DevOps от А до Я

Никита Самохвалов,
заместитель тех. директора «Нотамедиа»

NOTAMEDIA

О наших проектах

— Долгосрочные проекты.

— Сервисно-ориентированная архитектура и «монолитные» сайты.

— Ежедневные релизы.

— Коммерческая и опенсорс-разработка.

Этот доклад для вас, если…

— Вы хотите быть эффективнее конкурентов.

— Вы хотите расти и развиваться.

— Каждая выкладка — длительный, сложный и нервный процесс.

— Переезд на новый хостинг и переустановка серверов для вас смертельный номер.

— А документация устаревает ещё не успев родиться.

Рабочий процесс курильщика

1. Разработчики написали новый код.

Рабочий процесс курильщика

1. Разработчики написали новый код.
2. Сисадмины недовольны: выбранное решение некорректно / нового железа нет / слишком поздно сообщили.

Рабочий процесс курильщика

1. Разработчики написали новый код.
2. Сисадмины недовольны: выбранное решение некорректно / нового железа нет / слишком поздно сообщили.
3. Сисадмины обновляют окружение, выкатывают на тест. Не работает.

Рабочий процесс курильщика

1. Разработчики написали новый код.
2. Сисадмины недовольны: выбранное решение некорректно / нового железа нет / слишком поздно сообщили.
3. Сисадмины обновляют окружение, выкатывают на тест. Не работает.
4. Сисадмины перекидываются с разработчиками обвинениями.

Рабочий процесс курильщика

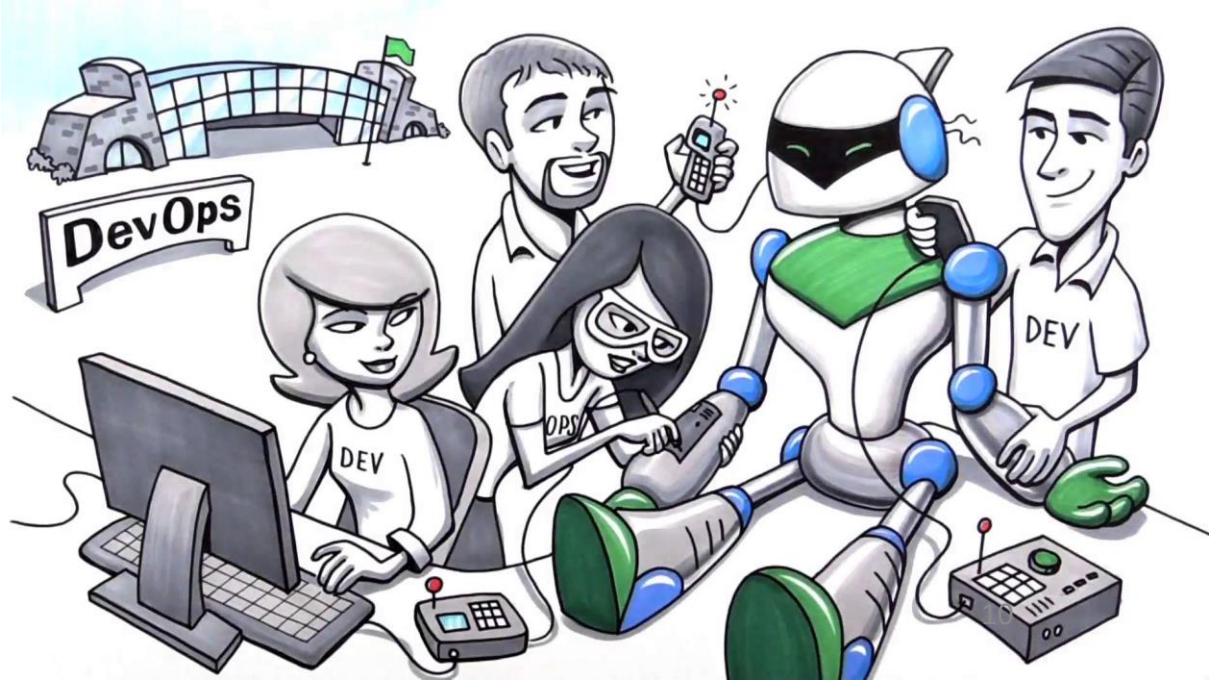
1. Разработчики написали новый код.
2. Сисадмины недовольны: выбранное решение некорректно / нового железа нет / слишком поздно сообщили.
3. Сисадмины обновляют окружение, выкатывают на тест. Не работает.
4. Сисадмины перекидываются с разработчиками обвинениями.
5. Заказчик обновляет окружение на проде и выкатывает продукт.

Рабочий процесс курильщика

1. Разработчики написали новый код.
2. Сисадмины недовольны: выбранное решение некорректно / нового железа нет / слишком поздно сообщили.
3. Сисадмины обновляют окружение, выкатывают на тест. Не работает.
4. Сисадмины перекидываются с разработчиками обвинениями.
5. Заказчик обновляет окружение на проде и выкатывает продукт.
6. На проде невоспроизводимые баги (окружение отличается, что-то не так обновили). Заказчик недоволен. Всё напрасно. Безысходность.

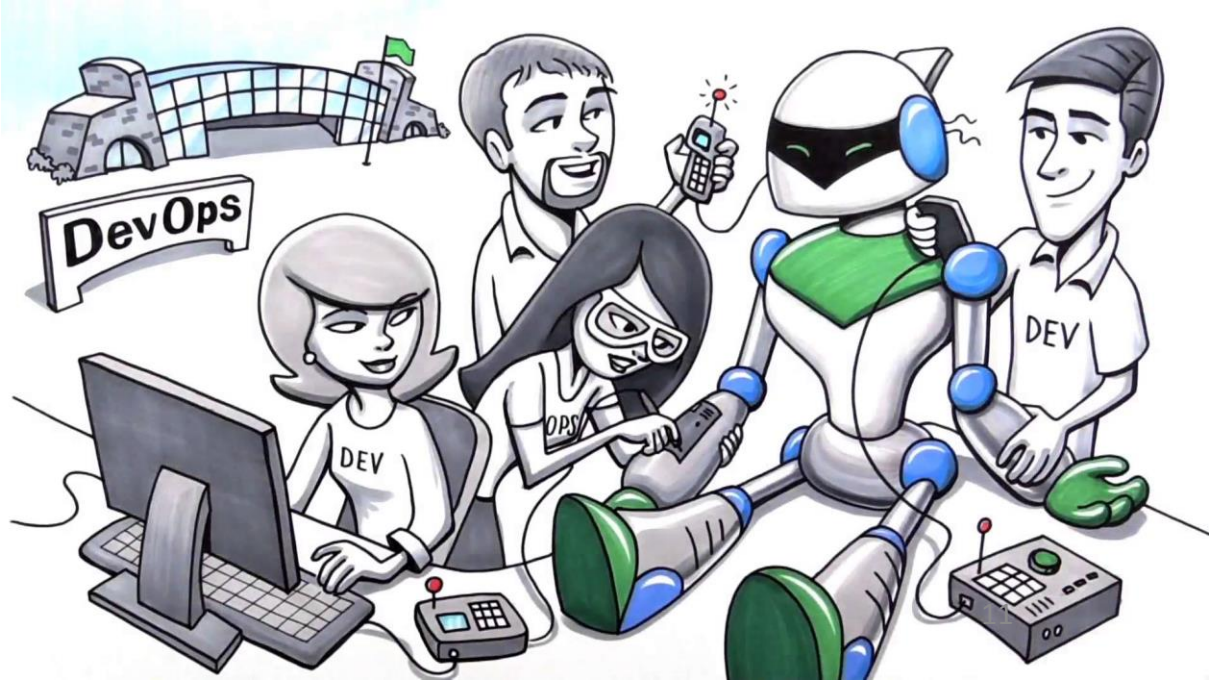
Рабочий процесс девопса

1. Разработчики и сисадмины согласовывают друг с другом техническое решение.



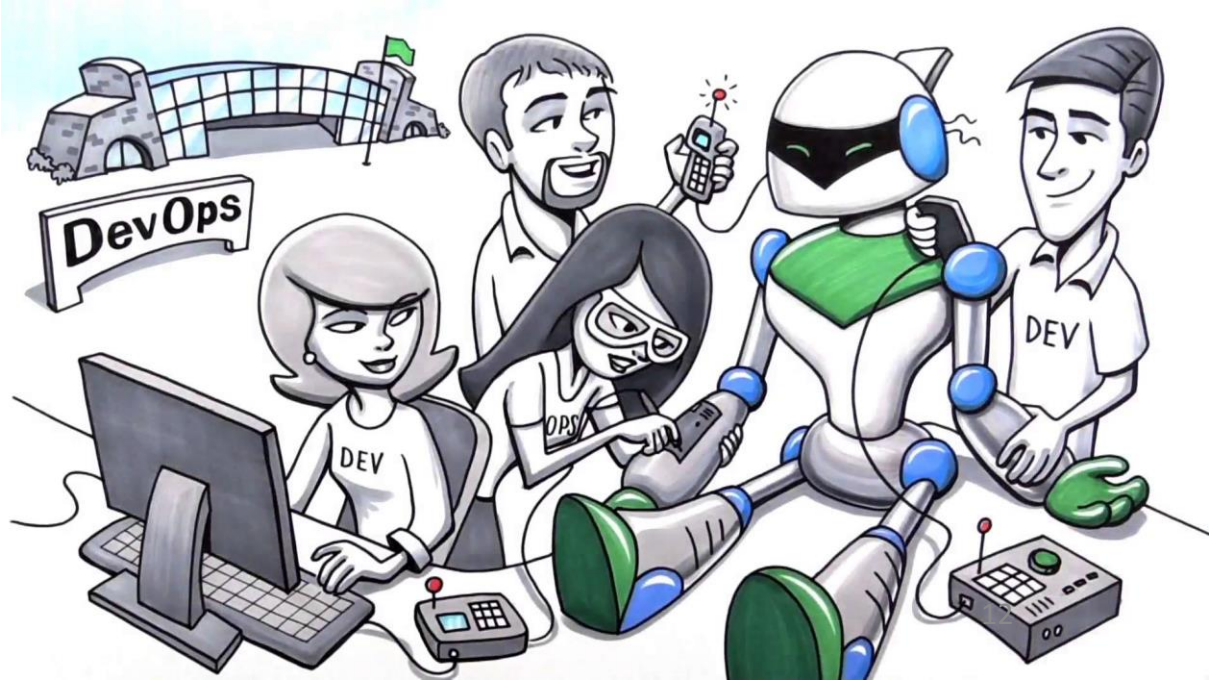
Рабочий процесс девопса

1. Разработчики и сисадмины согласовывают друг с другом техническое решение.
2. Разработчики и сисадмины написали новый код: бизнес-логика продукта, серверные настройки, ПО и т. д.



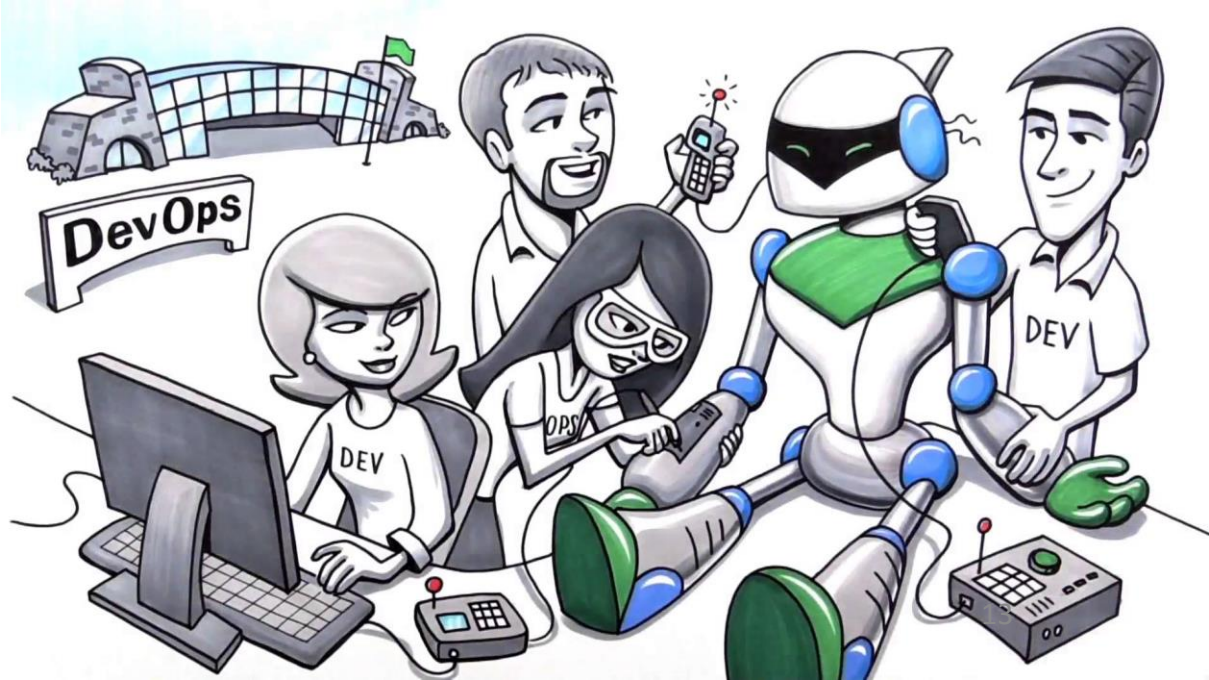
Рабочий процесс девопса

1. Разработчики и сисадмины согласовывают друг с другом техническое решение.
2. Разработчики и сисадмины написали новый код: бизнес-логика продукта, серверные настройки, ПО и т. д.
3. Выкатили на тестирование (никаких ручных операций).



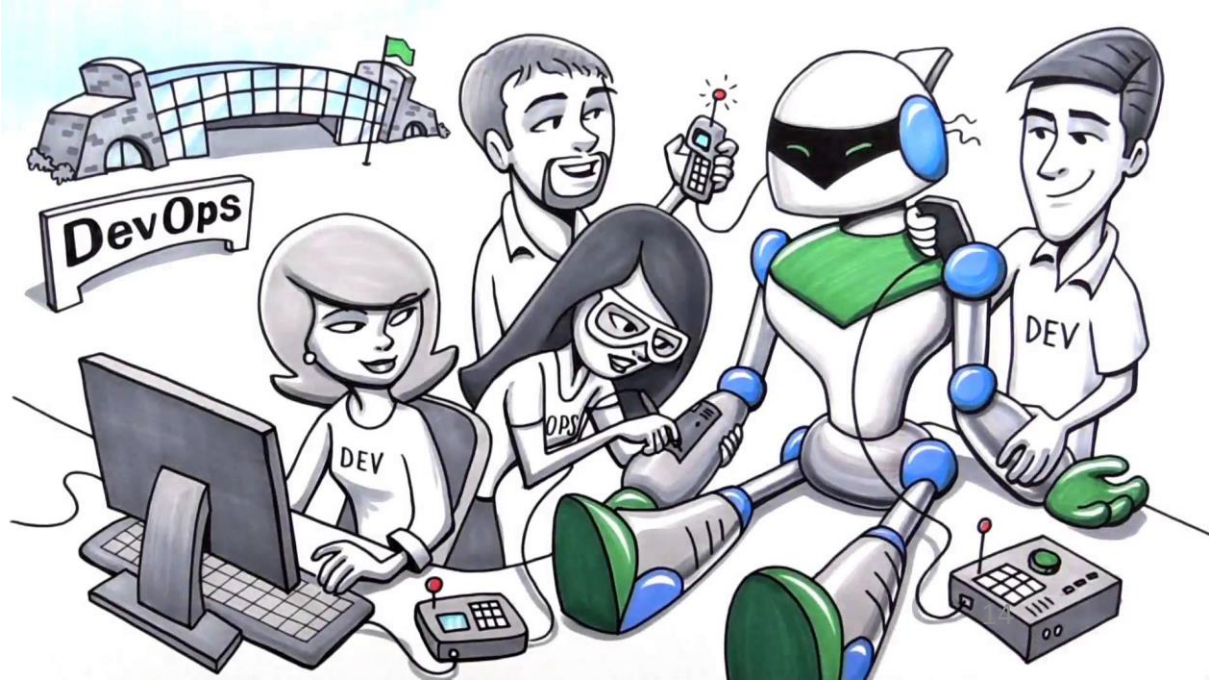
Рабочий процесс девопса

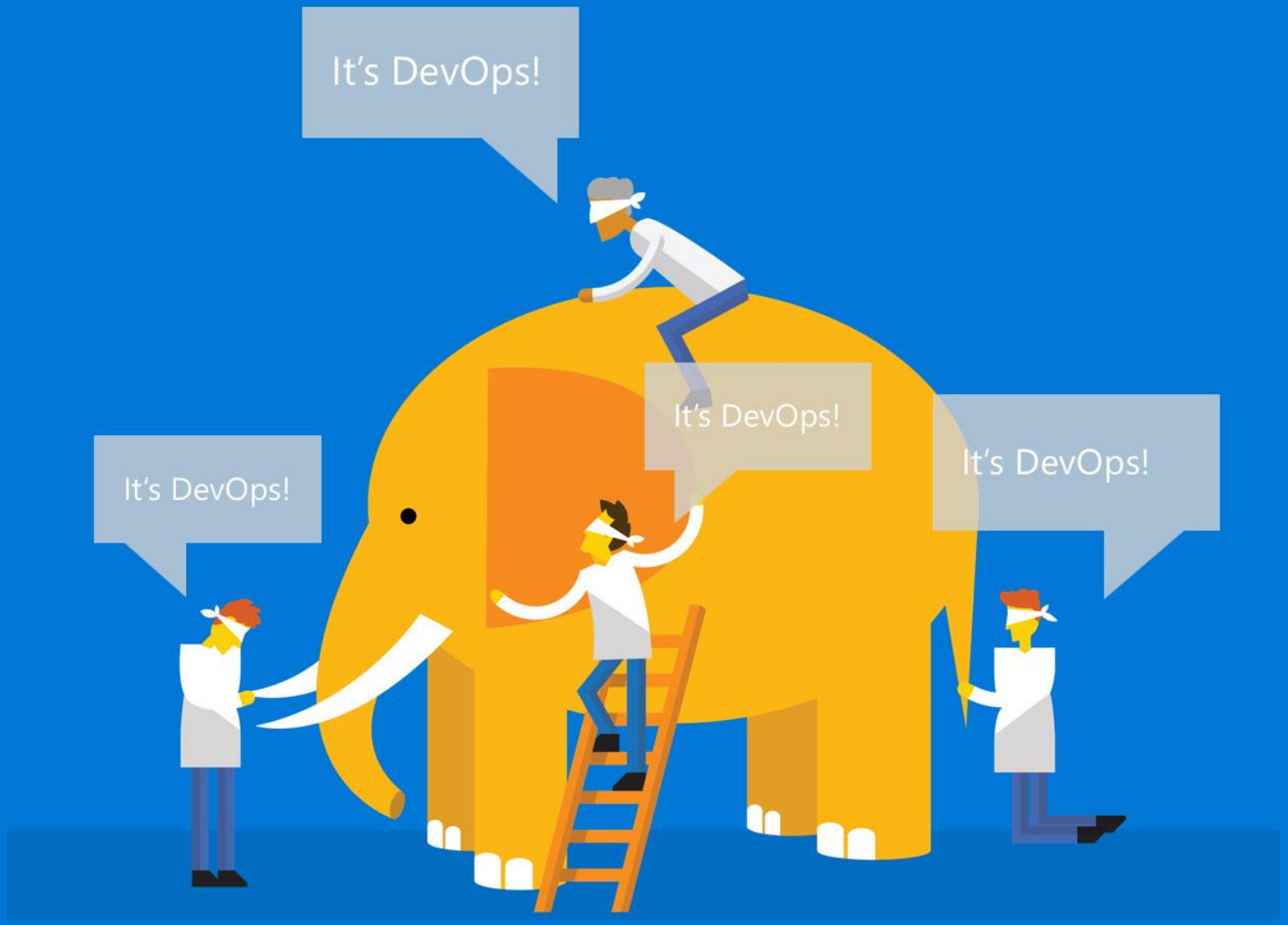
1. Разработчики и сисадмины согласовывают друг с другом техническое решение.
2. Разработчики и сисадмины написали новый код: бизнес-логика продукта, серверные настройки, ПО и т. д.
3. Выкатили на тестирование (никаких ручных операций).
4. Выкатили на прод (всё автоматизировано).



Рабочий процесс девопса

1. Разработчики и сисадмины согласовывают друг с другом техническое решение.
2. Разработчики и сисадмины написали новый код: бизнес-логика продукта, серверные настройки, ПО и т. д.
3. Выкатили на тестирование (никаких ручных операций).
4. Выкатили на прод (всё автоматизировано).
5. Мониторим продукт.



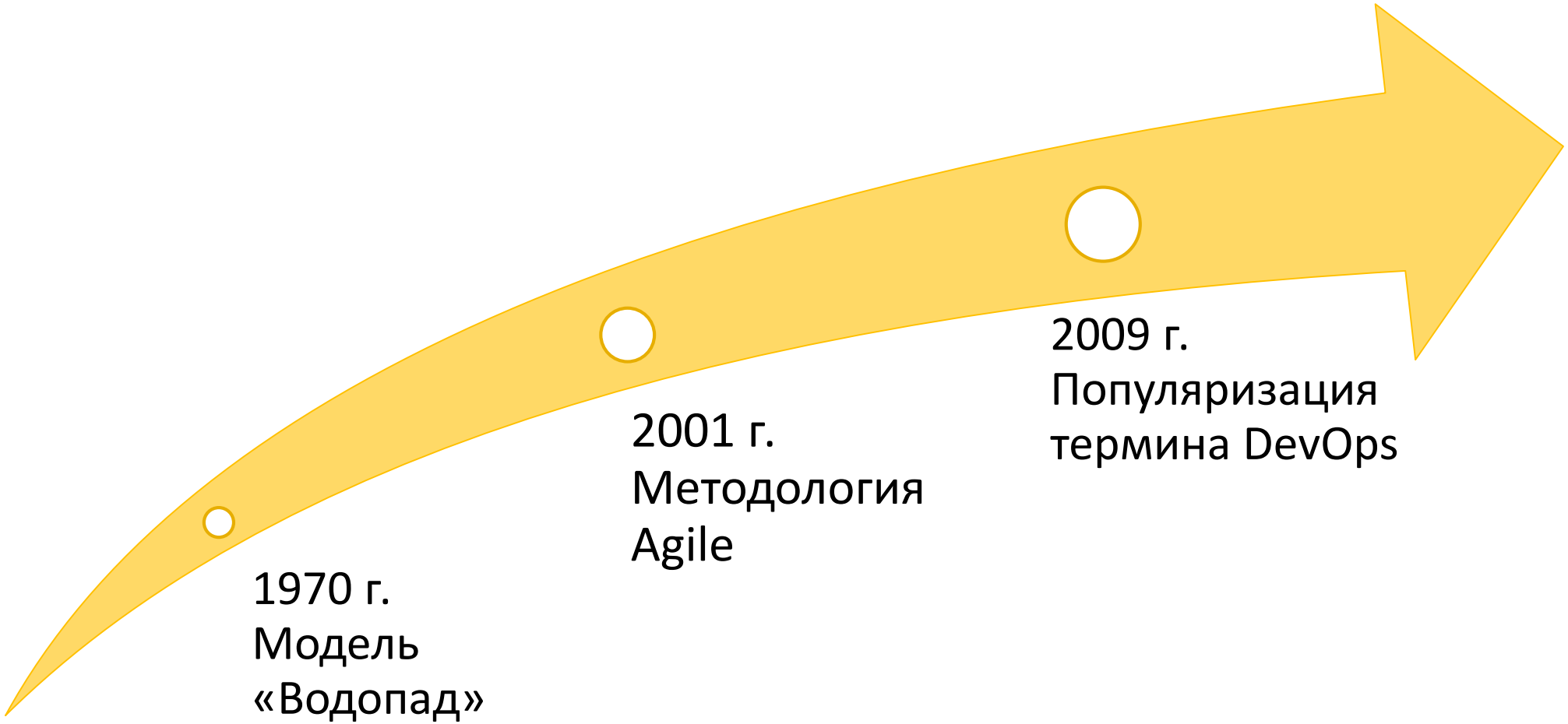

It's DevOps!
It's DevOps!
It's DevOps!
It's DevOps!

DevOps

Это набор практик, нацеленных на активное взаимодействие и интеграцию специалистов по разработке и эксплуатации. Нацелена на то, чтобы помогать организациям быстрее создавать и обновлять программные продукты.

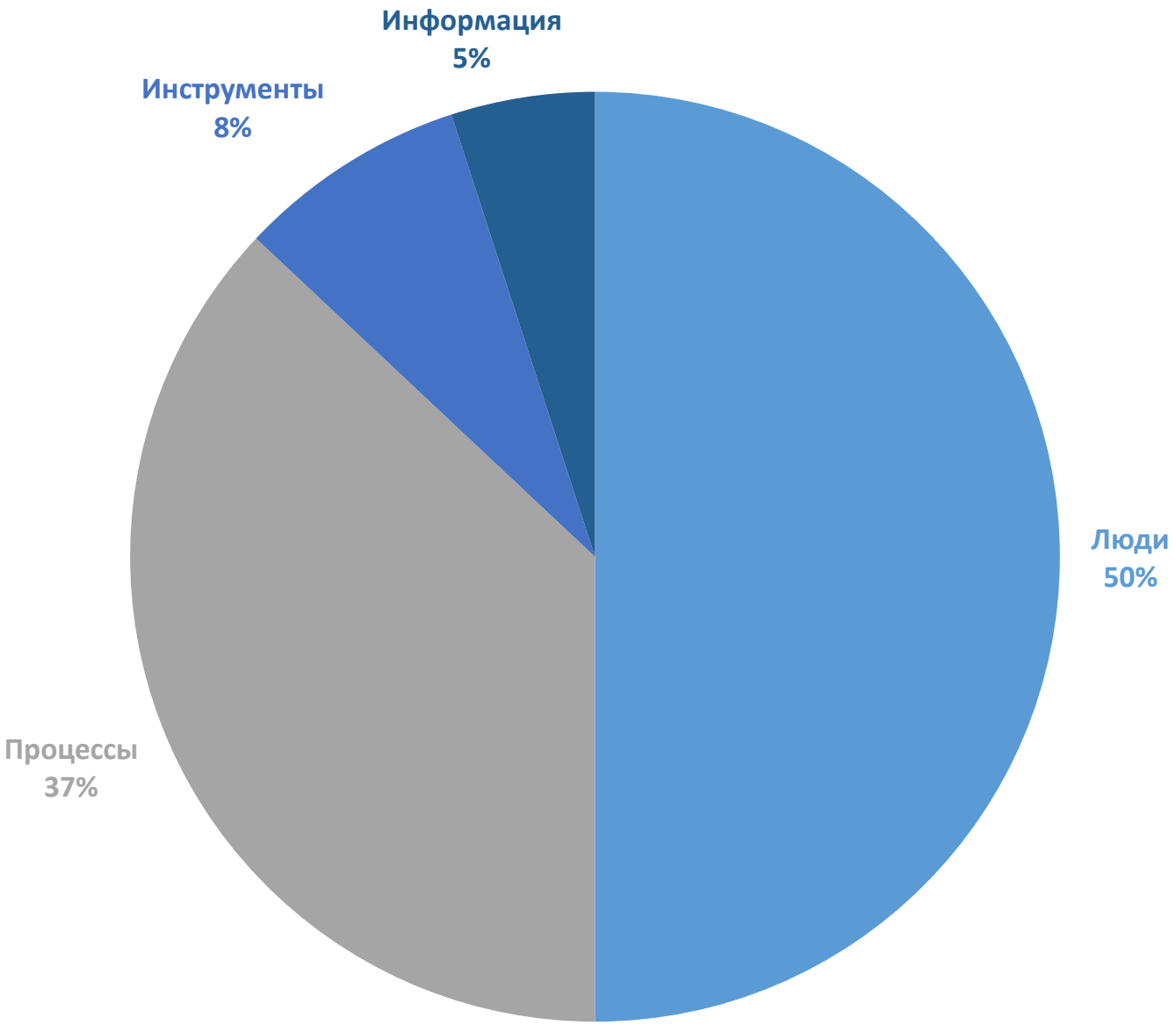
Манифест: https://goo.gl/bSe1NR

Немного истории



Рецепт успеха

1. Сплочённая команда.
2. Правильно выстроенные процессы разработки и эксплуатации.
3. Автоматизация рутины, хорошие инструменты.



Источник: gartner.com/smarterwithgartner/the-science-of-devops-decoded

Люди и процессы

Хорошая, сплочённая команда и правильно построенные процессы производства и эксплуатации — большая часть успеха.

Командная работа

— Общие каналы связи.

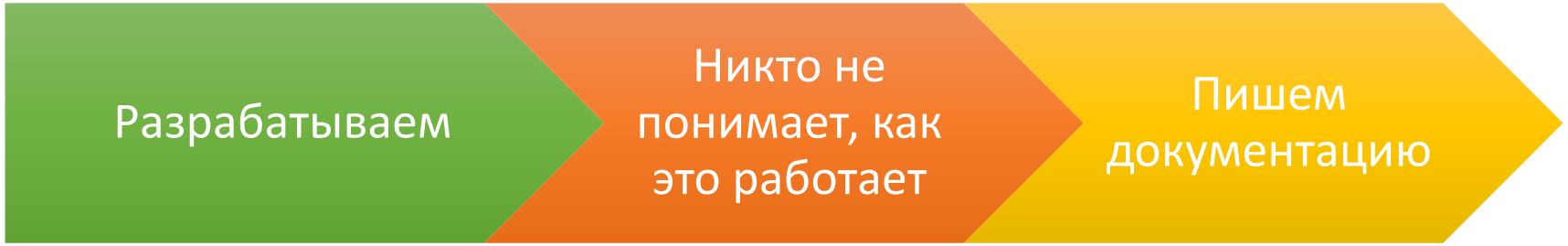
— Обмен знаниями и опытом.

— Совещания специалистов разных отделов: от дизайна до эксплуатации.

— Согласование технических решений.

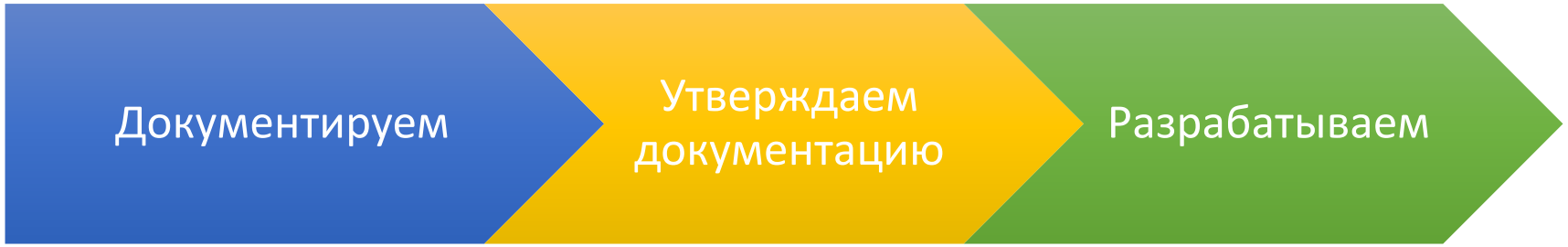
— Совместное составление тест-кейсов, проведение испытаний.

Документация



Документация

Документация



Документация

Что?

— Технические решения, инструкции по эксплуатации.

— API сервисов.

— Стандарты, регламенты, правила, лучшие практики.

Документация

Что?

— Технические решения, инструкции по эксплуатации.

— API сервисов.

— Стандарты, регламенты, правила, лучшие практики.

Зачем?

— Взаимозаменяемость сотрудников.

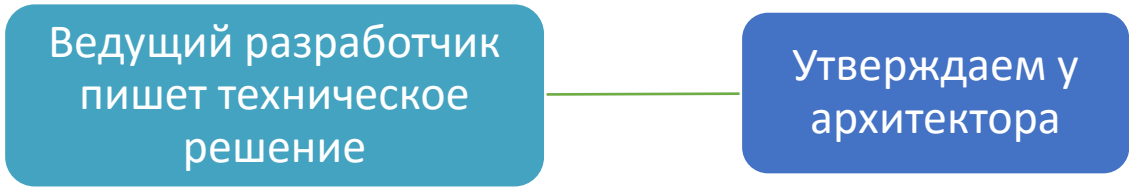
— Передача знаний.

— Исключение архитектурных ошибок.

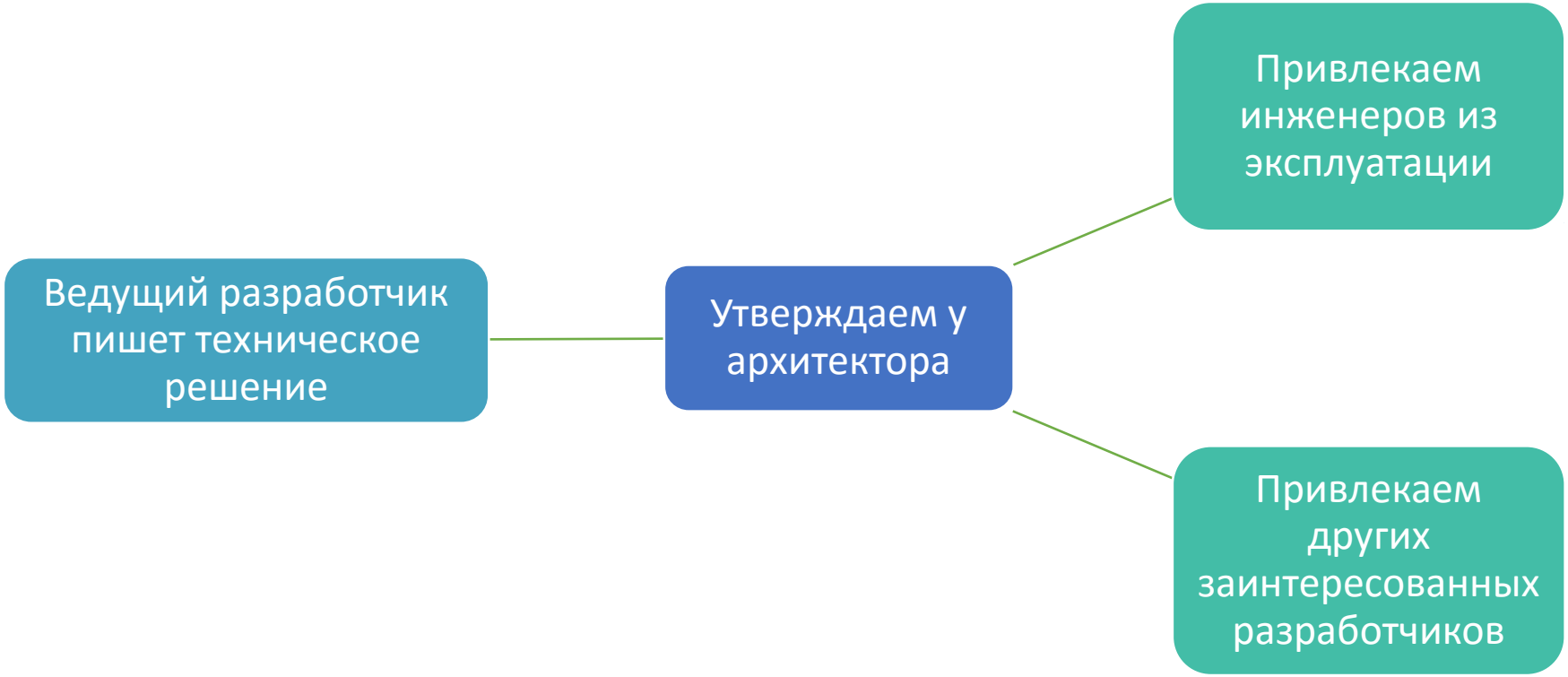
Обсуждение и согласование

Ведущий разработчик пишет техническое решение

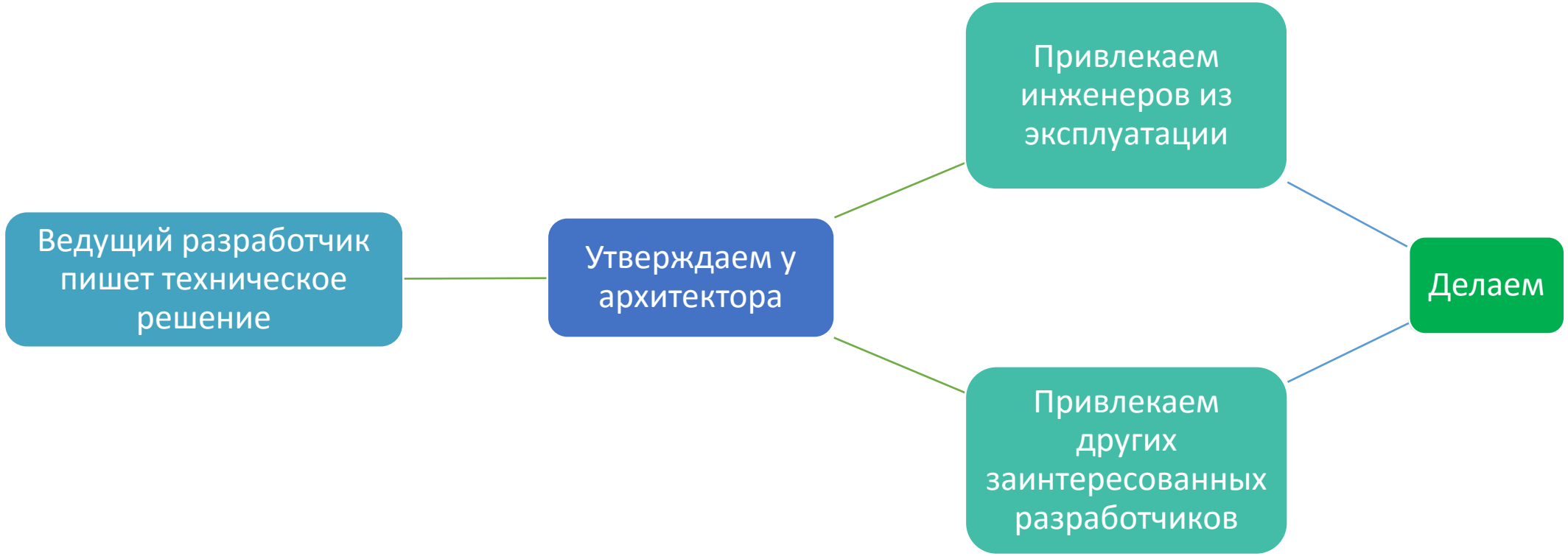
Обсуждение и согласование



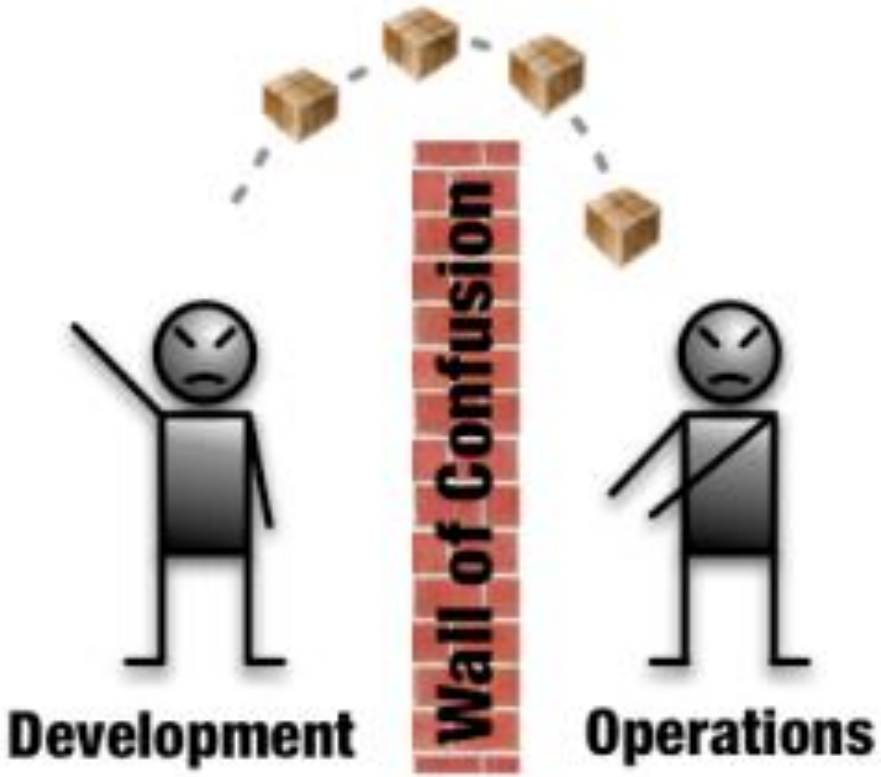
Обсуждение и согласование



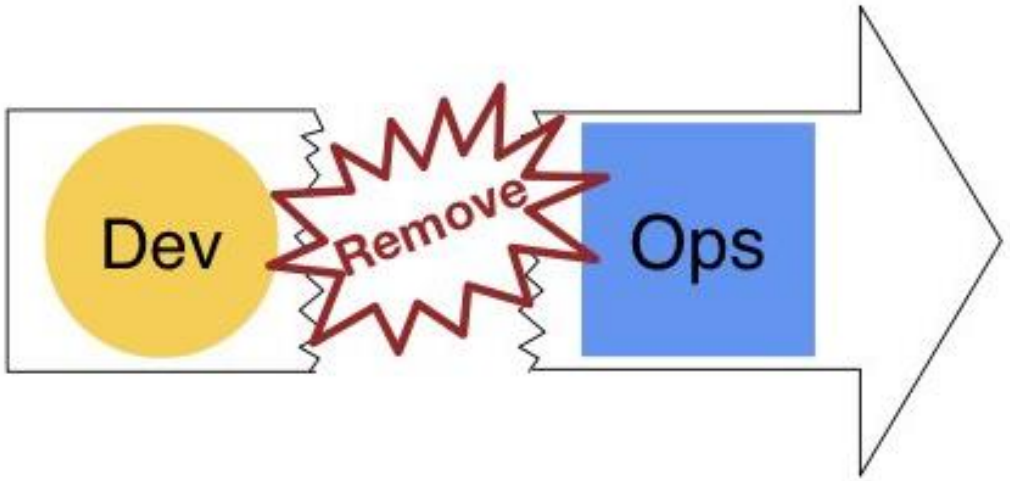
Обсуждение и согласование



Эксплуатация



Эксплуатация



Эксплуатация

Дежурный релиз-инженер
Проверка кода на уязвимости,
планирование и осуществление выкладок.

Эксплуатация

Дежурный релиз-инженер
Проверка кода на уязвимости,
планирование и осуществление выкладок.

Дежурный инженер ТП
Диагностика багов.

Эксплуатация

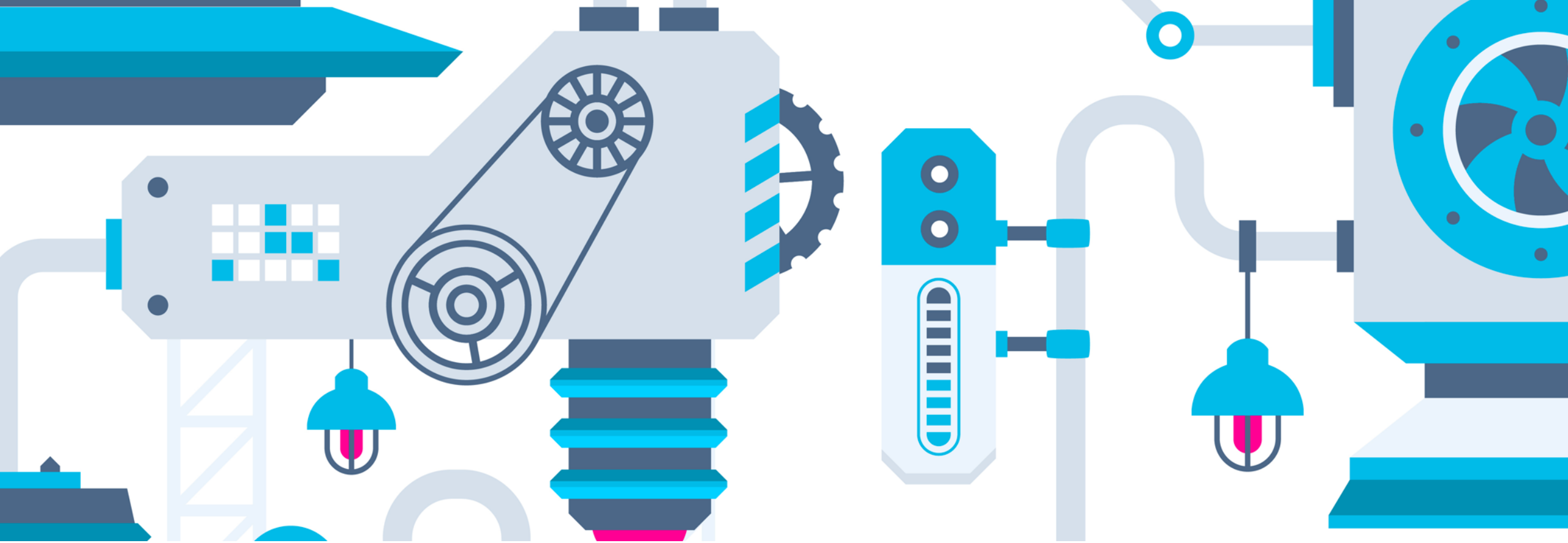
Дежурный релиз-инженер
Проверка кода на уязвимости,
планирование и осуществление выкладок.

Дежурный инженер ТП
Диагностика багов.

Владеют устройством инструментов сборки и деплоя.

Знают все компоненты облака, их зависимости.

Имеют полный доступ ко всему облаку.



Технологии и инструменты

Люди должны проектировать и кодить,
а рутину выполнять машины.

Инфраструктура как код (IaC)

IaC — это когда продукт может быть полностью восстановлен из хранилища исходного кода и бекапов данных.

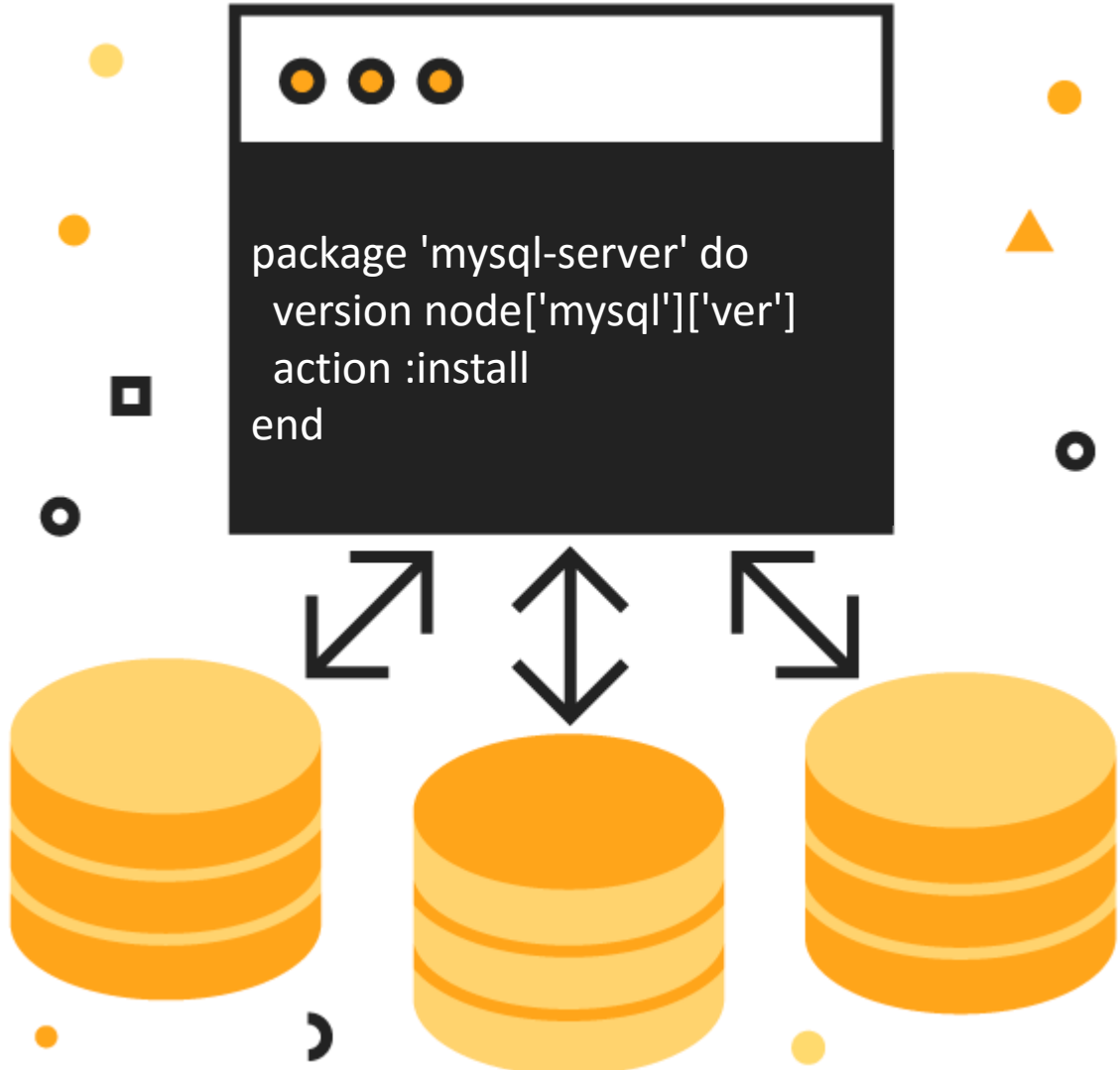
— Инфраструктура должна быть модульной, легко поддающейся автоматизации.

— Описывается инфраструктура с помощью языка высокого уровня.

directory '/var/lib/foo' do
owner 'root'
group 'root'
mode '644'
action :create
end

cron 'name_of_cron_entry' do
minute '0'
hour '8'
weekday '6‘
action :create
end

apt_update 'all platforms' do
frequency 86400
action :periodic
end


package 'mysql-server' do
version node['mysql']['ver']
action :install
end

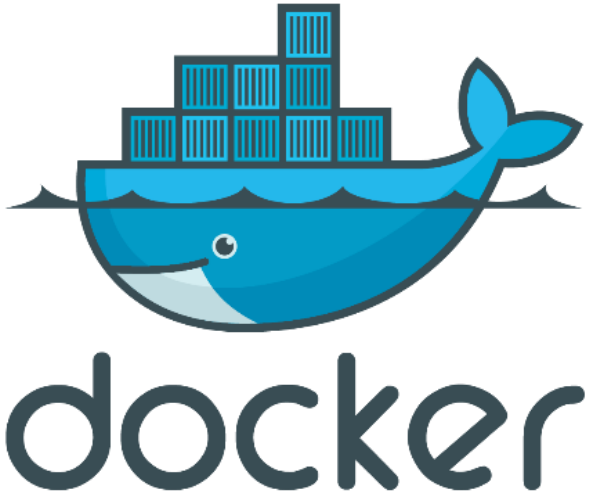
Управление настройками серверов

chef.io

saltstack.com

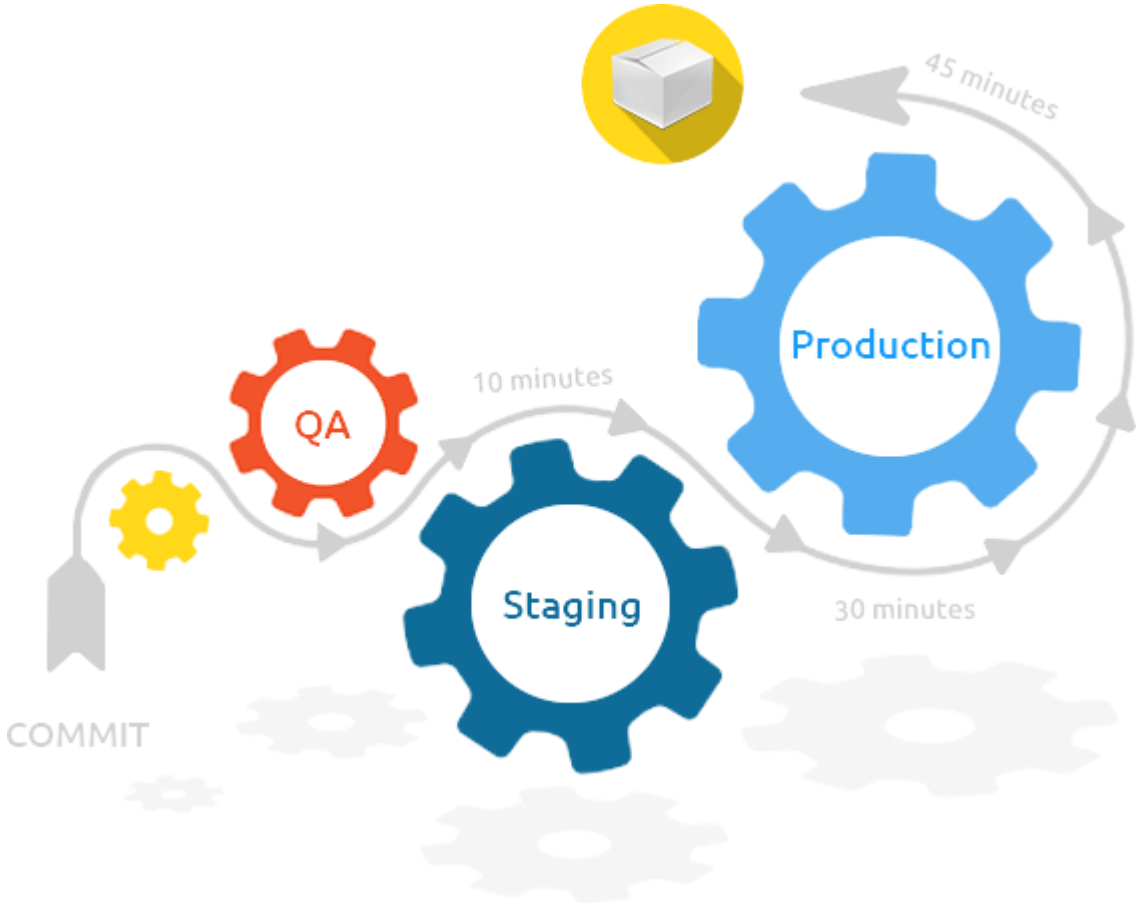
Виртуализация и контейнеризация

vagrantup.com

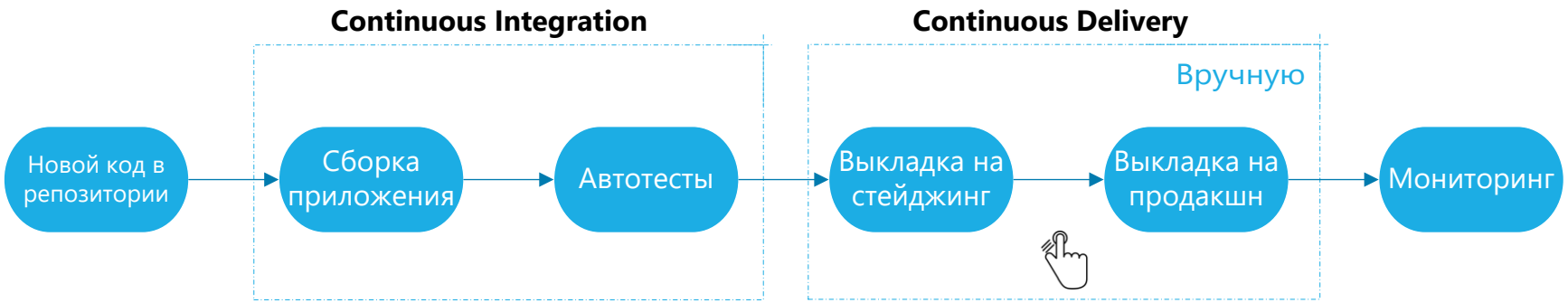


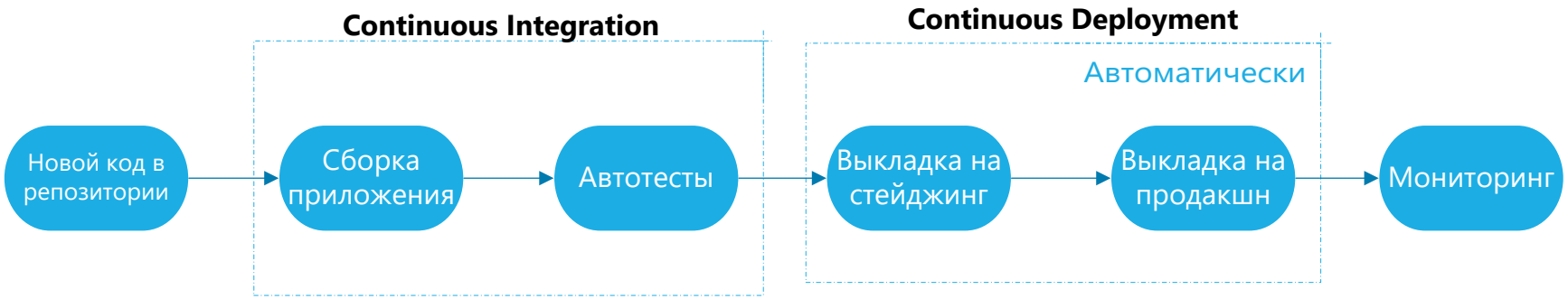
docker.com

Непрерывная интеграция и поставка



Непрерывная интеграция и поставка





Непрерывная интеграция и поставка

Существующие решения:

— Jenkins

— Travis CI

— Gitlab CI/CD

— TeamCity

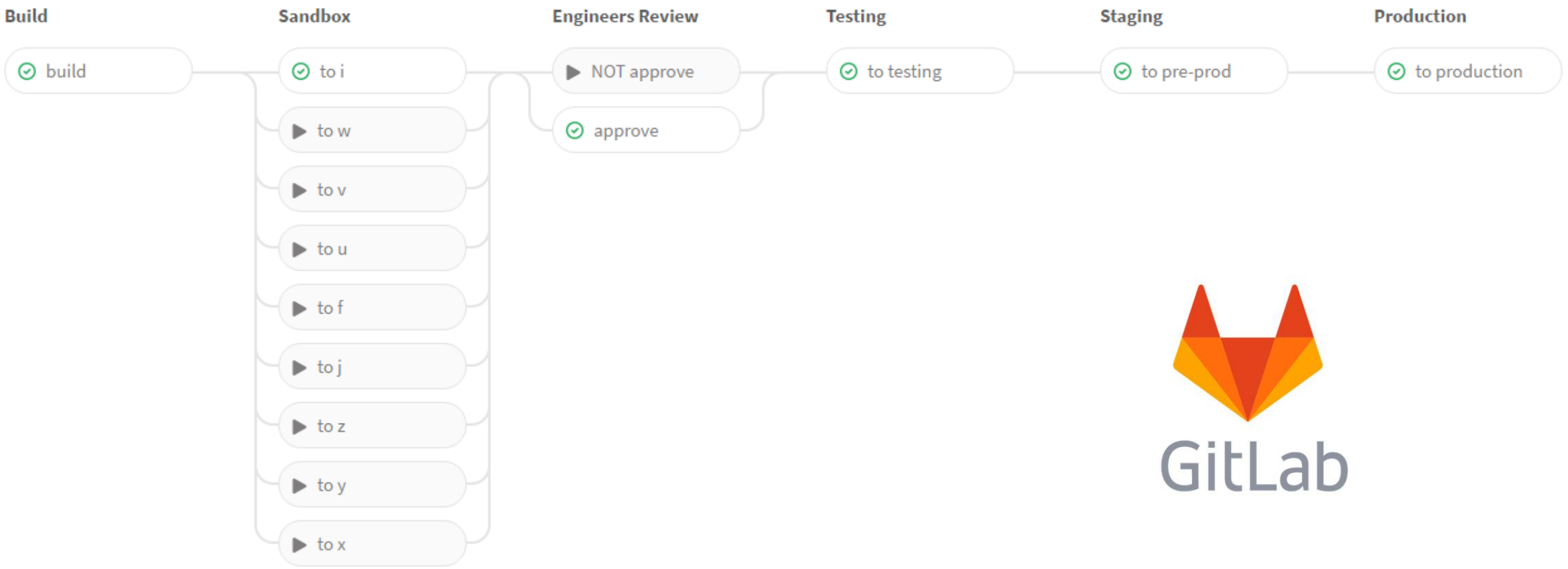
— Bamboo

— Codeship

— CircleCI

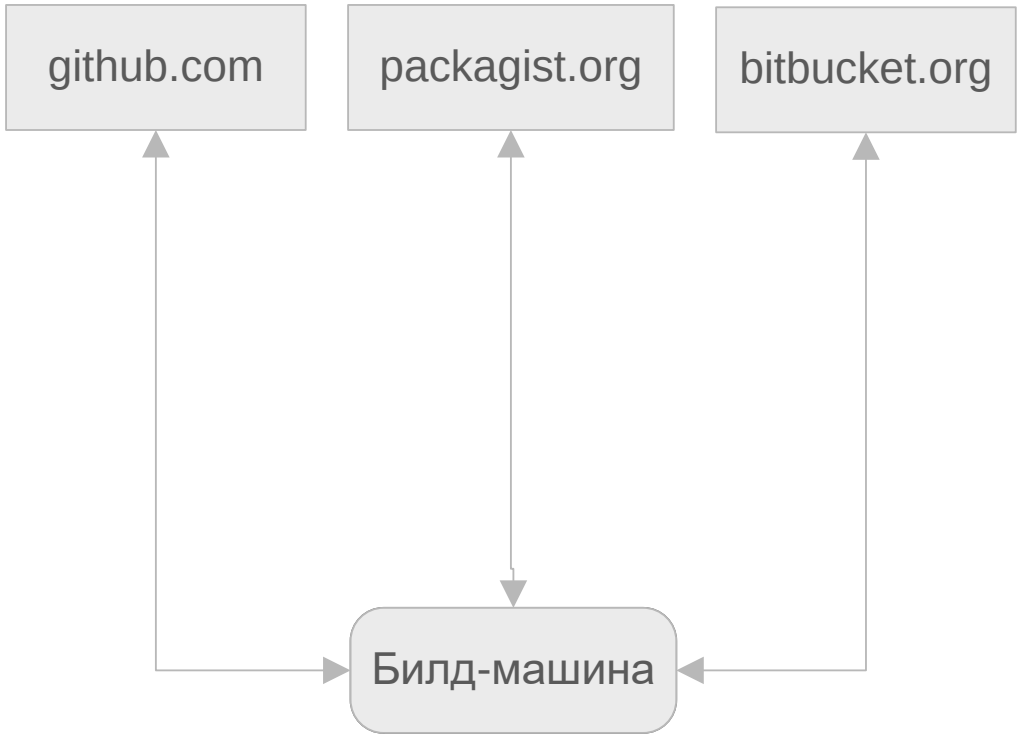
и др.

Непрерывная интеграция и поставка

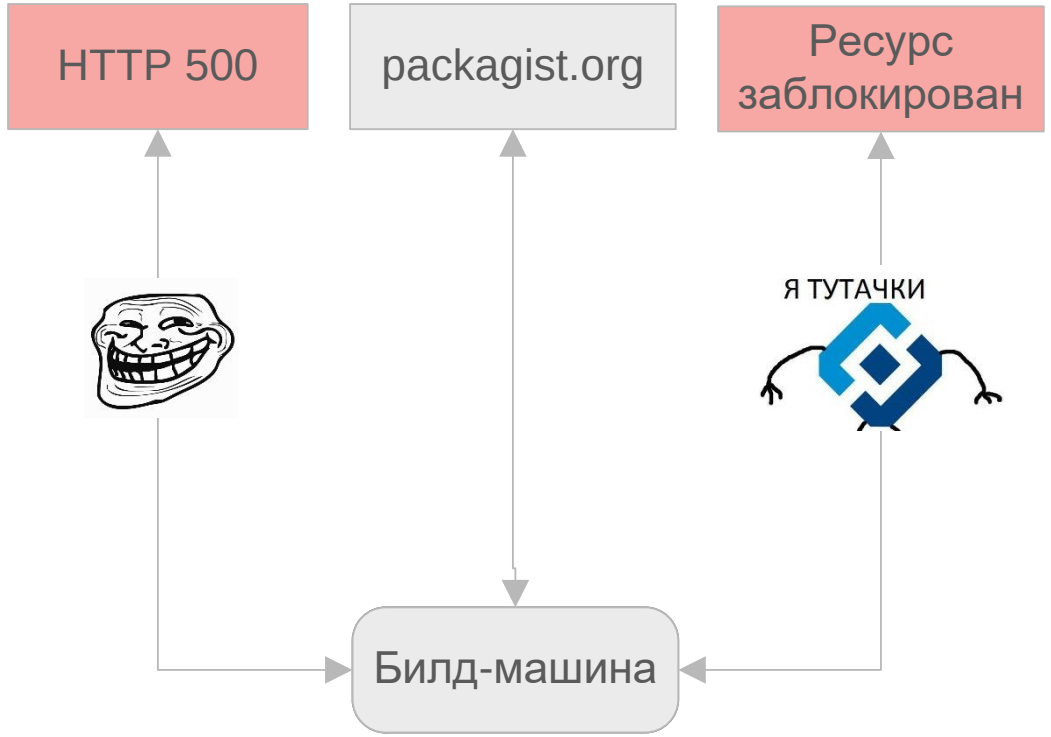


GitLab

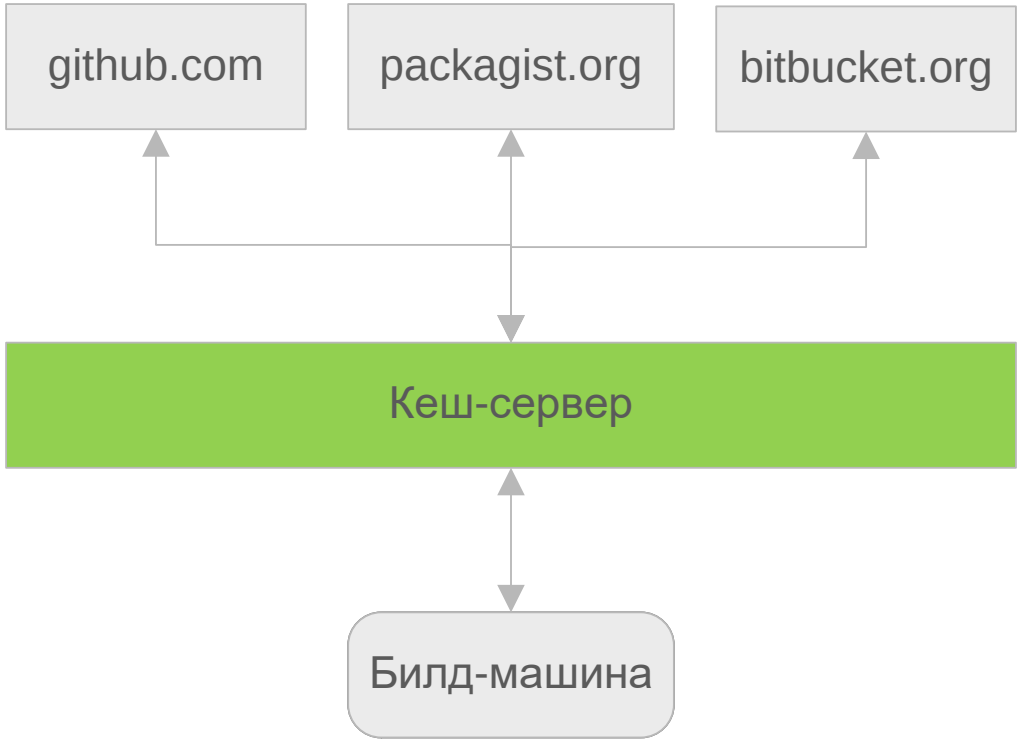
Резервирование зависимостей



Резервирование зависимостей



Резервирование зависимостей



Резервирование зависимостей

Существующие решения:

— Satis github.com/composer/satis

— Toran Proxy toranproxy.com

— Nexus sonatype.org/nexus

— Sinopia npmjs.com/package/sinopia

и др.

Мониторинг

Следим за продуктом, своевременно узнаём о сбоях.

«Что сейчас происходит в системе»?

«Кто изменял эту запись?»

«Когда и кем был загружен файл?»

«Что делал этот посетитель?»

«Почему не работает?»

Мониторинг

Существующие решения

Мониторинг:

— Zabbix

— Grafana

— New Relic

и др.

Логирование:

— Kibana (+ logstash + elasticsearch)

— Graylog (+ logstash + elasticsearch)

— Sentry

и др.

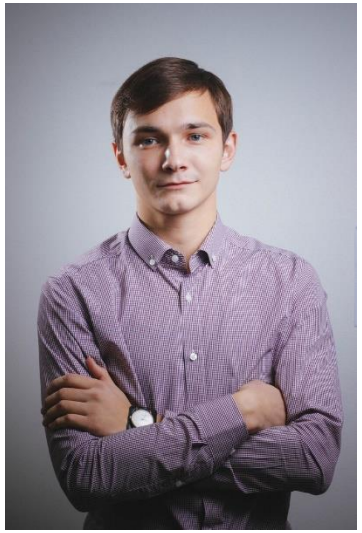
Не останавливайтесь на достигнутом

1. Сделать инфраструктуру как код (IaC).
2. Внедрить непрерывную интеграцию и доставку (CI, CD).
3. Зарезервировать зависимости.
4. Покрыть системы мониторингом.
5. …

Выводы

1. Полноценный DevOps — это дорого и не для всех проектов.
2. В первую очередь люди и процессы, уже потом — инструменты.
3. Автоматизация экономит время для разработки и эксплуатации.
4. Заказчик получает новый продукт быстрее и стабильнее.

Спасибо за внимание



Никита Самохвалов,
заместитель тех. директора «Нотамедиа»

nik@samokhvalov.info
github.com/niksamokhvalov