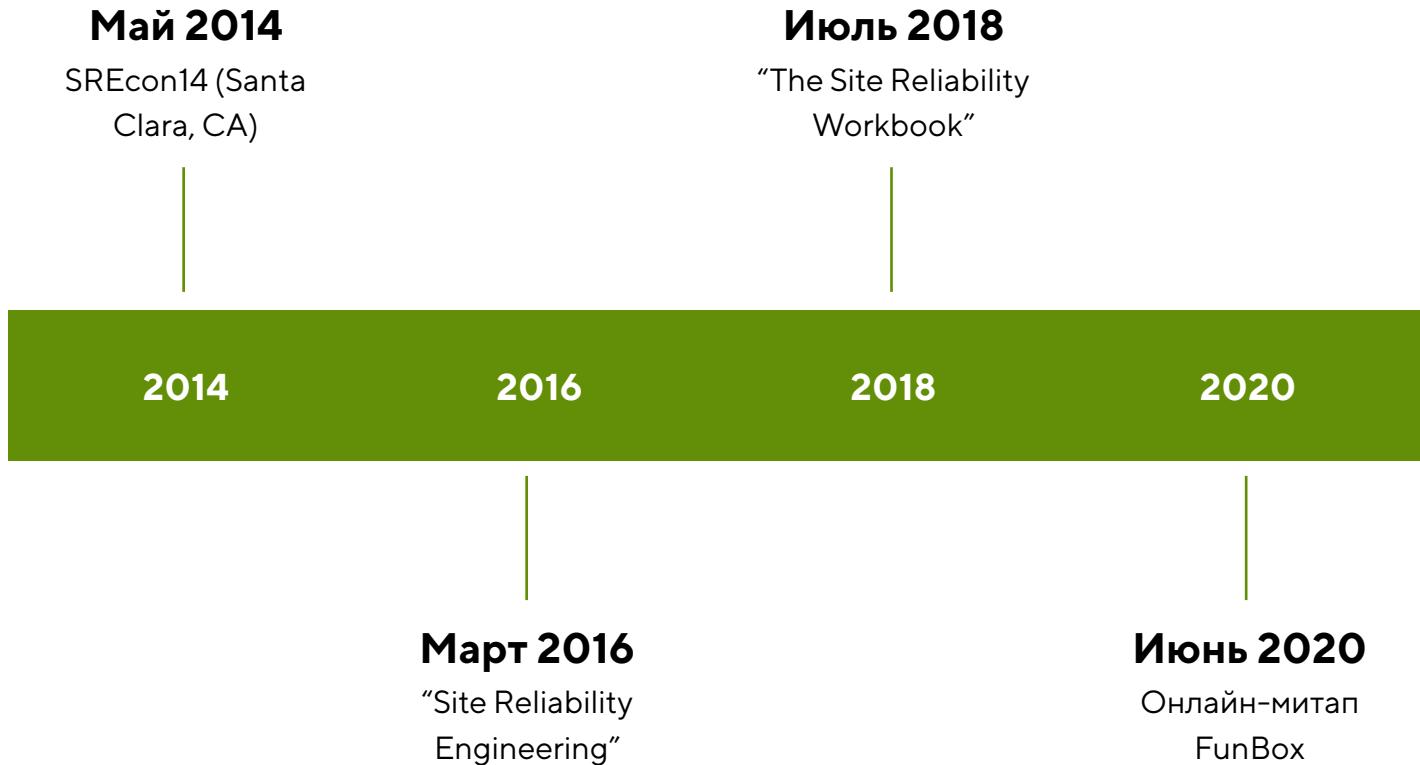
**Июнь 2020**

Онлайн-митап  
FunBox

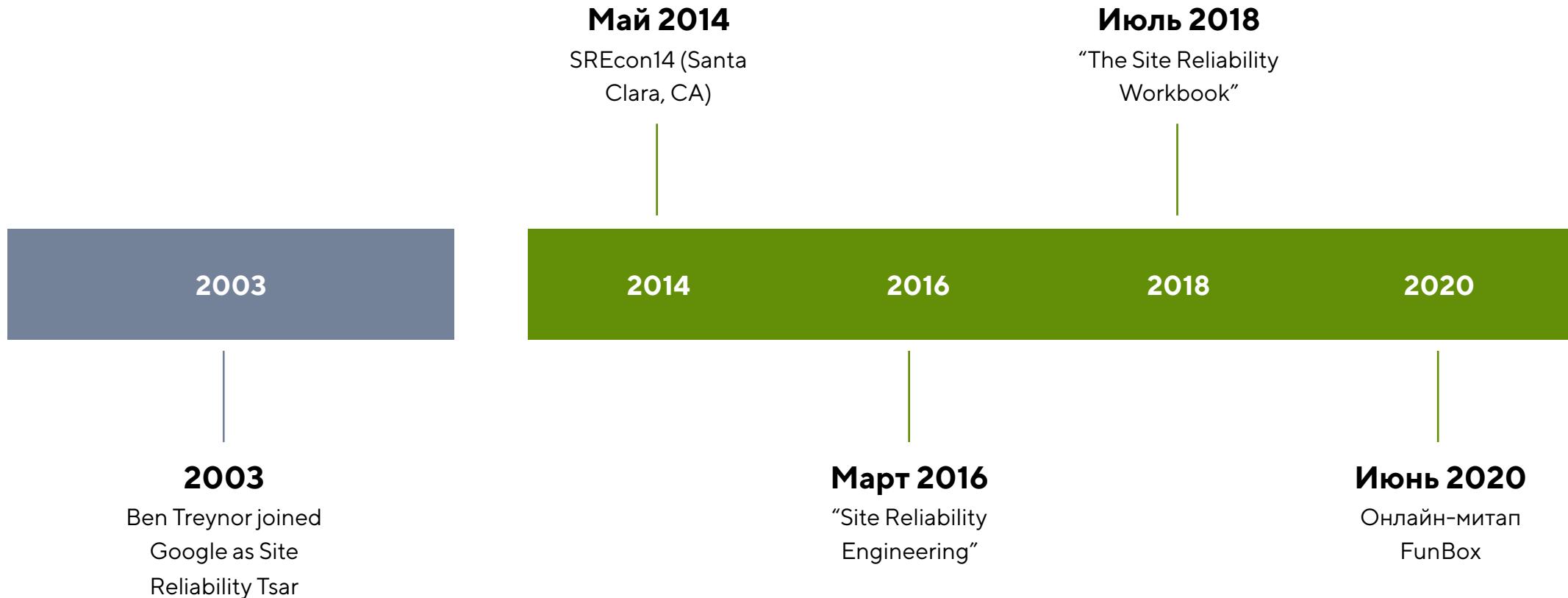
# Хронология развития SRE



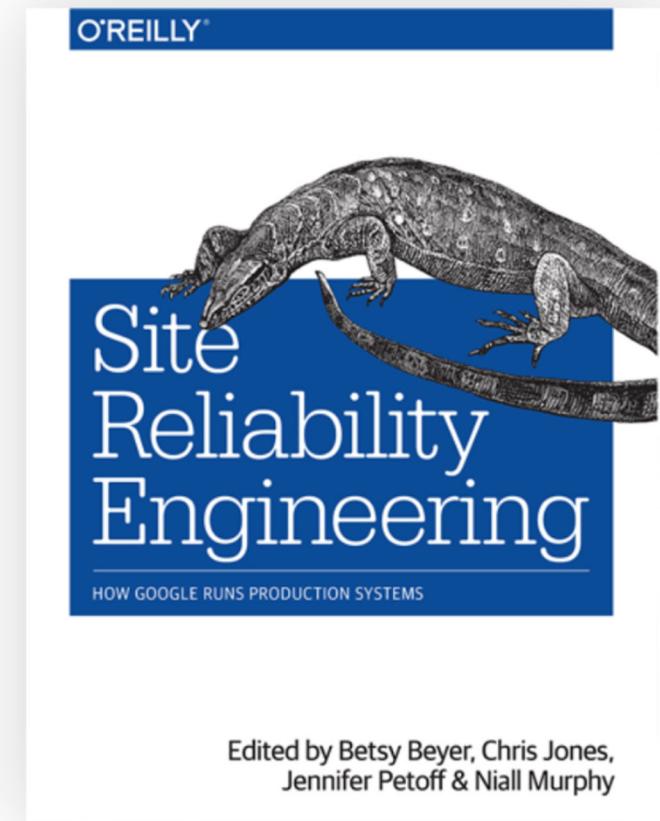
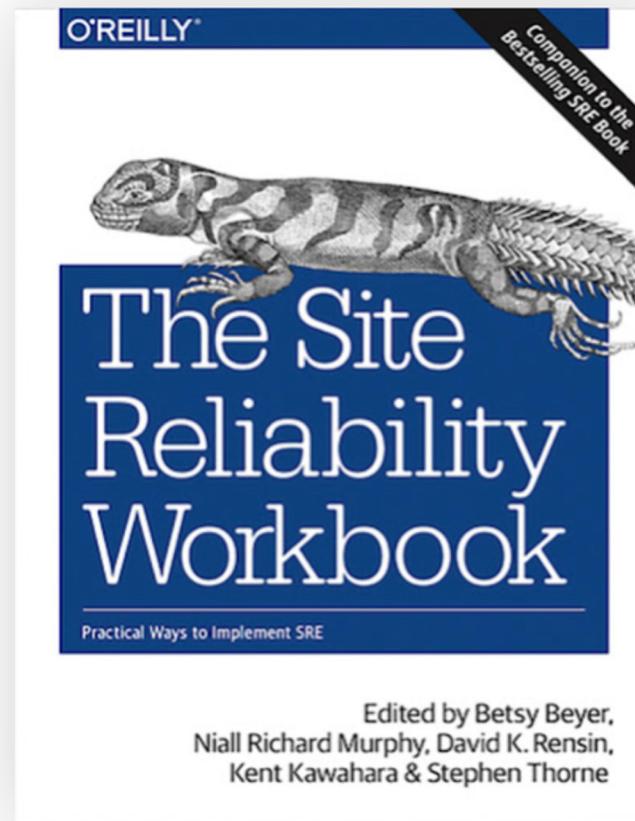
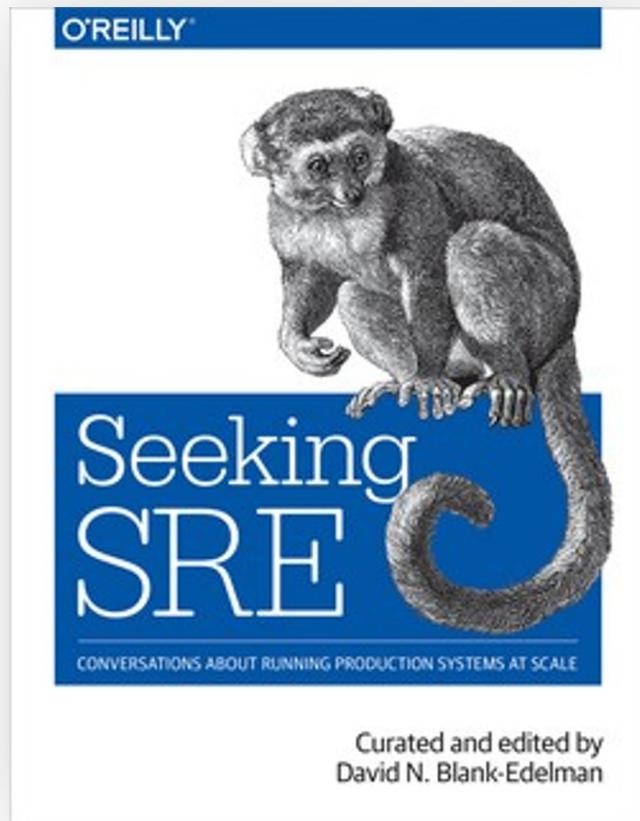
# Хронология развития SRE



# Хронология развития SRE



# Те самые книги про SRE



# Podlodka Podcast

Выпуск #138. SRE



ЗАПИСЫВАЕТ

## Дмитрий Масленников

Разработчик и архитектор больших нагруженных проектов.  
Работал SRE в Google в команде поиска. В настоящее время  
Head of SRE в Tinkoff

# SRE по версии Wikipedia

**Site Reliability Engineering (SRE)** is a discipline that incorporates aspects of [software engineering](#) and applies them to infrastructure and operations problems. The main goals are to create [scalable](#) and [highly reliable](#) software systems. According to Ben Treynor, founder of [Google's Site Reliability Team](#), SRE is "what happens when a software engineer is tasked with what used to be called [operations](#)".<sup>[1]</sup>

## Contents [hide]

- 1 [Roles](#)
  - 1.1 [DevOps vs SRE](#)
- 2 [See also](#)
- 3 [References](#)
- 4 [External links](#)

## Roles [edit]

A site reliability engineer (SRE) will spend up to 50% of their time doing "ops" related work such as issues, on-call, and manual intervention. Since the [software](#) system that an SRE oversees is expected to be highly automatic and self-healing, the SRE should spend the other 50% of their time on development tasks such as new features, scaling or automation. The ideal site reliability engineer candidate is either a software engineer with a good administration background or a highly skilled system administrator with knowledge of coding and automation.

## DevOps vs SRE [edit]

*Main article: [DevOps](#)*

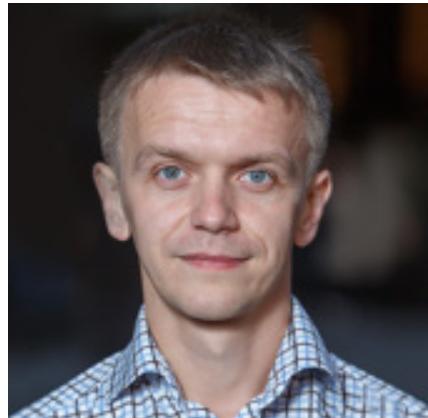
Coined around 2008, DevOps is a philosophy of cross-team empathy and business alignment. It's also been associated with a practice that encompasses automation of manual tasks, [continuous integration](#) and [continuous delivery](#). SRE and DevOps share the same foundational principles. SRE is viewed by many (as cited in the Google SRE book) as a "specific implementation of DevOps with some [idiosyncratic](#) extensions." SREs, being developers themselves, will naturally bring solutions that help remove the barriers between development teams and operations teams.

TL;DR

# **DevOps vs SRE**

# Podlodka Podcast

Выпуск #104. DevOps

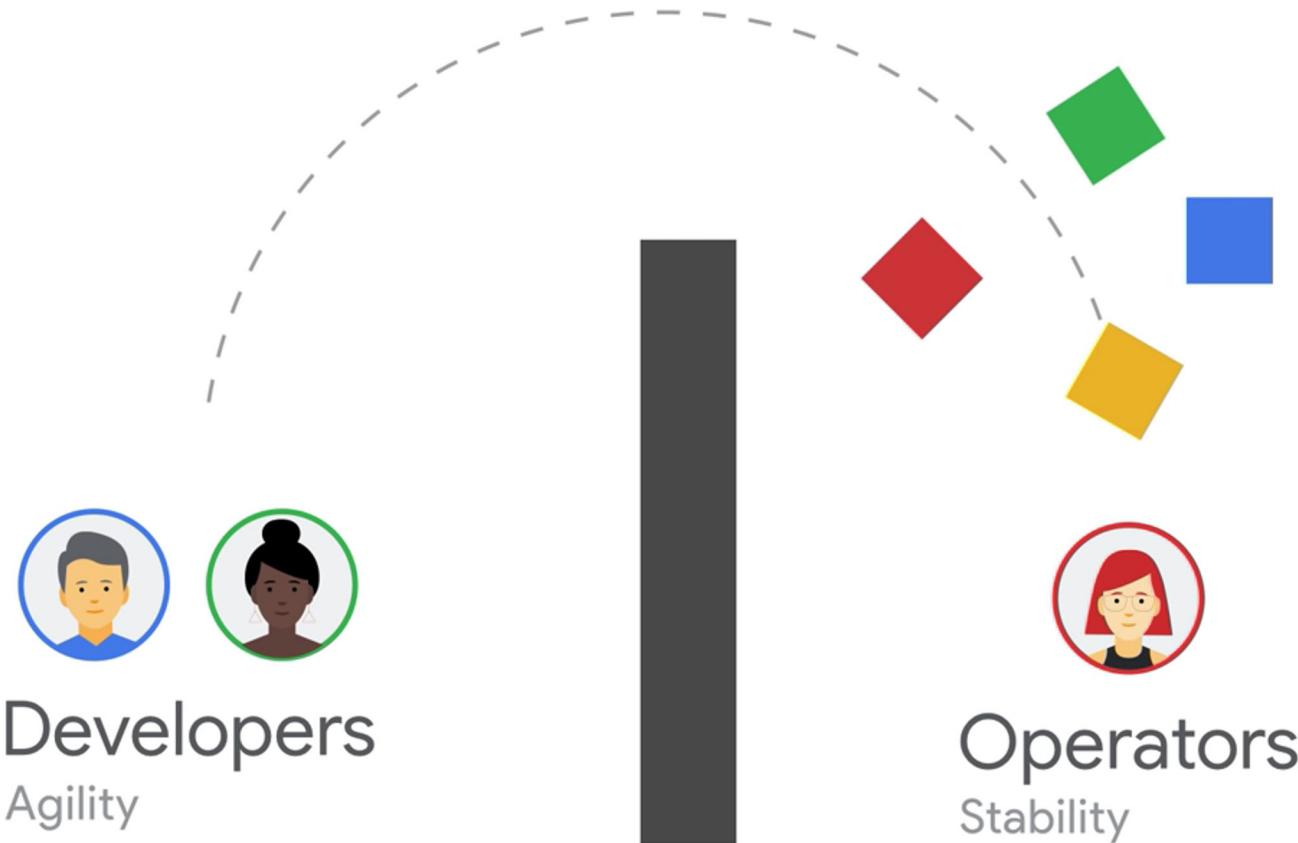


ЗАПИСЫВАЕТ

## Иван Евтухович

В прошлом разработчик (php, ruby, .Net).  
Был организатором конференции RailsClub. Вел подкаст  
Devops Deflope. В настоящее время CEO Express 42

# Что нужно знать о DevOps / Проблематика



# **Что нужно знать о DevOps / Проблематика**

- Модель обучающихся организаций.

# Что нужно знать о DevOps / Проблематика

- Модель обучающихся организаций.
- Agile-манифест.

# Что нужно знать о DevOps / Проблематика

- Модель обучающихся организаций.
- Agile-манифест.
- Теория бережливого производства.

# Что нужно знать о DevOps / Проблематика

- Модель обучающихся организаций.
- Agile-манифест.
- Теория бережливого производства.
- Теория ограничений.

# Что нужно знать о DevOps / Проблематика

- Модель обучающихся организаций.
- Agile-манифест.
- Теория бережливого производства.
- Теория ограничений.
- Производственная система компании Toyota.

# Что нужно знать о DevOps / Проблематика

- Модель обучающихся организаций.
- Agile-манифест.
- Теория бережливого производства.
- Теория ограничений.
- Производственная система компании Toyota.
- Непрерывная поставка.

# Что нужно знать о DevOps / 2009. DevOpsDays

Увидев на Velocity Conference 2009 презентацию Джона Олспоу и Пола Хаммонда “10+ Deploys Per Day: Dev and Ops Cooperation at Flickr”, я убедился: у меня есть единомышленники. Поэтому я решил организовать первую конференцию DevOpsDays и так случайно создал новый термин – DevOps.

Патрик Дюбуа

# Что нужно знать о DevOps / 2012. Техрадар

## DevOps | Technology Radar

- Август 2010. Оценка.
- Январь 2011. Пробный период.
- Март 2012. Рекомендован к внедрению.

# **Что нужно знать о DevOps / 2013. DevOps топологии**

What Team Structure is Right for DevOps to Flourish?

Posted on [October 22, 2013](#) by Matthew Skelton.

# Что нужно знать о DevOps / 2013. DevOps топологии

What Team Structure is Right for DevOps to Flourish?

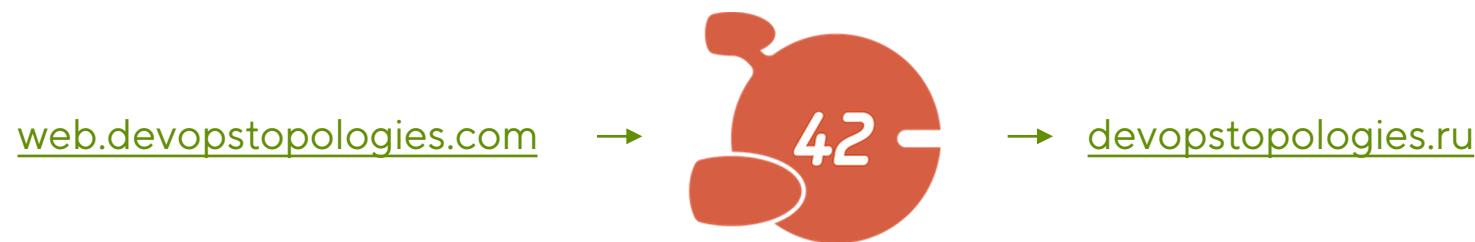
Posted on October 22, 2013 by Matthew Skelton.



# Что нужно знать о DevOps / 2013. DevOps топологии

What Team Structure is Right for DevOps to Flourish?

Posted on October 22, 2013 by Matthew Skelton.

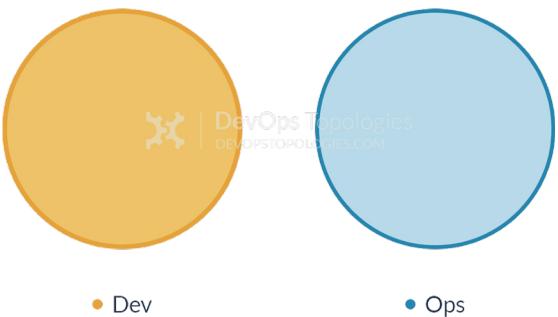


«Для эффективного взаимодействия разработки и эксплуатации в разных организациях нужны разные структуры команд».

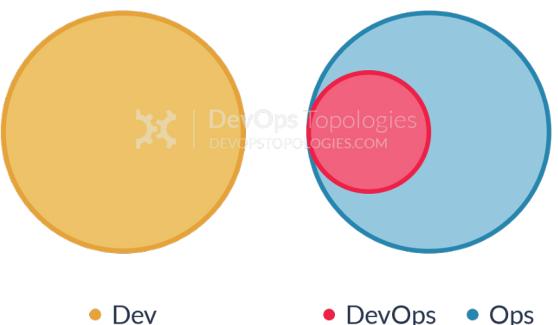
# DevOps топологии

## Антипаттерны DevOps

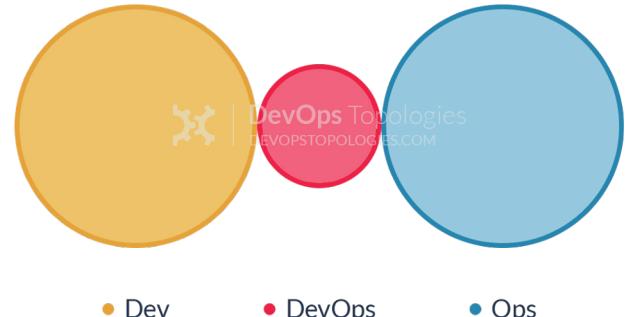
Антипаттерн А:  
Изоляция Разработки  
и Эксплуатации



Антипаттерн Е:  
Переименованные  
сисадмины



Антипаттерн В:  
Отдельная DevOps  
команда



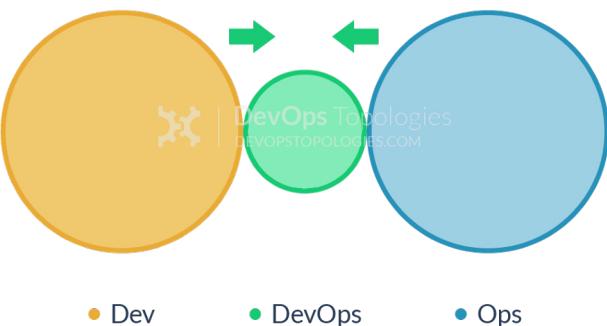
# DevOps топологии

DevOps структуры команд

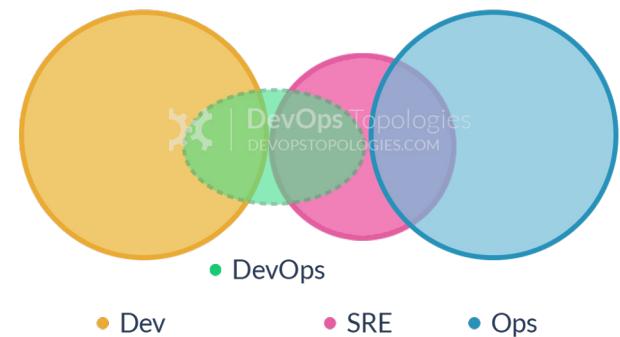
**2 тип:**  
Полностью  
распределённые  
функции эксплуатации



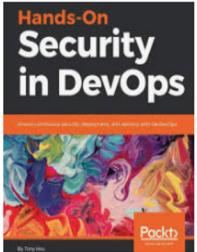
**7 тип:**  
SRE Команда  
(Модель Google)



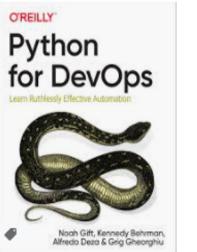
**5 тип:**  
DevOps команда  
с ограниченным  
сроком действия



# Что нужно знать о DevOps / Книги



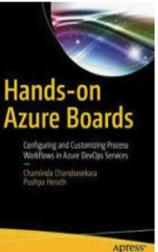
Hands-On Security in DevO...  
wish4book.com



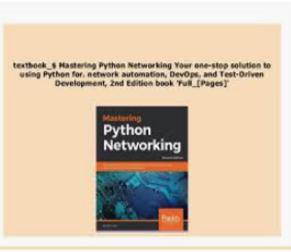
Python for DevOps: Learn...  
amazon.com



DevOps for Azure Appli...  
allitebooks.org



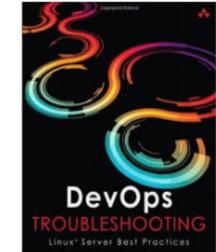
PDF Free] Hands-on Az...  
pinterest.ru



Download\_[P.d.f]^@@ Mastering Python Net...  
es.slideshare.net



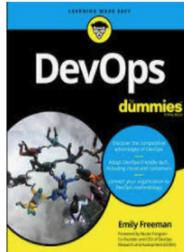
Learning DevOps  
packtpub.com · В наличии



DevOps Archives - Page 2 o...  
allfullbooks.com



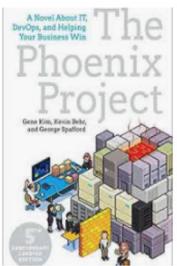
Download or Read Onli...  
sites.google.com



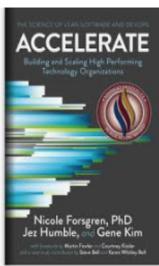
PDF [Books] Devops for D...  
skyreadingbooks-xayom504...



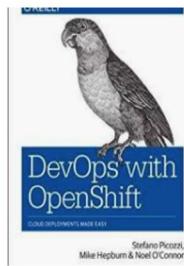
Mobile DevOps by Rohin...  
perlego.com



FREE~DOWNLOAD The ...  
medium.com



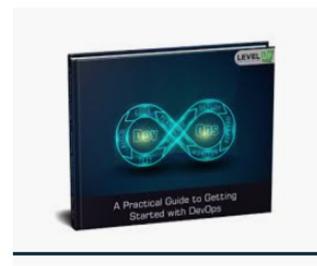
Accelerate | by Nicole Fo...  
itrevolution.com



DevOps with OpenShift [PD...  
programmer-books.com



v.Vcx) PDF Download The ...  
yumpu.com



DevOps Tutorial PDF Book - Level Up  
level-up.one



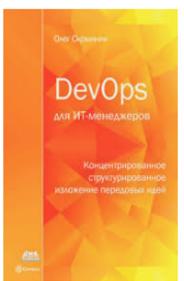
Книга: "Безопасный DevO...  
labirint.ru



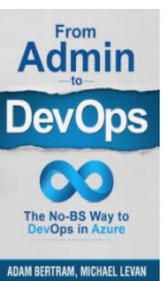
Azure DevOps Interview Que...  
dotnettricks.com



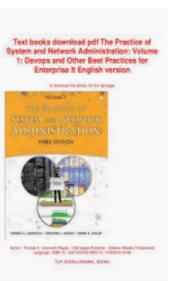
PDF] [EPUB] Hands-On Soft...  
pdfget.com



DevOps для ИТ-менедж...  
monster-book.com



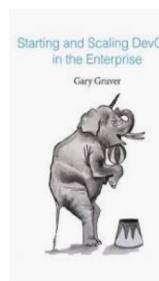
From Admin to ... by Ad...  
leanpub.com



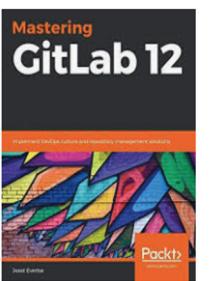
Text books download pd...  
slideshare.net



Implementing Modern Dev...  
cfalevel2bookspdfreadownlo...



How to get a PDF of Gary...  
quora.com



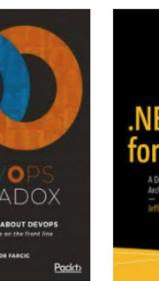
Read Ebooks Mastering Git...  
zedfkilaf.ygto.com



Hands-On DevOps for Archit...  
packtpub.com · В наличии



DevOps for Digital Lea...  
link.springer.com

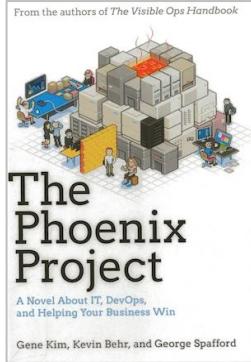


DevOps Paradox by Vik...  
perlego.com

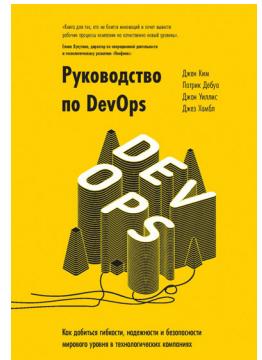
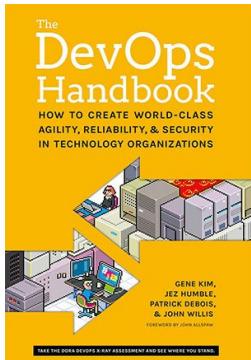


.NET DevOps for Azure....  
itbook.download

# Что нужно знать о DevOps / Книги



Первая публикация: январь 2013 г.  
Переведено на русский и опубликовано: 2015 г.  
Авторы: Джин Ким, Кевин Бер, Джордж Спаффорд



Первая публикация: 2016 г.  
Переведено на русский и опубликовано: 2018 г.  
Авторы: Джин Ким, Джез Хамбл, Патрик Дебуа, Джон Уиллес

# **DevOps**

**DevOps** – это сочетание разработки (Dev) и эксплуатации (Ops). Это объединение людей, процессов и технологий, позволяющее постоянно предоставлять преимущества клиентам.

# **DevOps**

**DevOps** – это сочетание разработки (Dev) и эксплуатации (Ops). Это объединение людей, процессов и технологий, позволяющее постоянно предоставлять преимущества клиентам.

**DevOps** – это набор практик для повышения эффективности процессов разработки и эксплуатации программного обеспечения за счёт их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации.

# **DevOps**

**DevOps** – это сочетание разработки (Dev) и эксплуатации (Ops). Это объединение людей, процессов и технологий, позволяющее постоянно предоставлять преимущества клиентам.

**DevOps** – это набор практик для повышения эффективности процессов разработки и эксплуатации программного обеспечения за счет их непрерывной интеграции и активного взаимодействия профильных специалистов с помощью инструментов автоматизации.

**DevOps** – методология разработки программного обеспечения, нацеленная на активное взаимодействие и интеграцию специалистов по разработке и специалистов по информационно-техническому обслуживанию.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.
- SRE – инженерная дисциплина, посвящённая надёжности.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.
- SRE – инженерная дисциплина, посвящённая надёжности.
- SRE может быть названием должности, а DevOps – нет.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.
- SRE – инженерная дисциплина, посвящённая надёжности.
- SRE может быть названием должности, а DevOps – нет.
- SRE содержит четкие инструкции, в DevOps они специально отсутствуют.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.
- SRE – инженерная дисциплина, посвящённая надёжности.
- SRE может быть названием должности, а DevOps – нет.
- SRE содержит чёткие инструкции, в DevOps они специально отсутствуют.
- SRE воплощает философию DevOps.

# DevOps vs SRE

- DevOps – культурное движение, возникшее из желания преодолеть разобщённость между разработкой и эксплуатацией.
- SRE – инженерная дисциплина, посвящённая надёжности.
- SRE может быть названием должности, а DevOps – нет.
- SRE содержит чёткие инструкции, в DevOps они специально отсутствуют.
- SRE воплощает философию DevOps.
- DevOps подобен интерфейсу в объектно-ориентированном программировании.

class SRE implements DevOps

# class SRE implements DevOps

SRE vs. DevOps: competing standards or close friends?

DEVOPS & SRE

## SRE vs. DevOps: competing standards or close friends?

What's the Difference Between DevOps and SRE? (class SRE implements DevOps)

SREs vs DevOps

SRE implements DevOps

**Seth Vargo**  
Developer Advocate and Product Manager

**Liz Fong-Jones**  
Site Reliability Engineer

May 8, 2018

Site Reliability Engineering (SRE) and DevOps are two trending disciplines with quite a bit of overlap. In the past, some have called SRE a competing set of practices to DevOps. But we think they're not so different after all.

What exactly is SRE and how does it relate to DevOps? Earlier this year, we ([Liz Fong-Jones](#) and [Seth Vargo](#)) launched a video series to help answer some of these questions and reduce the friction between the communities. This blog post summarizes the themes and lessons of each video in the series to offer actionable steps toward better, more reliable systems.

class SRE implements DevOps

SREs vs DevOps

SRE implements DevOps

PLAY ALL

**class SRE implements DevOps**

10 videos • 92,483 views • Last updated on Oct 9, 2018

Seth Vargo and Liz Fong-Jones go head-to-head to determine which is better: DevOps or Site Reliability Engineering (SRE).

Google Cloud Platform

SUBSCRIBE

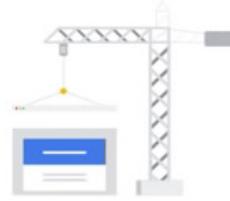
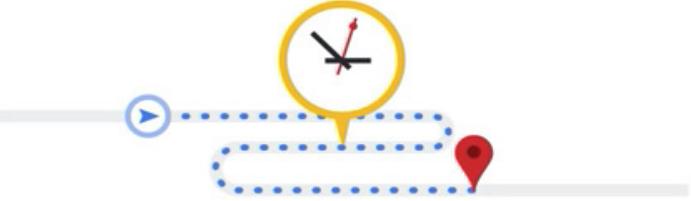
# What's the Difference Between DevOps and SRE? (class SRE implements DevOps)

1 марта 2018 года



Share ownership 	SLOs & Blameless PMs 	Reduce costs of failure 
Reduce Organization Silos 	Accept Failure as Normal 	Implement Gradual Change 
Automate this year's job away 		Measure toil and reliability 
Leverage Tooling & Automation 	Measure Everything 	



<p>Share ownership</p>  <p>Reduce Organization Silos</p> <p></p>	<p>SLOs &amp; Blameless PMs</p>  <p>Accept Failure as Normal</p> <p></p>	<p>Reduce costs of failure</p>  <p>Implement Gradual Change</p> <p></p>
<p>Automate this year's job away</p>  <p>Leverage Tooling &amp; Automation</p> <p></p>		<p>Measure toil and reliability</p>  <p>Measure Everything</p> <p></p>

# И снова Wikipedia

## DevOps vs SRE [ edit ]

*Main article: [DevOps](#)*

Coined around 2008, DevOps is a philosophy of cross-team empathy and business alignment. It's also been associated with a practice that encompasses automation of manual tasks, [continuous integration](#) and [continuous delivery](#). SRE and DevOps share the same foundational principles. SRE is viewed by many (as cited in the Google SRE book) as a "specific implementation of DevOps with some [idiosyncratic](#) extensions." SREs, being developers themselves, will naturally bring solutions that help remove the barriers between development teams and operations teams.

DevOps defines 5 key pillars of success:

1. Reduce organizational [silos](#)
2. Accept failure as normal
3. Implement gradual changes
4. Leverage tooling and automation
5. Measure everything

SRE satisfies the DevOps pillars as follows:<sup>[2]</sup>

1. Reduce organizational silos
  - SRE shares ownership with developers to create shared responsibility<sup>[3]</sup>
  - SREs use the same tools that developers use, and vice versa

# И снова Wikipedia

## DevOps vs SRE [ edit ]

Main article: [DevOps](#)

Coined around 2008, DevOps is a philosophy of cross-team empathy and business alignment. It's also been associated with a practice that encompasses automation of manual tasks, [continuous integration](#) and [continuous delivery](#). SRE and DevOps share the same foundational principles. SRE is viewed by many (as cited in the Google SRE book) as a "specific implementation of DevOps with some [idiosyncratic](#) extensions." SREs, being developers themselves, will naturally bring solutions that help remove the barriers between development teams and operations teams.

DevOps defines 5 key pillars of success:

1. Reduce organizational silos
2. Accept failure as normal
3. Implement gradual changes
4. Leverage tooling and automation
5. Measure everything

SRE satisfies the DevOps pillars as follows:<sup>[2]</sup>

1. Reduce organizational silos
  - SRE shares ownership with developers to create shared responsibility<sup>[3]</sup>
  - SREs use the same tools that developers use, and vice versa



- ([cur](#) | [prev](#)) ○ 16:24, 12 March 2018 [Sethvargo](#) ([talk](#) | [contribs](#)) . . . (5,457 bytes) (+49) . . . (*Update link attribution title*) ([undo](#))
- ([cur](#) | [prev](#)) ○ 16:17, 8 March 2018 [Sethvargo](#) ([talk](#) | [contribs](#)) . . . (5,408 bytes) (+1,572) . . . (*Made link a ref to make XLink happy*) ([undo](#)) (*Tag: Undo*)
- ([cur](#) | [prev](#)) ○ 20:10, 7 March 2018 [Sethvargo](#) ([talk](#) | [contribs](#)) m . . . (5,599 bytes) (0) . . . (*capitalize SRE*) ([undo](#))
- ([cur](#) | [prev](#)) ○ 20:09, 7 March 2018 [Sethvargo](#) ([talk](#) | [contribs](#)) . . . (5,599 bytes) (+1,763) . . . (*→DevOps vs SRE*) ([undo](#))

# И снова Wikipedia

## DevOps vs SRE [ edit ]

Main article: [DevOps](#)

Coined around 2008, DevOps is a philosophy of cross-team empathy and business alignment. It's also been associated with a practice that encompasses automation of manual tasks, [continuous integration](#) and [continuous delivery](#). SRE and DevOps share the same foundational principles. SRE is viewed by many (as cited in the Google SRE book) as a "specific implementation of DevOps with some [idiosyncratic](#) extensions." SREs, being developers themselves, will naturally bring solutions that help remove the barriers between development teams and operations teams.

DevOps defines 5 key pillars of success:

1. Reduce organizational silos
2. Accept failure as normal
3. Implement gradual changes
4. Leverage tooling and automation
5. Measure everything

SRE satisfies the DevOps pillars as follows:<sup>[2]</sup>

1. Reduce organizational silos
  - SRE shares ownership with developers to create shared responsibility<sup>[3]</sup>
  - SREs use the same tools that developers use, and vice versa

## DevOps vs SRE

Main article: [DevOps](#)

Coined around 2008, DevOps is a philosophy of cross team empathy and business alignment. It's also been associated with a practice that encompasses automation of manual tasks, [continuous integration](#) and [continuous delivery](#). SRE and DevOps share the same foundational principals. SRE is viewed by many (as cited in the Google SRE book) as a "specific implementation of DevOps with some idiosyncratic extensions." SREs, being developers themselves, will naturally bring solutions that help remove the barriers between development teams and operations teams.

## See also

- [Cloud computing](#)
- [Data center](#)
- [High availability software](#)
- [Operations management](#)
- [Operations, administration and management](#)
- [Reliability engineering](#)
- [System administration](#)

- 
- (cur | prev) ○ 16:24, 12 March 2018 Sethvargo ([talk](#) | [contribs](#)) . . . (5,457 bytes) (+49) . . . (*Update link attribution title*) ([undo](#))
  - (cur | prev) ○ 16:17, 8 March 2018 Sethvargo ([talk](#) | [contribs](#)) . . . (5,408 bytes) (+1,572) . . . (*Made link a ref to make XLink happy*) ([undo](#)) (*Tag: Undo*)
  - (cur | prev) ○ 20:10, 7 March 2018 Sethvargo ([talk](#) | [contribs](#)) m . . . (5,599 bytes) (0) . . . (*capitalize SRE*) ([undo](#))
  - (cur | prev) ○ 20:09, 7 March 2018 Sethvargo ([talk](#) | [contribs](#)) . . . (5,599 bytes) (+1,763) . . . (*→DevOps vs SRE*) ([undo](#))

# **Принципы DevOps Agile Skills Association (DASA)**

1. Customer-centric action.
2. Create with the end in mind.
3. End-to-end responsibility.
4. Cross-functional autonomous teams.
5. Continuous improvement.
6. Automate everything you can.

# SRE принципы

- Риск-менеджмент.
- Уровни качества обслуживания.
- Борьба с рутиной.
- Мониторинг.
- Автоматизация.
- Простота.

# SRE принципы

## Риск-менеджмент

- Управление рисками.
- Измерение рисков.
- Рискоустойчивость сервисов.
- Бюджет ошибок.

# SRE принципы

## Уровни качества обслуживания

Service Level...

- Indicator.
- Objectives.
- Agreements.

# SRE принципы

## Борьба с рутиной

Если во время выполнения стандартных операций в работу системы должен вмешаться оператор, то у вас есть ошибка.

Google SRE

# SRE принципы

## Мониторинг

Простейший мониторинг:

- Время отклика.
- Величина трафика.
- Уровень ошибок.
- Степень загруженности.

# SRE принципы

## Автоматизация

- Единообразие.
- Повторяемость.
- Быстрое восстановление.
- Быстродействие.
- Экономия времени.

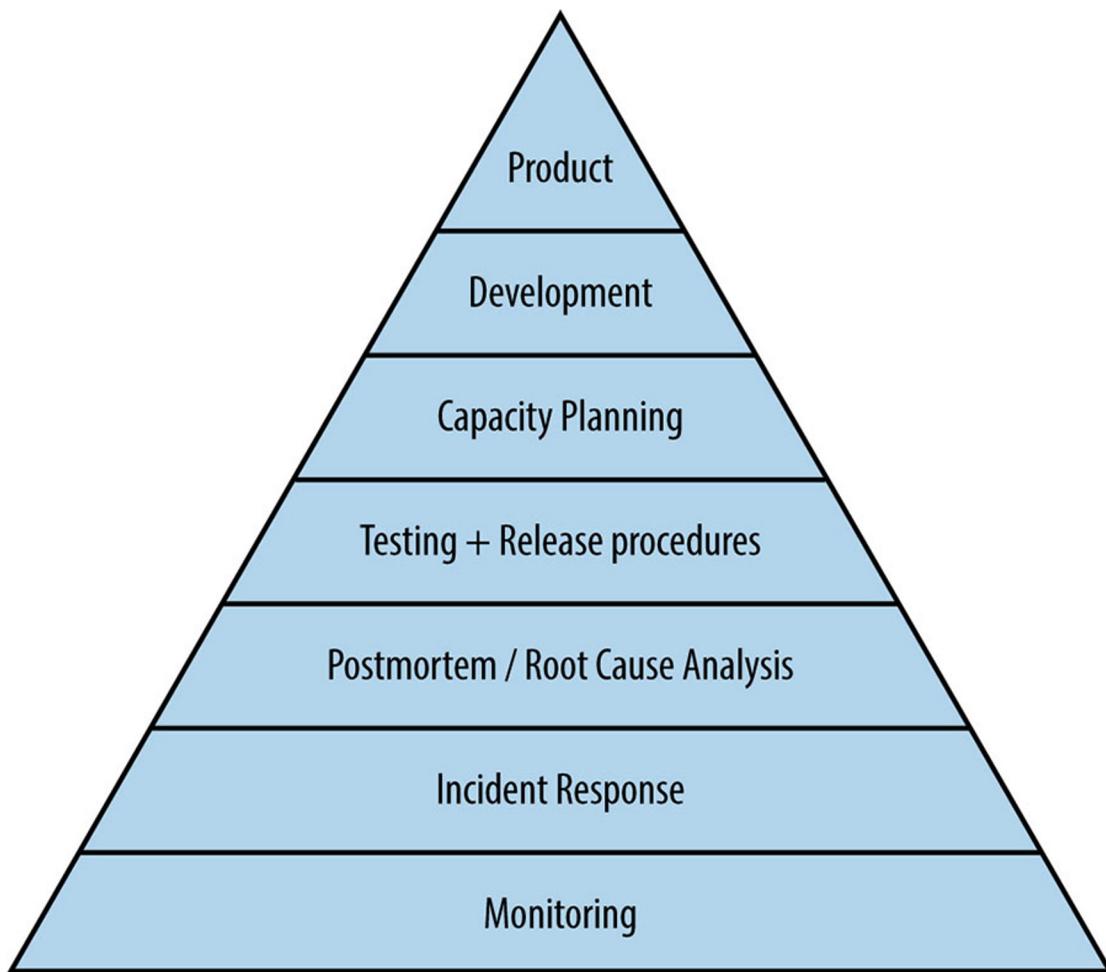
# SRE принципы

## Простота

Как видно, совершенство достигается не тогда, когда уже  
ничего прибавить, но когда уже ничего нельзя отнять.

Антуан де Сент-Экзюпери, 1939

# SRE принципы



# SRE принципы

## Оповещения

- Актуальность.
- Приоритетность.
- Уникальность.
- Правомерность.
- Достоверность.

# SRE принципы

## On-call дежурство

- Доступность 24/7.
- Контроль времени реакции.
- Оценка эффективности.

# SRE принципы

## Траблшутиng

- Реагирование.
- Диагностика проблемы.
- Восстановление рабочего состояния системы.
- Устранение проблемы.
- Создание отчёта.

# SRE принципы

## Постмортемы

- Безобвинительная культура.
- Хронология событий.
- Скриншоты.
- Наблюданное влияние.
- Ожидания.
- Результаты.
- Рекомендации.

# SRE принципы

## Тестирование

- Тесты в рабочей среде.
- Тестирование конфигураций.
- Нагрузочное тестирование.
- Канареичное тестирование.

# SRE принципы

## Релиз-инжиниринг

- Подготовка к релизу.
- Критерии успешности релиза.
- Управление релизом.
- Развёртывание.
- Проверка.

# SRE принципы

## Разработка

- 50% инженерной работы.
- Снижение риска выгорания.
- Карьерный рост.
- “Eating your own dog food.”
- “You build it, you run it.”

# SRE принципы

## Запуск новых продуктов

- Управление запусками.
- Настройка процесса запуска.
- Чек-лист для запуска.
- Планирование производительности.
- Устранение сбоев.

# Рекомендации SREless командам

- 12factor.app.
- «Не нужно оверинженириТЬ» / KISS.
- Мониторинг, логирование, профайлинг, трейсинг.
- Планирование ёмкостей.
- Лимиты, таймауты, ретраи, хелсчеки, рестарты.
- Документация.

# **И ещё несколько слов про SRE**

**Конец теоретической  
части...**