



# Основы DevOps

Преподаватель:  
Валентин Степанов

# Преподаватель

## Валентин Степанов

- Больше 20 лет в программировании
- 10 лет в автоматизации бизнес-процессов
- 10 лет в веб-программировании
- 3 года преподавания

[mail@v-stepanov.ru](mailto:mail@v-stepanov.ru)

<https://t.me/valentinstepanov>

[https://t.me/stepanov\\_software](https://t.me/stepanov_software)

# DevOps: современный подход к организации работы ИТ

Модуль 1. Введение. Основы подхода DevOps

Модуль 2. Контекст подхода DevOps: Разработка и сопровождение

Модуль 3. DevOps для сотрудников ИТ

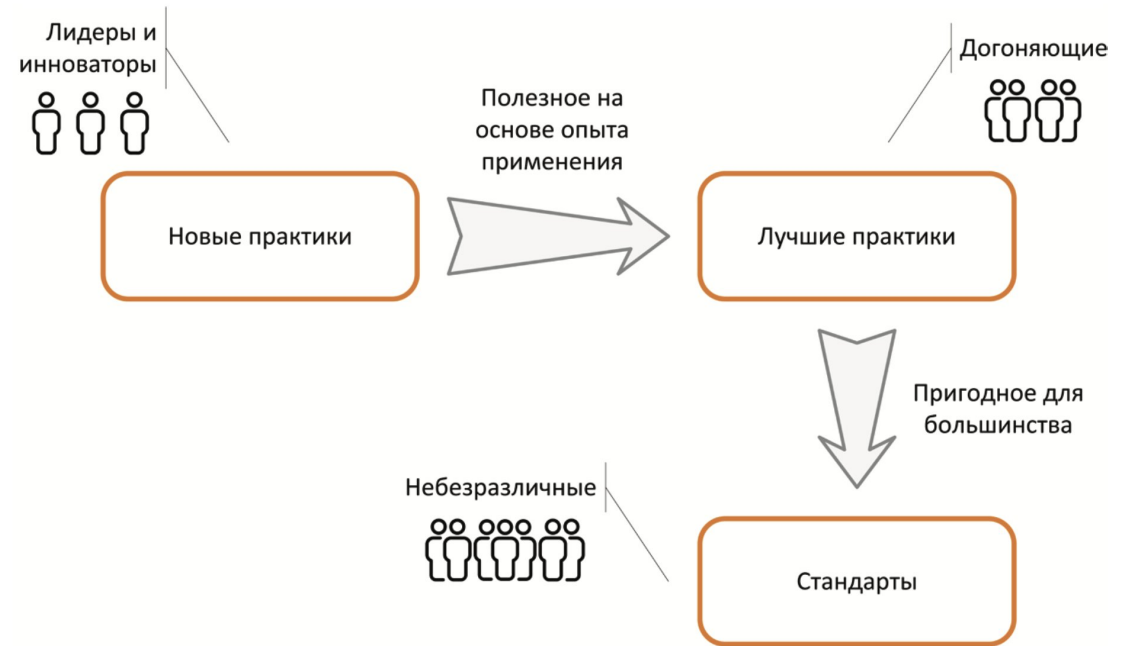
Модуль 4. DevOps для команд и компаний

Модуль 5. Практики DevOps

Модуль 6. Шаги по внедрению DevOps подхода

# История

- ITIL (2000–2001)
- BS 15000–1:2002  
«Управление ИТ-услугами»  
(2002)
- ITSM (IT Service Management)
- DevOps (Development & Operations)
- COBIT 5 (2012)



# ИСТОКИ

- Развитие гибких методов разработки ПО
- Управление ИТ-инфраструктурой как программным кодом (IoT)

# DevOps

**DevOps** – продолжение идей гибкой разработки программного обеспечения и бережливого производства, примененное к полной цепочке создания ценности в ИТ, позволяющее добиваться большего от современных информационных технологий за счет **культурных, организационных и инструментальных** изменений.

# DevOps

- продолжение идей гибкой разработки программного обеспечения и бережливого производства
- примененное к полной цепочке создания ценности в ИТ
- позволяющее добиваться большего от современных информационных технологий
- за счет культурных, организационных и инструментальных изменений.

# Зачем нужен DevOps

- Уменьшение времени вывода на рынок:
  - рождение гипотезы
  - реализация гипотезы на практике
  - измерение результата, A/B-тестирование, сравнение с целевыми значениями
  - корректировка по итогам анализа, переход на первый или второй шаг
- Снижение технического долга
- Устранение хрупкости



# История возникновения DevOps

- 2006. Марсель Вигерманн (Marcel Wegermann). Статья о применении принципов гибкой разработки к работе системных администраторов
- 2008. Конференция Agile
- 2008. Люк Канис (Luke Kanies) [Puppet Labs]. Доклад про управление конфигурациями
- 2009. Velocity. Джон Олспо (John Allspaw) и Пол Хаммонд (Paul Hammond)
- 2009. DevOpsDays (Гент, Бельгия)
- 2013. Джин Ким. «Проект Феникс»

# Частые заблуждения

- DevOps – это часть Agile
- DevOps – это автоматизация и инструменты
- DevOps – это новая профессия

# Бережливое производство

- Потери
  - Мури – работа сомнительной ценности
  - Мура – уровень плавности, равномерности, предсказуемости потока выполнения работ
  - Муда – потери, проявляющиеся в процессе выполнения работ

# Виды потерь на производстве и их аналоги в ИТ

- Запасы – частично выполненная работа
- Дополнительная обработка – дополнительные этапы
- Перепроизводство – дополнительные возможности
- Транспортировка – многозадачность
- Ожидание – ожидание
- Перемещение – передача работы и информации
- Дефекты – дефекты

# Дополнительные потери

- Управленческие расходы
- Продукция или сервисы, не соответствующие ожиданиям и потребностям клиентов
- Неиспользование творческого и интеллектуального потенциала сотрудников
- Неиспользование ресурса работников для улучшения процессов и технологий
- Недостаточное обучение персонала
- Использование некорректных метрик
- Неэффективное использование информационных систем

# Применение идей бережливого производства

1. Использовать специализированные инструменты для выявления потерь
2. Применить другие специализированные инструменты для устранения или уменьшения потерь
3. Повторите шаг 1
4. ...
5. PROFIT!!!

# Сложности применения

- Перестройка работы организации
- Lean ради Lean
- Сложность применения принципов

# Agile

- Организация самостоятельных и самодостаточных команд небольшого размера
- Итеративный процесс создания и тестирования программного кода (спринты)
- Ведение списка функциональных и нефункциональных требований (бэклог)
- Разбиение больших задач на небольшие части (истории)
- Активное вовлечение представителей заказчика в работу команды
- Регулярные краткосрочные совещания внутри команды
- Регулярные ретроспективы



# Сложности применения

- Agile охватывает только часть цепочки создания ценности
- Agile не учитывает специфику и сложность эксплуатации информационных технологий
- Движение от спринта к спринту по кругу

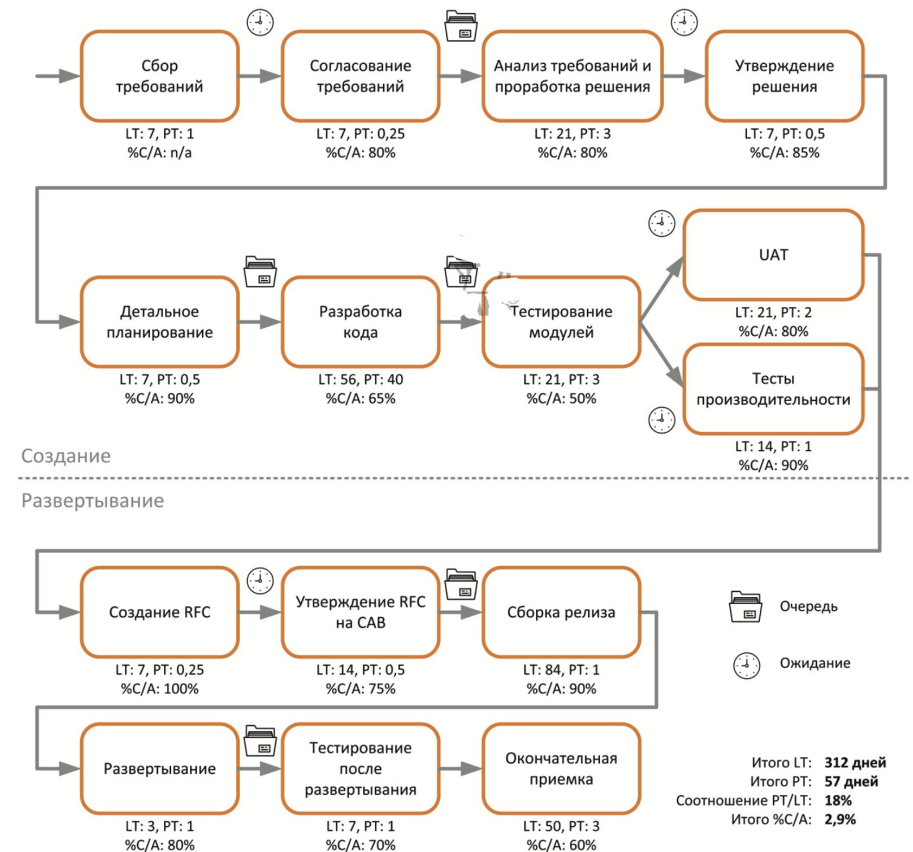
# Принципы DevOps

- Поток создания ценности (Value Stream)
- Конвейер развертывания (Deployment Pipeline)
- Система контроля версий

# Поток создания ценности (Value Stream)

- Моделирование потока – картирование (Value Stream Mapping):

- Как есть
- Как будет



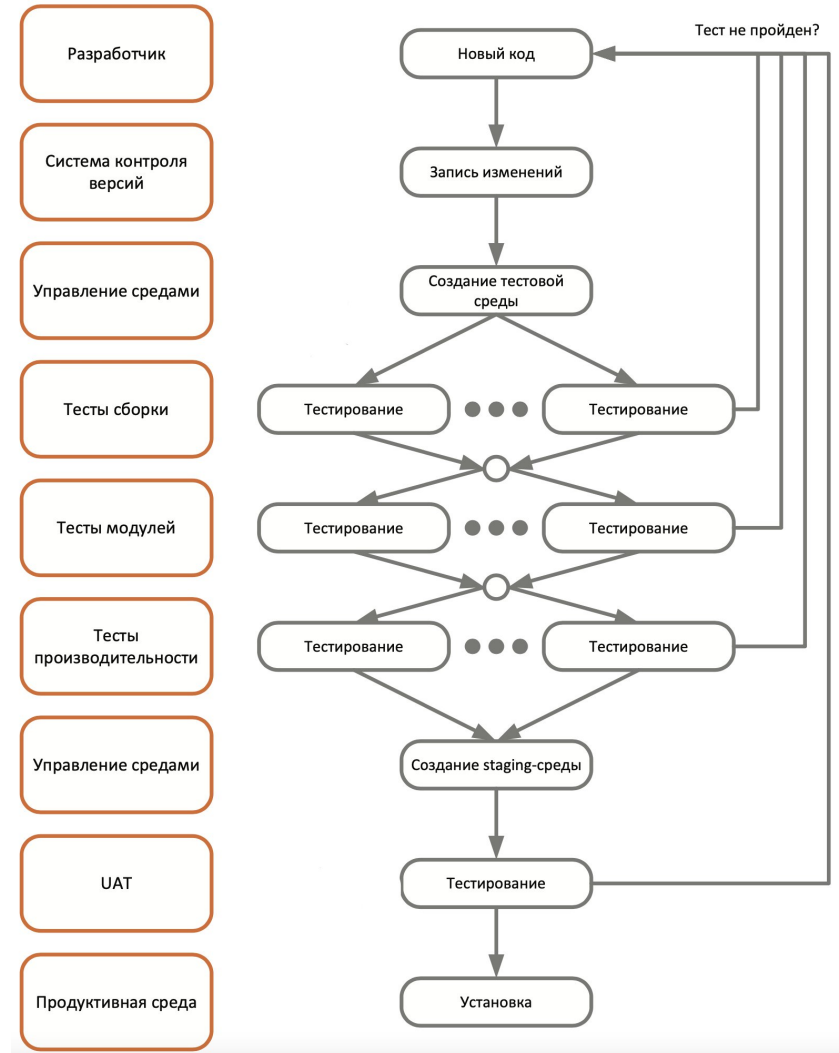
# Картирование потока

- Определить ключевые шаги обработки запроса
  - Указать суть работы
  - Не больше 15 шагов
- Выстроить шаги в последовательность получения полезного результата
- Наполнение
  - Исполнители
  - Очереди объектов
  - Ожидание
  - Время выпуска (LT)
  - Время обработки (PT)
  - Доля работ без ошибок (%C/A)

# Анализ потока

- Почему  $[\%C/A] < 100$  ?
- На что расходуется  $[LT]$  ?
- Как уменьшить  $[PT]$  ?

# Конвейер развертывания (Deployment Pipeline)



# Сложности построения конвейера

- Чрезмерное увлечение автоматизацией
- Нет достаточного количества разработанных тестов
- Большой поток небольших изменений

# Связанные понятия

- Непрерывная интеграция (Continuous Integration)
- Непрерывная поставка (Continuous Delivery)
- Непрерывное развертывание (Continuous Deployment)



# Система контроля версий



# Автоматизированное управление конфигурациями

- Любые изменения только скриптами
- Все скрипты в системе контроля версий
- Автоматическое создание сред

# Определение завершения

- Фокус на результатах, ценности для клиента
- Коллективная ответственность за общий результат команды

