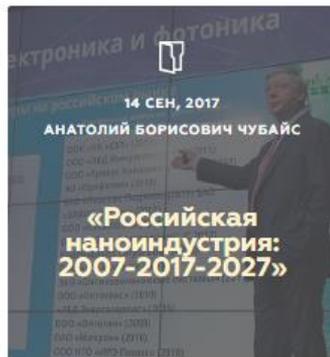
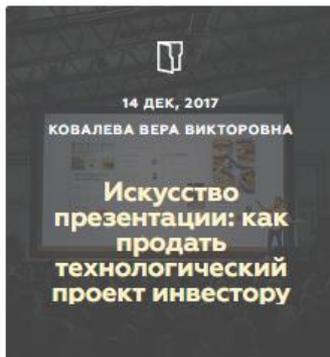
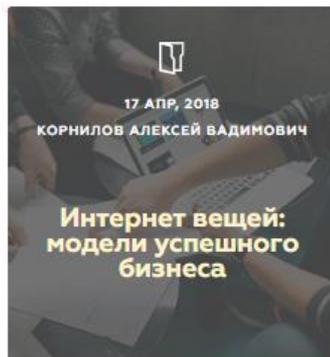
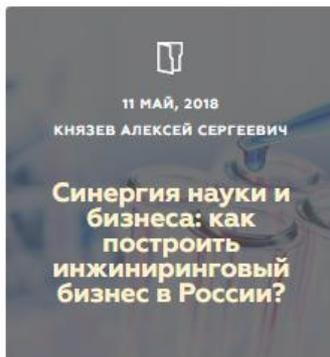
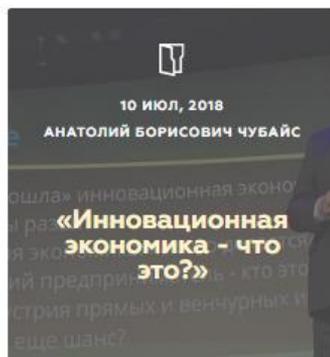
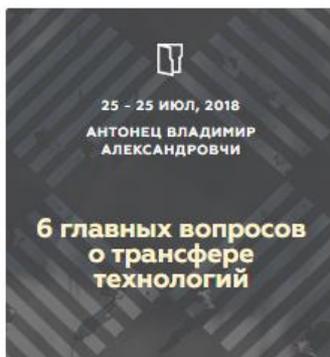


**лекторий**  
#техпред

«7» ноября 2018 года

**Вебинар**  
«Практическое применение DevOps  
в технологических стартапах  
и не только»



**Лекторий #техпред** - серия публичных лекций, организованных совместно Фондом инфраструктурных и образовательных программ РОСНАНО и Межвузовской программой подготовки инженеров в сфере высоких технологий.

Проект направлен на формирование открытого профессионального сообщества ученых, инноваторов и технопредпринимателей, деятельность которых ориентирована на развитие сферы высоких технологий.

# ПРАВИЛА ВЕБИНАРА

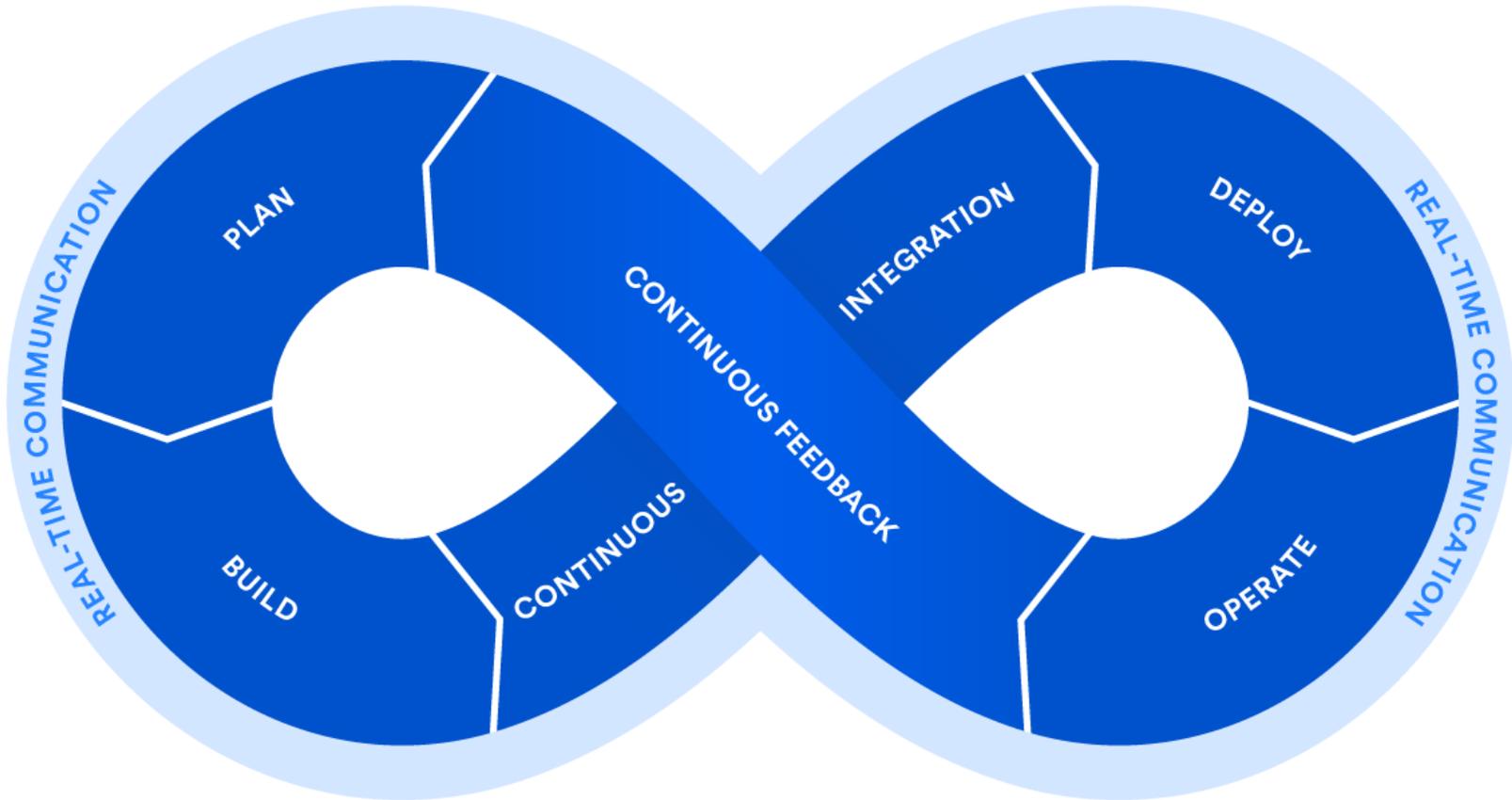
- Вопросы докладчику можно задавать по ходу вебинара во вкладке ?
- Технические и организационные вопросы можно задавать во вкладке **ЧАТ**
- По итогам вебинара все зарегистрированные участники получают ссылку на видеозапись и презентацию

# ГРАХОВАЦ ЕЛЕНА

Lead TechOps Automation Engineer  
N26 (Берлин, Германия)



# Что такое DevOps?



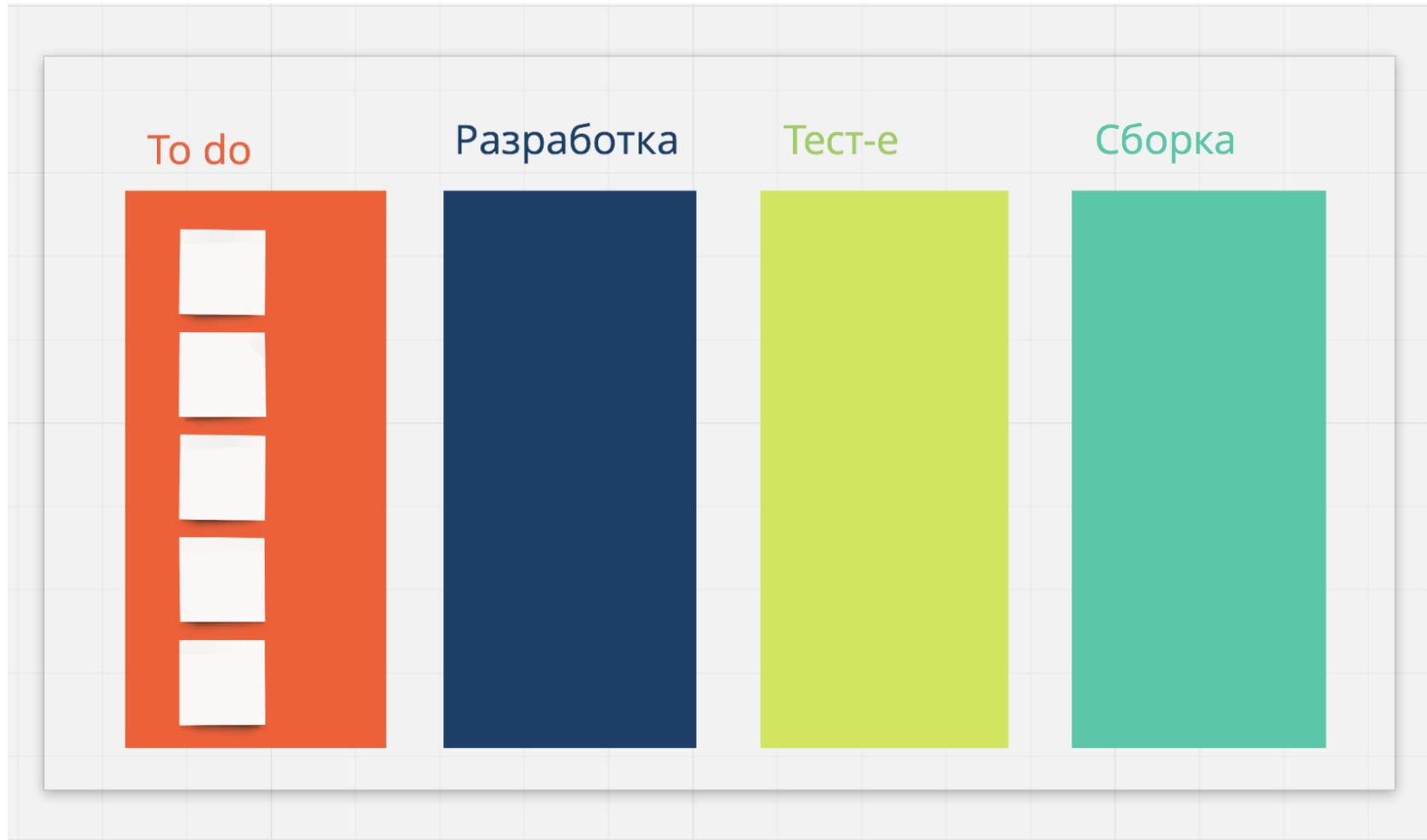
# Ключевые ценности

- Системный подход
- Обратная связь
- Обмен знаниями

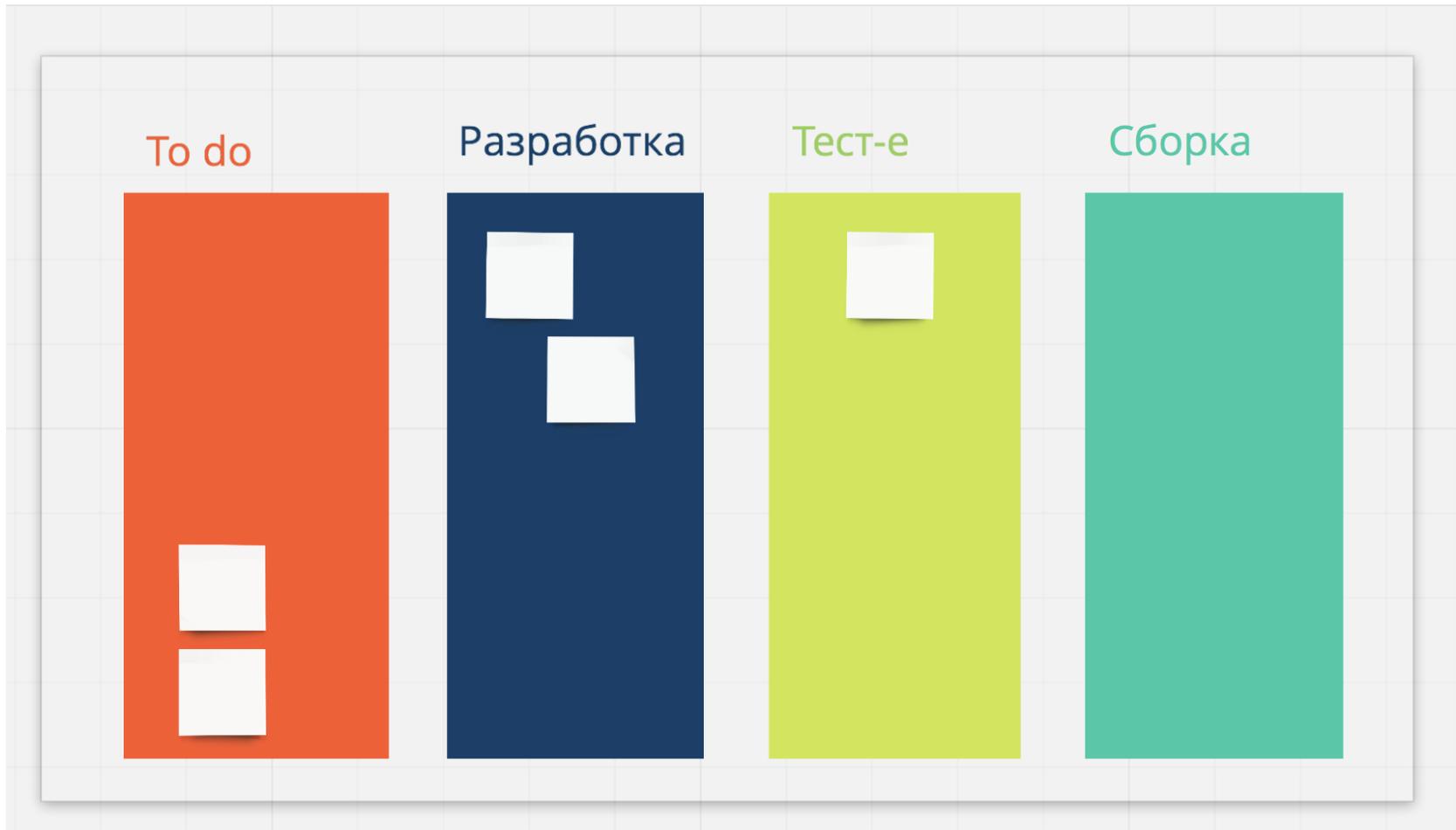
# Проблемы итеративной методологии разработки.

## История возникновения DevOps

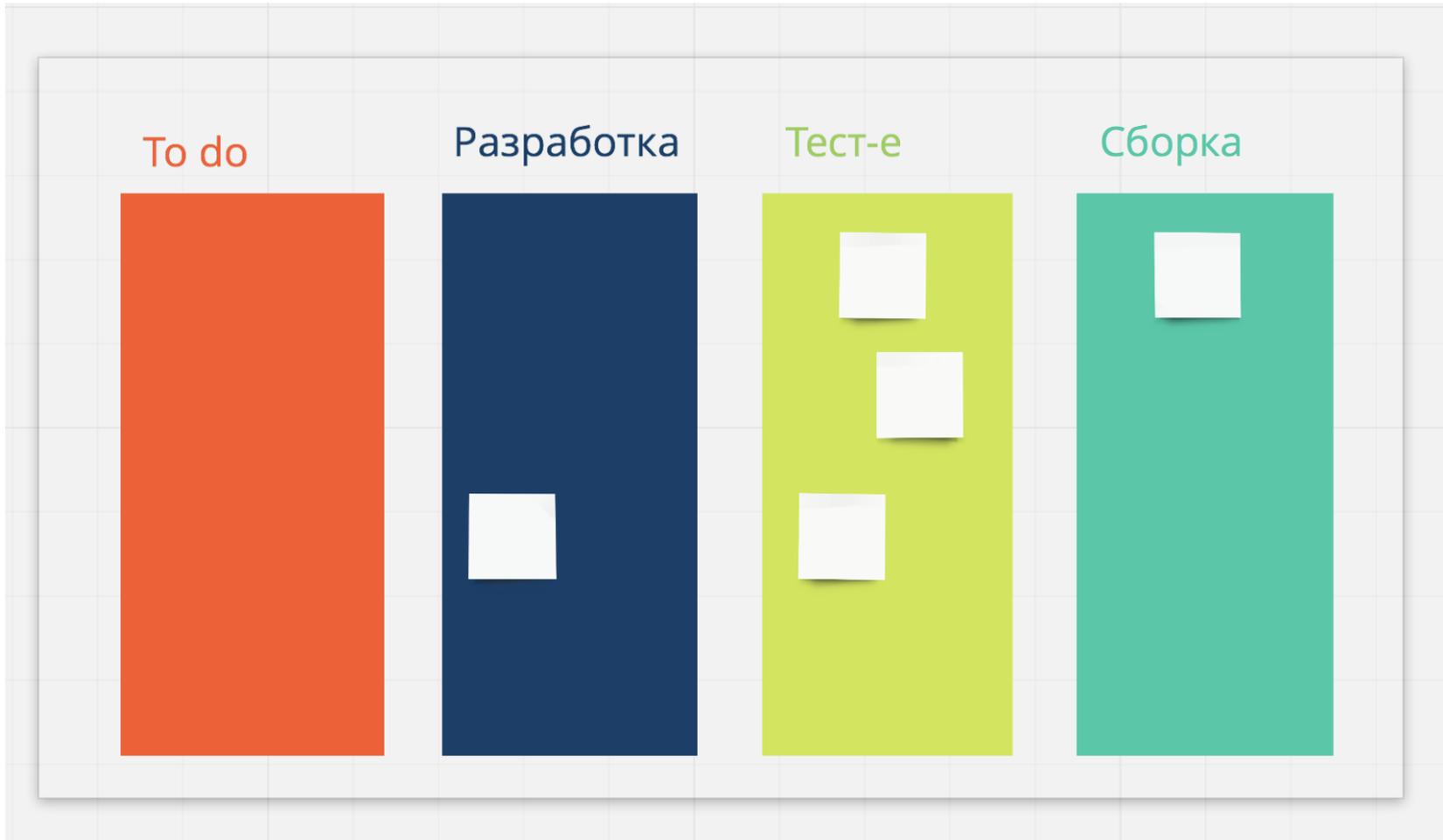
# Scrum. Начало итерации



# Scrum. Середина итерации



# Scrum. Ближе к концу итерации



## Scrum. Потенциальные проблемы

- Задачи от разработчиков пришли в конце итерации
- При регрессионном тестировании сборки нашли слишком много багов
- Сборка готова, но нет инструкции для релиза
- Забыли про отдел эксплуатации

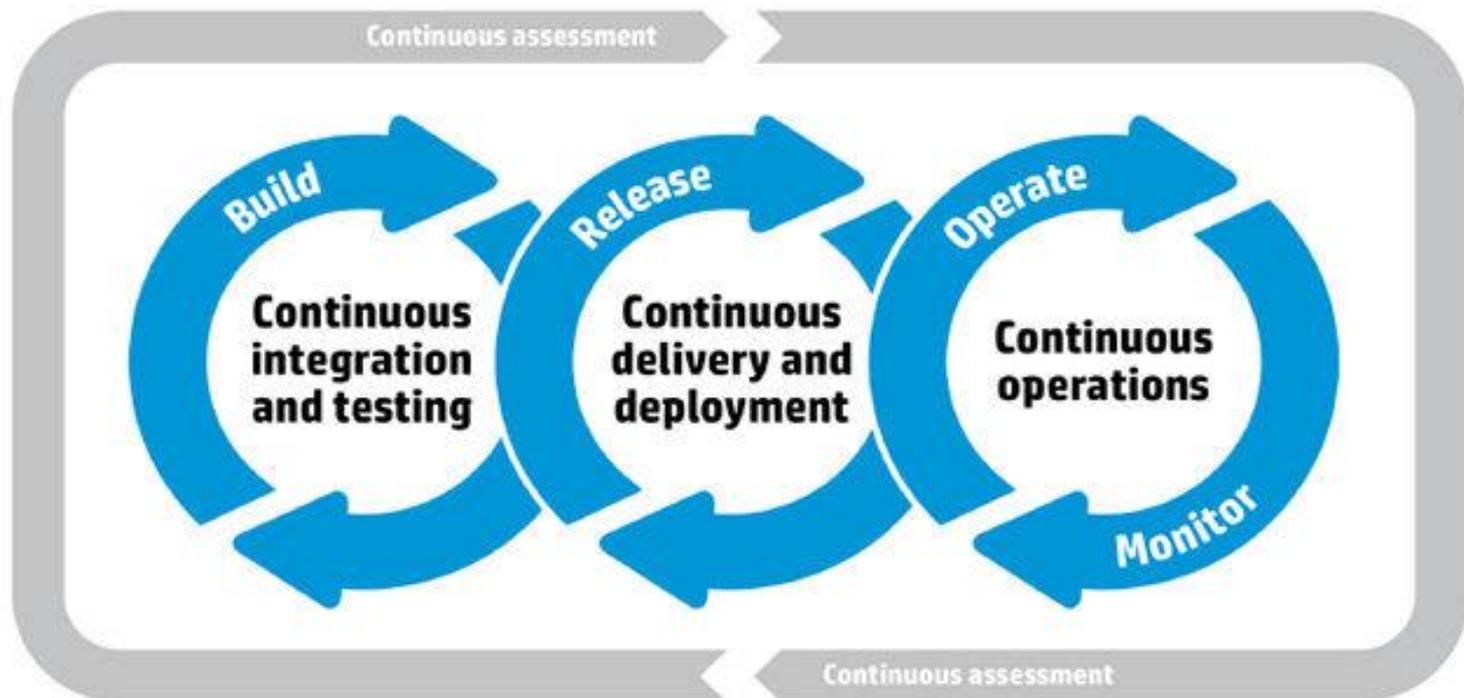
# От итераций к непрерывности

- Регрессионное тестирование каждой задачи
- Сборка каждой задачи
- Релиз каждой задачи

# От итераций к непрерывности

- Continuous Integration
- Continuous Deployment

# От итераций к непрерывности



# Автоматизация

- Проверка качества кода
- Сборка
- Мониторинг

# Ценность для стартапа

- Гарантия выхода задачи
- Минимизация времени прохождения задачи от разработки к релизу
- Минимизация отладки проблем, обнаруженных после релиза

# Современное состояние DevOps

# DevOps как культура

- DevOps – это не роль!
- DevOps – это не отдел!
- DevOps – это не набор инструментов!

# DevOps – не односторонняя дорога!

# DevOps - это команда

- Инженер разработки
- Инженер контроля качества
- Инженер безопасности
- Инженер инфраструктуры

# Развитие принципов разработки ПО

- Частые релизы
- Разделение кода на независимые компоненты
- Микросервисы
- Контейнеризация

# Cloud Native

- Управляемая инфраструктура
- Коммуникации через сеть
- Контейнеры
- Функция как сервис - serverless

# Требования к сервисам

- Отказоустойчивость (Fault Tolerance)
- Надежность (Reliability)
- Пригодность к обслуживанию (Maintainability)
- Операбельность (Operability)
- Наблюдаемость (Observability)

# Требования к сервисам

- Безопасность (Security)
- Автоматизация (Automation)
- Простота (Simplicity)

## Инструменты и практики

- Непрерывные интеграция и развертывание (Continuous Integration and Deployment)
- Визуализация и ChatOps
- Инфраструктура как код (IaC)
- Мониторинг, логирование и алертинг

## Инструменты и практики

- 12-факторные (12 factor) приложения
- Дежурства (On-Call)
- Управление инцидентами и postmortem-митинги

# Применение DevOps вне ИТ

# DevOps как культура

- Распределенная ответственность
- Коммуникации между смежными командами
- Обратная связь и обмен знаниями
- Автоматизация