

<ерат>

ТРАНСФОРМАЦИЯ В ПОСТ-AGILE МИРЕ

Декабрь 2020

CONFIDENTIAL | © 2020 EPAM Systems, Inc.



Мы поговорим о РЕАЛИЯХ [Agile] трансформации



1. Agile умер? Да здравствует... что?
2. Назад в будущее – Agile и адаптивность
3. Орг. структура в мире гибких методологий
4. Культура менеджмента: от «командования-и-контроля» до «лидерства служения»
5. Agile в узком смысле: фреймворки
6. Подводные камни трансформации
7. Методология EngX360
8. Итоги и вопросы



На примере практических кейсов разобрать опыт в сфере трансформации и больше узнать о том, как избежать ошибок и добиться конкретных результатов.



- Пожалуйста, спрашивайте и спорьте!
- Вопросы в чат
- Обсуждения в конце каждого модуля
- Сессия записывается на видео

Знакомимся...



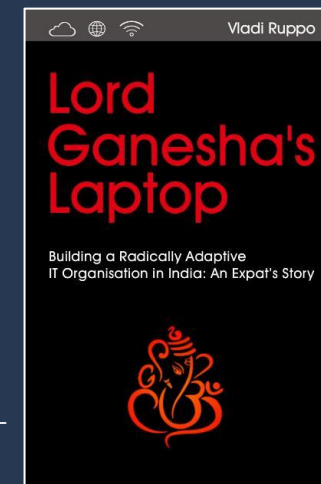
Влад Руппо (Vladi Rупpo): EPAM, глава EngX360 Transformation consulting lab

- Основал EngX360 лабораторию трансформации. Задача: радикальное улучшение [эффективности и культуры] инженерных экосистем.
- До этого, с 2000 по 2017 год работал в Индии, как глава индийского центра разработки видео-технологий Сиско (2,500 инженеров на пике).
- Провёл радикальную Agile трансформацию команды из 1,500 инженеров, разрабатывающую платформу цифрового телевидения

Автор бизнес новеллы «Ноутбук Ганеши, или Я уволил Сталина» (на русском и английском), о реалиях создания радикально адаптивной организации в Индии



https://ridero.ru/books/noutbuk_ganeshi_ili_ya_uv_olil_stalina/

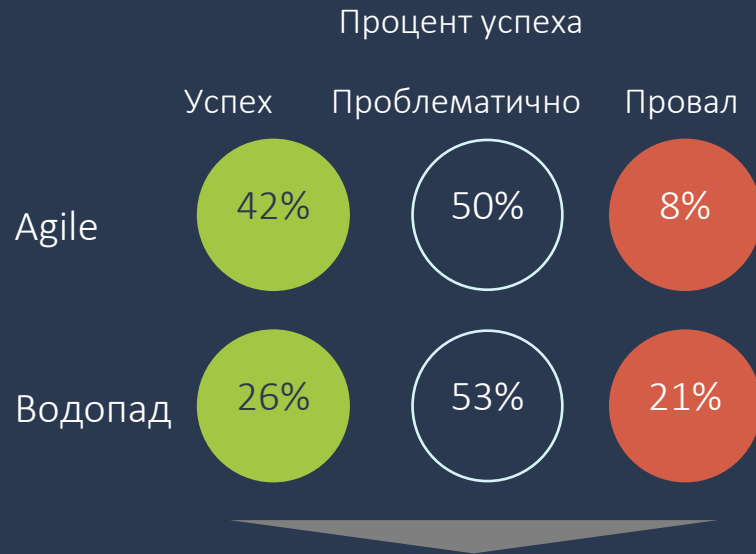


<https://www.amazon.com/Lord-Ganeshas-Laptop-Radically-Organization-ebook/dp/B08M8YWF5R>

1.

Аджайл умер? Да здравствует...

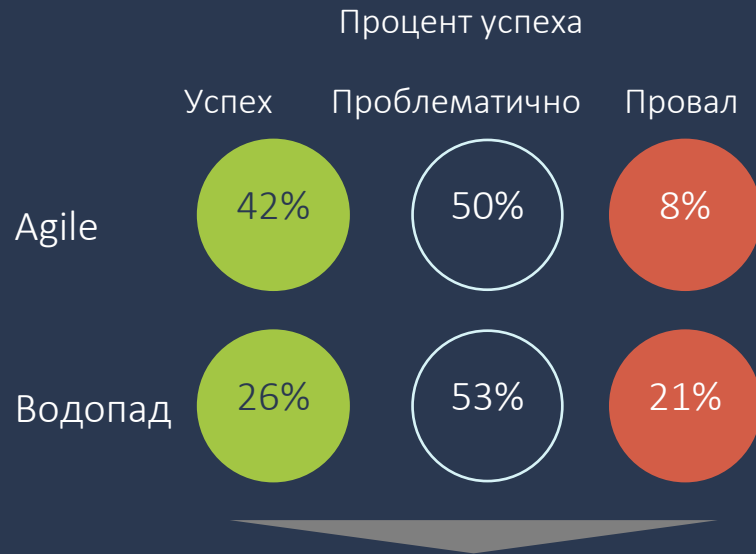
Agile парадокс: успех или провал?



Вероятность успеха проекта в Agile в **2X** раза выше, а вероятность провала на **1/3** ниже, чем в Водопаде (Source: Standish Group, 2017)

Более **70%** компаний используют Agile

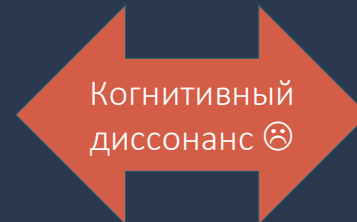
Agile парадокс: успех или провал?



Но
Agile часто не оправдывает ожиданий

84% организаций ниже удовлетворительного уровня.

От 40% до 96% трансформаций не приносят желаемых результатов.



Вероятность успеха проекта в Agile в **2X** раза выше, а вероятность провала на **1/3** ниже, чем в Водопаде (Source: Standish Group, 2017)

Более **70%** компаний используют Agile

«На первый взгляд, мир гибкой разработки программного обеспечения в прекрасном состоянии и приняты повсеместно (mainstream).

Но реальность вызывает беспокойство: большая часть - псевдо-Аджайл, который игнорирует ценности и принципы Agile.»

Martin Fowler, 2018

Иллюстрация: кейс «М»



Одна из крупнейших американских компаний в финансовом секторе «Аджайл-трансформация» проведена (SAFe фреймворк)

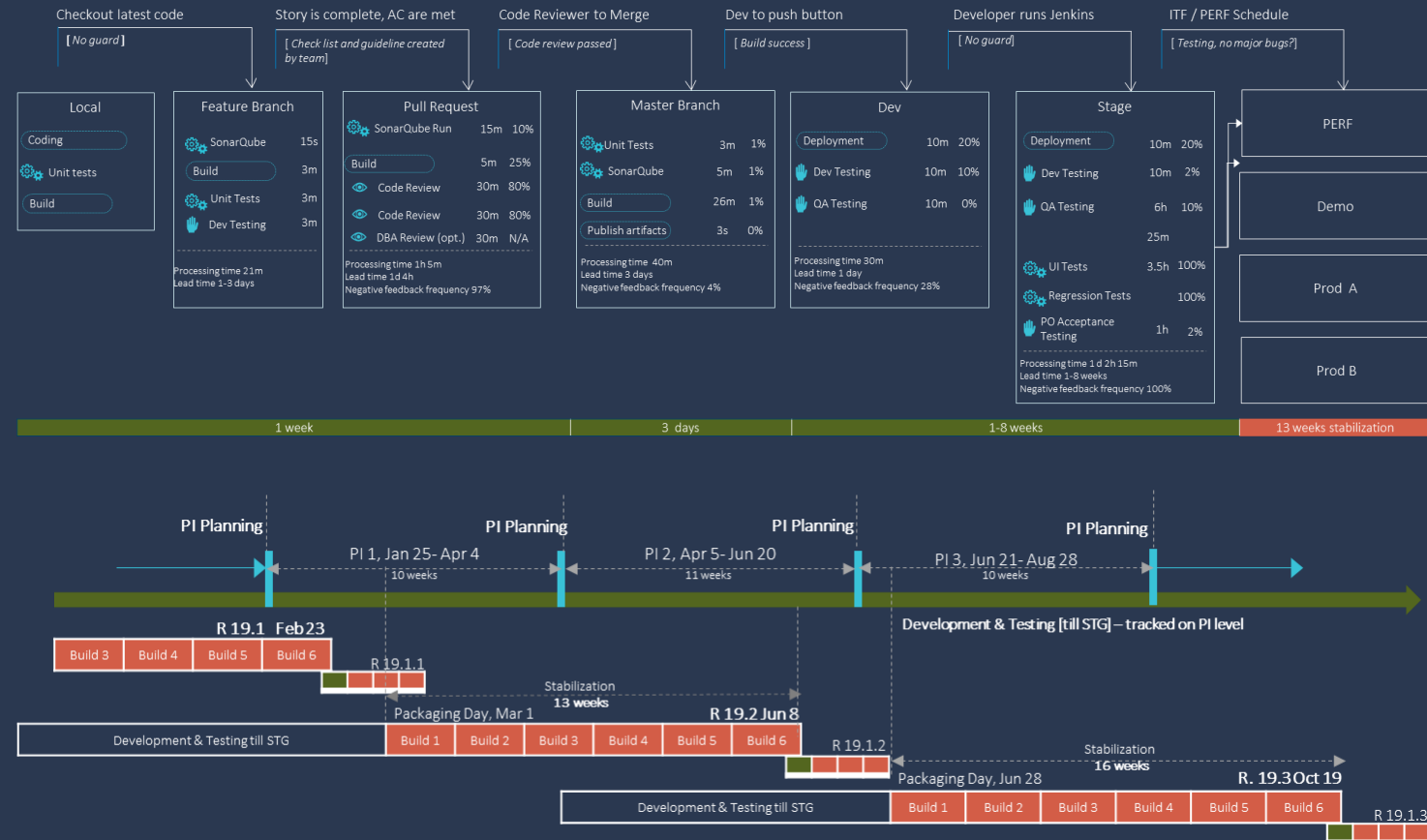


И это – Agile?!



Результаты EngX360 обледования:

- Lead Time (от бэклога до разворачивания): **22 недели**
- Эффективность процесса (время кодирования / общее время до релиза): **4.5%**
- **61%** багов исправляются позже 14 дней (примерная цена позднего исправления)
- Переделки и исправление багов (rework): **44%**
- DoD не включает автоматизацию тестирования → долг растёт.
- «Продвижение кода» (code promotion) не автоматизировано
- Проектные менеджеры руководят командами; культура «командования и контроля»



Что происходит?



Что такое Agile? Способ мышления? Фреймворк? «Дух» - или «буква закона»?



Есть ли ясные критерии («лакмусовая бумажка») Agile?



Мёртв ли Agile? Есть ли истории его реального успеха?



Нужно сперва ввязаться в бой, а там видно будет.

Наполеон Бонопарт

Итак, не заботьтесь о завтрашнем дне, ибо завтрашний сам будет заботиться о своём: довольно для каждого дня своей заботы.
Евангелие от Матфея 6:34 – Мф 6:34



2.

Проблемы: структуры и процессы

Кейс Cisco Video Technologies, India: проблема



В2В2С бизнес (цифровое телевидение)
Десятки миллионов установок
Растущий бизнес!



~1,200 инженеров в Индии
~150 инженеров в Китае



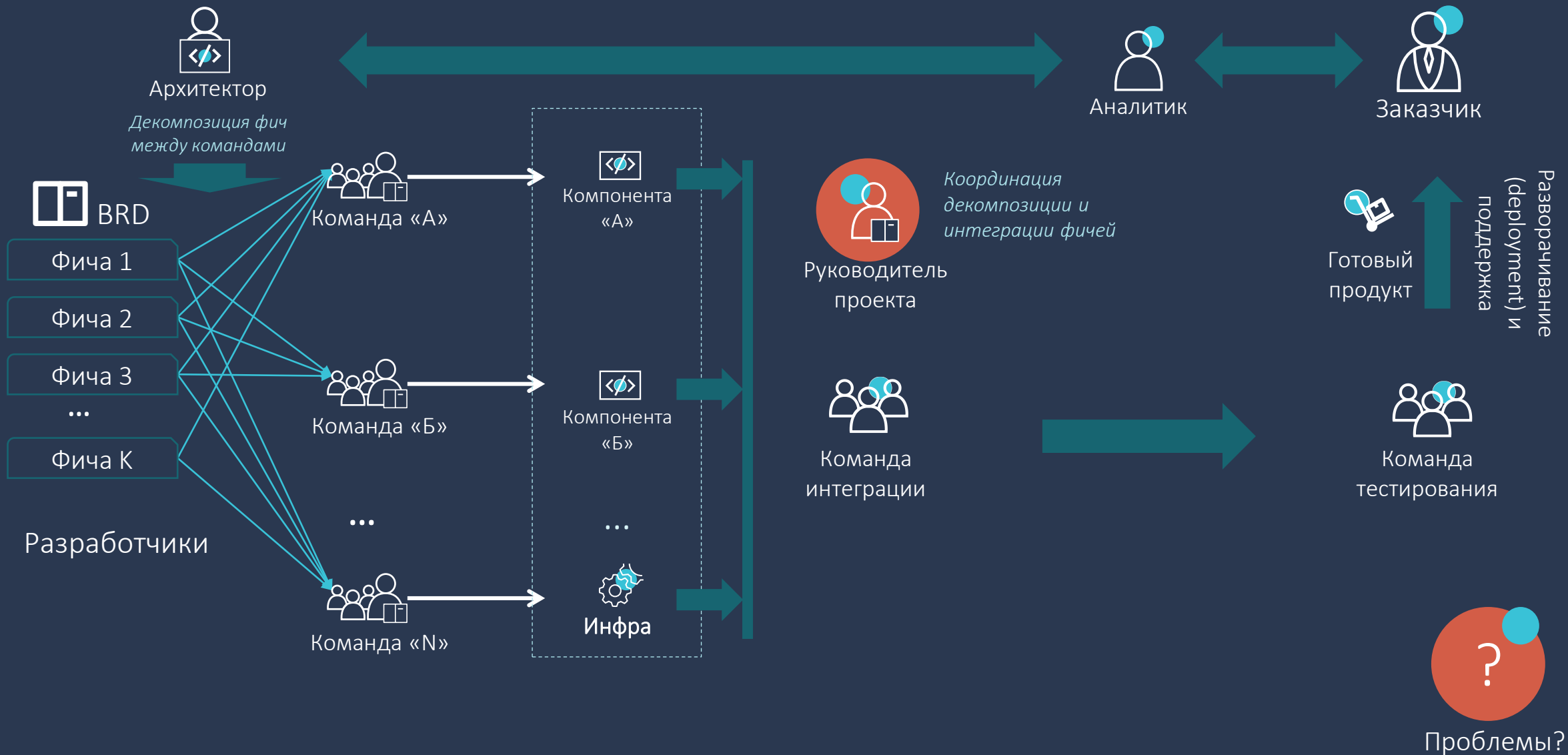
Архитектура – устаревшая;
«монолит». «С», HTML5, Java

Покупка



Повысить маржинальность на **30%**.
Сокращения вместо найма – на
растущем рынке → повысить
производительность в 2 раза

Процесс и орг. структура Cisco Video - детали



Процесс и орг. структура Cisco Video – долго, дорого, больно



Корневая проблема – парадигма собственности и подчинения



Архитектура

Закон Конвея?



Орг. структура

- У каждой функции (контроль качества, безопасность, разработка) и у каждой архитектурной единицы (компоненты, системы, микро-сервиса и т.д.) есть **команда**, которая ей **владеет**. Другие команды могут заказывать работу у команды-владельца, если им требуются изменения.
- У каждой команды есть **менеджер**, лично ответственный за состояние этой функции / компоненты.
- **Глубокая иерархия** менеджеров отражает архитектуру системы

Структура (и метрики) определяют организационную культуру и архитектуру



Локальная оптимизация

Конфликты и войны

Зависимости и долгие ожидания.



Корп.
воеводства

Интеграция между командами

«Это не моя проблема»

Что мы выиграем, если перевести команды в той же структуре на Agile?.... - НИЧЕГО 😊

3.

Решение?

Орг. структура в мире гибких методологий

Меняем парадигму?



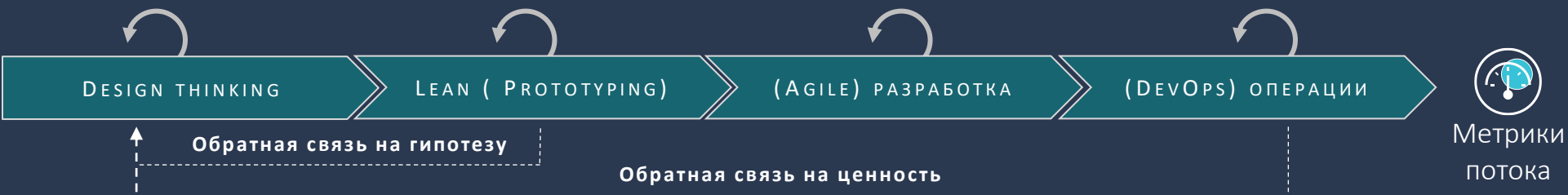
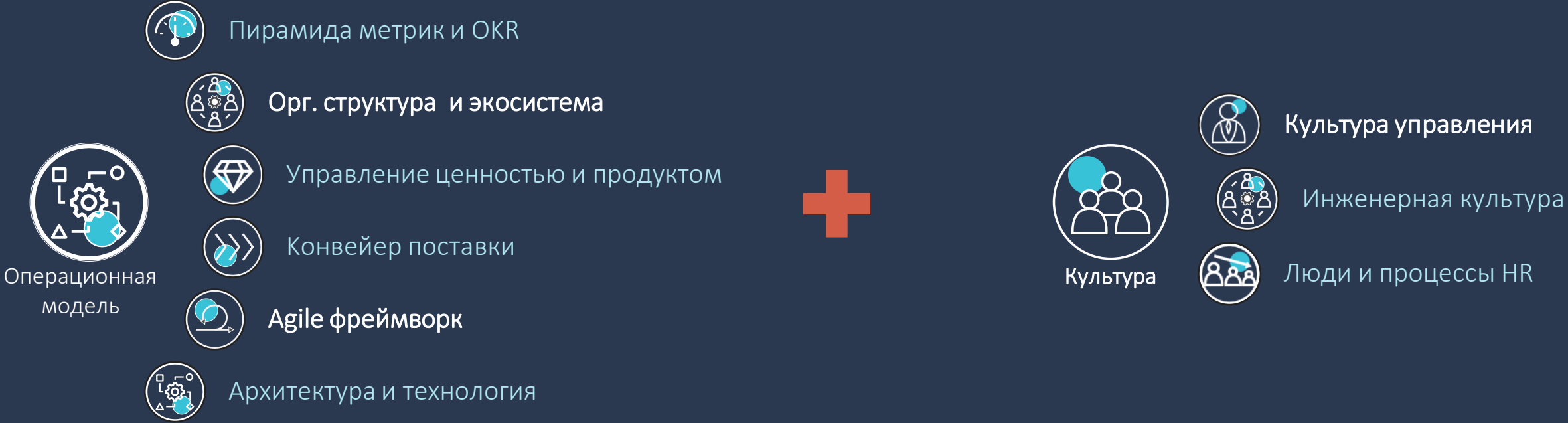
Предиктивная модель «не стреляет» по движущейся мишени (быстро меняющимся потребностям рынка).



Чтобы попасть по движущейся мишени, нужна «самонаводящаяся ракета» – **адаптивная модель** (= Agile, в широком смысле)

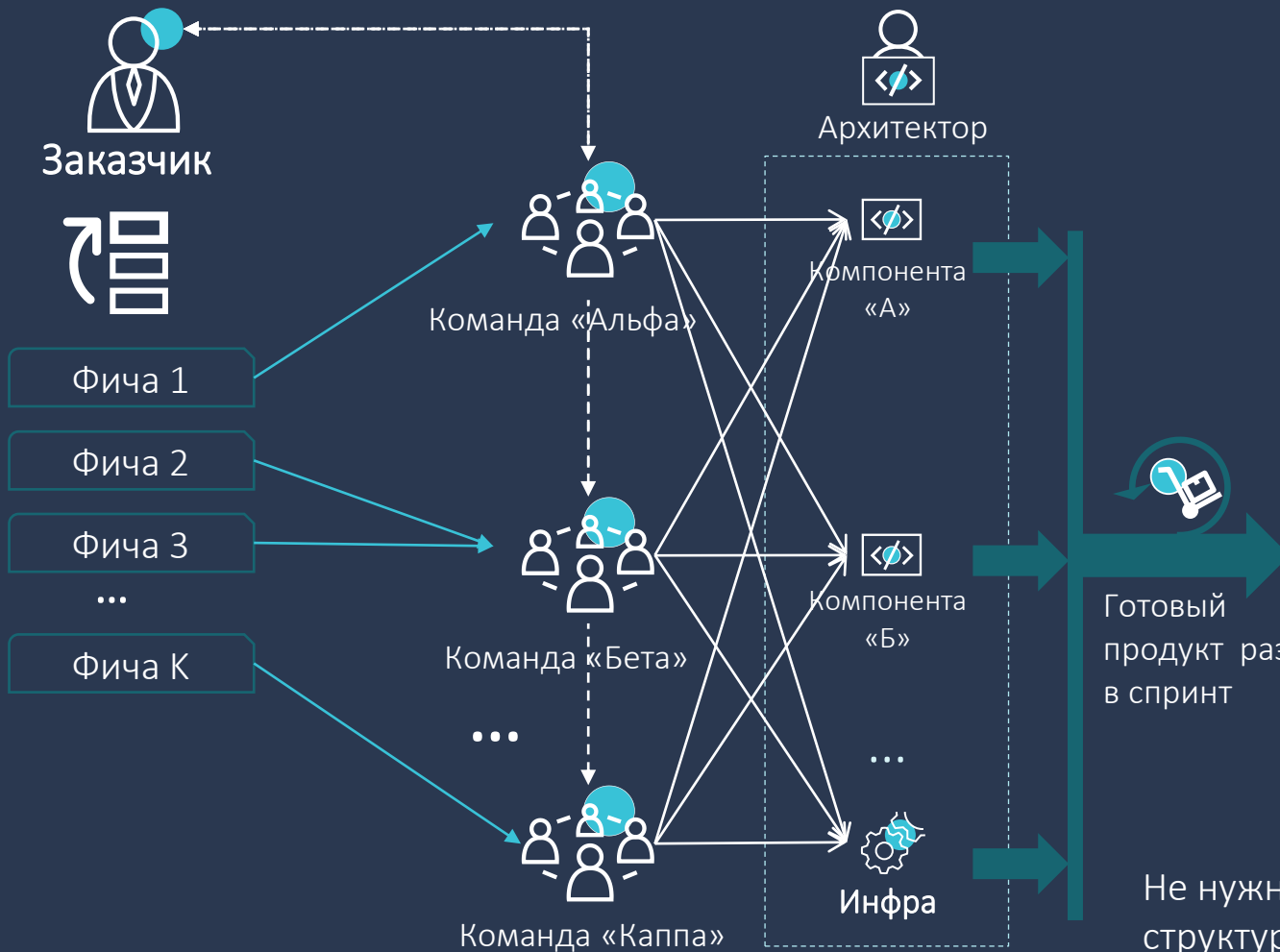


Agile больше, чем «способ мышления» и «фреймворк»: адаптивность требует многомерных изменений



Альтернатива: (кросс-) функциональные автономные команды

«Открытый код» – включая «платформу» и «инфра»



Автономные команды (а не руководители проектов) поставляют ценность клиенту →

- ✓ Рост мотивации
- ✓ «Включение мозгов»
- ✓ Команда, не «герои»
- ✓ Самоуправляемые
- ✓ Постоянно улучшающиеся

Не нужна интеграция между «воеводствами» - структура сонаправлена потоку ценности

~~Отдел тестирования~~

~~Отдел интеграции~~

~~Офис проектных менеджеров~~

~~Владельцы компонент~~

Автономность против согласованности (модель Spotify)



Как сохранить автономность команд, но сшить их в единое целое?

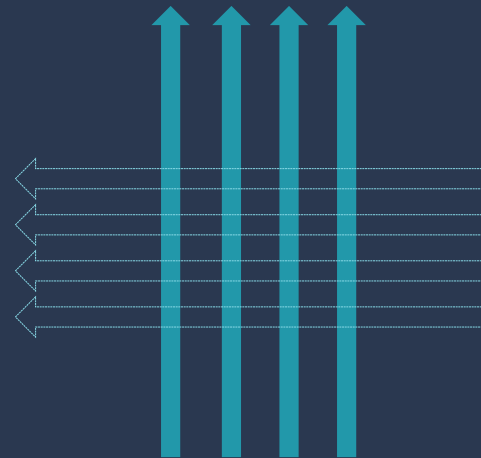


Как сохранить архитектурную целостность и не испортить код ?



Функциональные команды (Feature teams):

- Упор на ценность для клиента
- Автономность



Организационная матрица

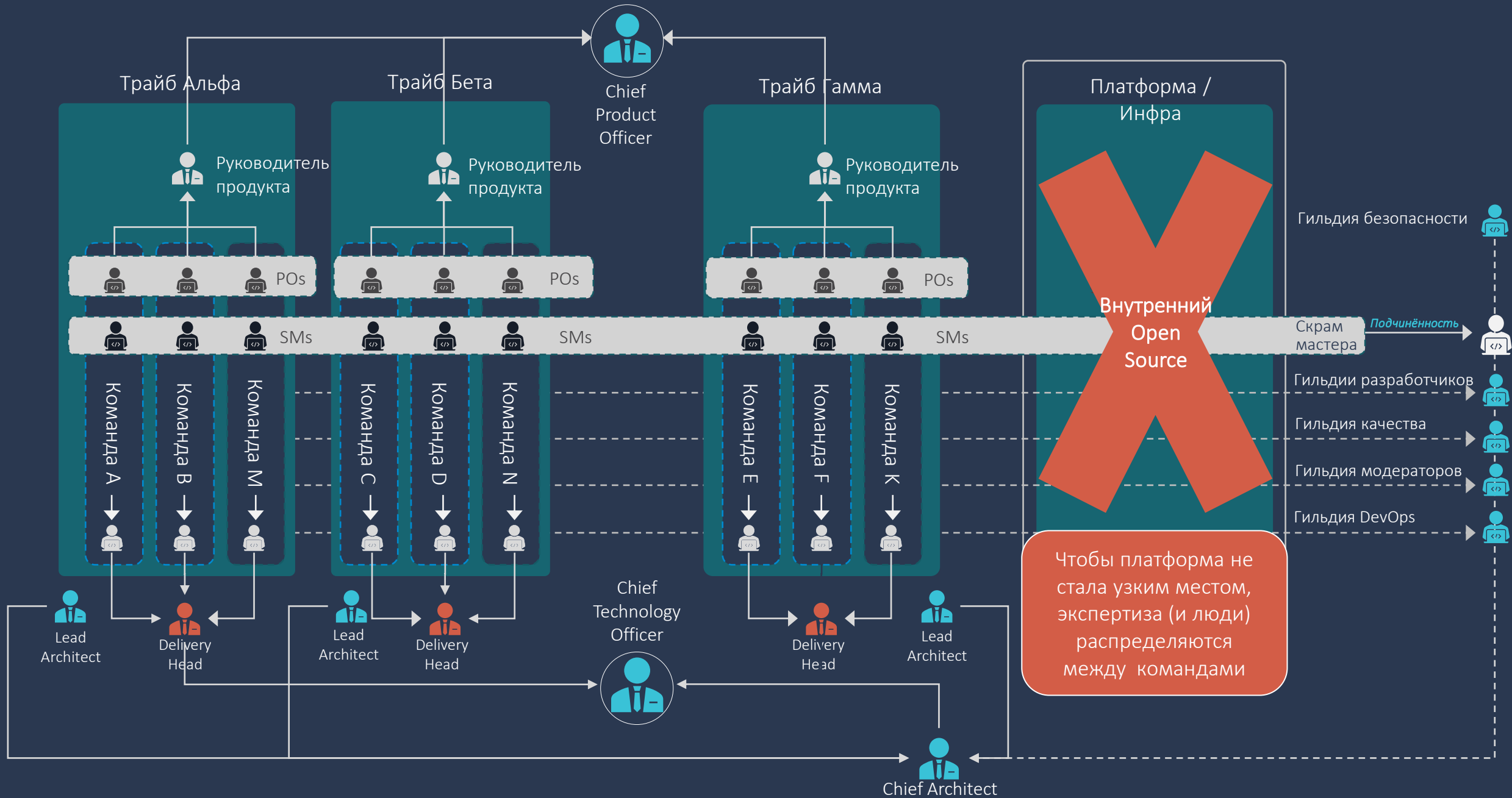


Баланс меняется со временем

Сообщества или «гильдии». Задачи:

- Архитектурная целостность
 - Обмен опытом / лучшими практиками
 - Качество кода
 - Переиспользование кода (reuse)
-
- Сообщества по компетенциям (например, front end)
 - Сообщество по технологиям (Azure cloud)
 - Сообщества по функциям (напр, тестирование и DevOps)
-
- Обеспечение работы «открытого кода»:
- Гильдия архитекторов: проверка дизайна
 - Гильдия модераторов кода (члены команд, 2 часа в день – проверка кода)

Итоги: пример «адаптивной» орг. структуры



4.

Культура менеджмента: от «командования и контроля» до «лидерства служения»

Какова роль культуры менеджмента при создании адаптивной организации?

Какова роль менеджеров – и есть ли она?

При чём тут культура?



Культура управления: определяет «включение мозгов» и умение учиться на (маленьких) ошибках



Инженерная культура: от специализации к Full stack, end-to-end и постоянному (само)обучению. От работника за конвейером – к 3D принтеру 😊



Люди и процессы HR: лояльность, заинтересованность, и построение эффективной обратной связи от сотрудников

Роль менеджеров в традиционной и в адаптивной организации

Командование-и-контроль



«Кнуты и пряники» в зависимости от результатов

Распределяет задачи и дает указания по выполнению

Проверяет результаты

«Пинает» и «мотивирует»



Ценность создаётся командой → цель менеджера - создание команды, а не поставка в срок. Владельцы продукта решают, что делать. Скрам мастера – как. Что остаётся менеджеру?

(Мета-)менеджмент, как керлинг



Руководители служат командам:

- Не участвуют в поставке (не «пинают» 😊)
- Отвечают за развитие команд.
- Устраняет препятствия в экосистеме
- Практикуют «иди и смотри»: выходят из штаба в поле

«Командование-и-контроль» и «прыжок веры»



- **Топ-менеджеры** становятся мета-менеджерами (если могут...)
- Менеджеры **среднего звена** переходят на новые роли (если могут...)
- **Команды** (не) принимают новую модель
- «**Звёзды**» могут уйти ☹️
- Чтобы уйти от «командования-и-контроля», его необходимо применить.


Трудно всем... Необходимо терпение и упорство!

5.

Agile в узком смысле: фреймворки

Agile фреймворк масштабирования – зачем?

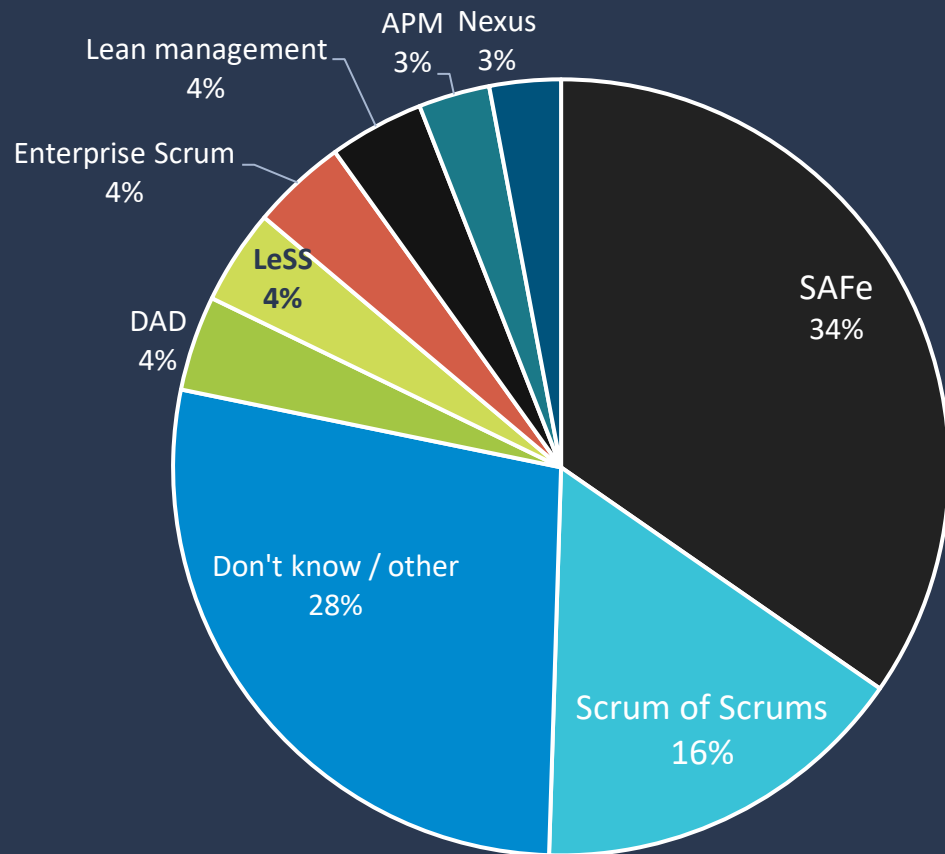
Преимущества фреймворка


Как множество функциональных команд работают вместе на большом продукте или на множестве проектов?




Что за диктатура??

Какой фреймворк выбрать?



Какая у нас частота развертывания и какой уровень «технического долга»?

Насколько мы готовы рисковать (и пить шампанское)?



Начать со скрама и канбана! Может быть, ими и ограничиться

Скрам: меню - или «комплексный обед»?

Скрам - не меню лучших практик на выбор, а целостная система, в которой все элементы взаимосвязаны



Скрам даёт лучшие результаты и легко масштабируется, когда он реализован строго «по методичке»



6.

Подводные камни и лучшие практики трансформации

Как и любой сложный и длительный процесс, Agile нельзя внедрить без ошибок. С какими проблемами организации сталкиваются чаще всего? Можно ли обойти «ловушки трансформации»?

Проблемы (руководства)



Недостаток **мотивации**. Избыток денег и времени



Нехватка дисциплины. Аджайл, как меню. **«Аджайл, но...»**. Химеры «скрамопада».



«Трансформация без изменений» - оставляем старую **операционную модель**, надеемся на улучшения культуры. Верим в чудеса 😊



«**Архитектура вначале**» – откладываем переход на новую модель из-за монолитной архитектуры.



«Трансформация проповедями». Чрезмерное **делегирование** (в т.ч. устаревшим структурам)



Неясные **цели** и метрики. Нет нацеленности на частоту релизов. Сырая операционная модель сбивает команды с толку



От 40% до 96% трансформаций не приносят желаемых результатов.

Кейс Cisco Video Technologies, India: решение 360°



Метрики: фокус на частоте релизов и Скрам метриках – burn-down, burn-up, velocity



Орг. структура: переход на автономные команды.



Конвейер поставки: авто-тестирование > 80%. Переход на trunk-based с фича-флажками. **Сирена**, когда «trunk» сломан.



Процесс: радикальный Скрам. Playbook. Agile+ коучи. Физические скрам борды. **Совместные тренинги с клиентами. Директора = скрам-мастера**



Управление продуктом: гибкость решений вместе с клиентом в рамках старых контрактов



Архитектура: глубокий рефакторинг



Культура: «мета-менеджмент». «Топы» - Agile эксперты. «Иди и смотри» и процесс устранения препятствий.

12 месяцев слёз, пота и крови.

РЕЗУЛЬТАТЫ:

Частота релизов – раз в две недели



45% меньше людей / затрат



Рост производительности в два раза



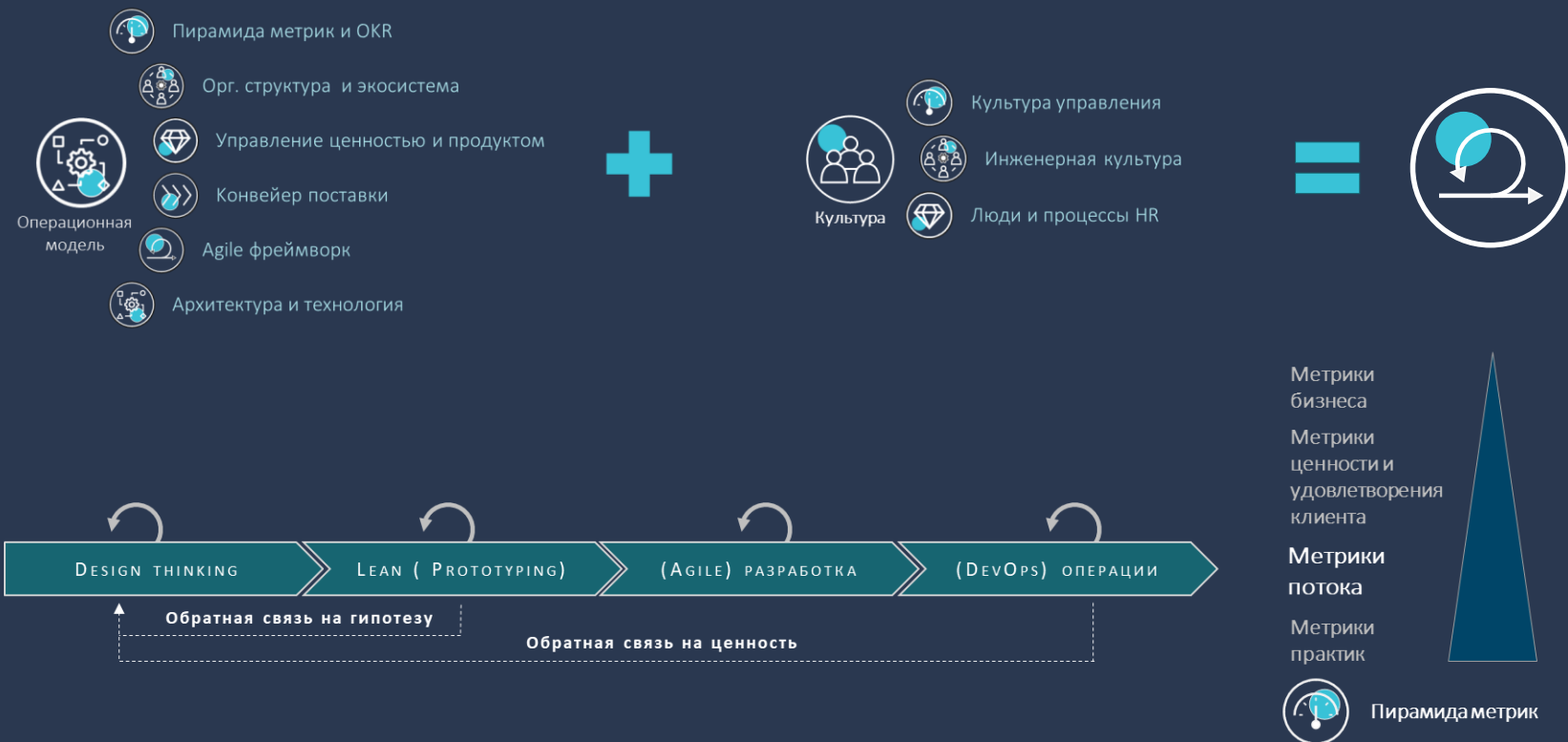
Рост бизнеса – до 80 миллионов установок



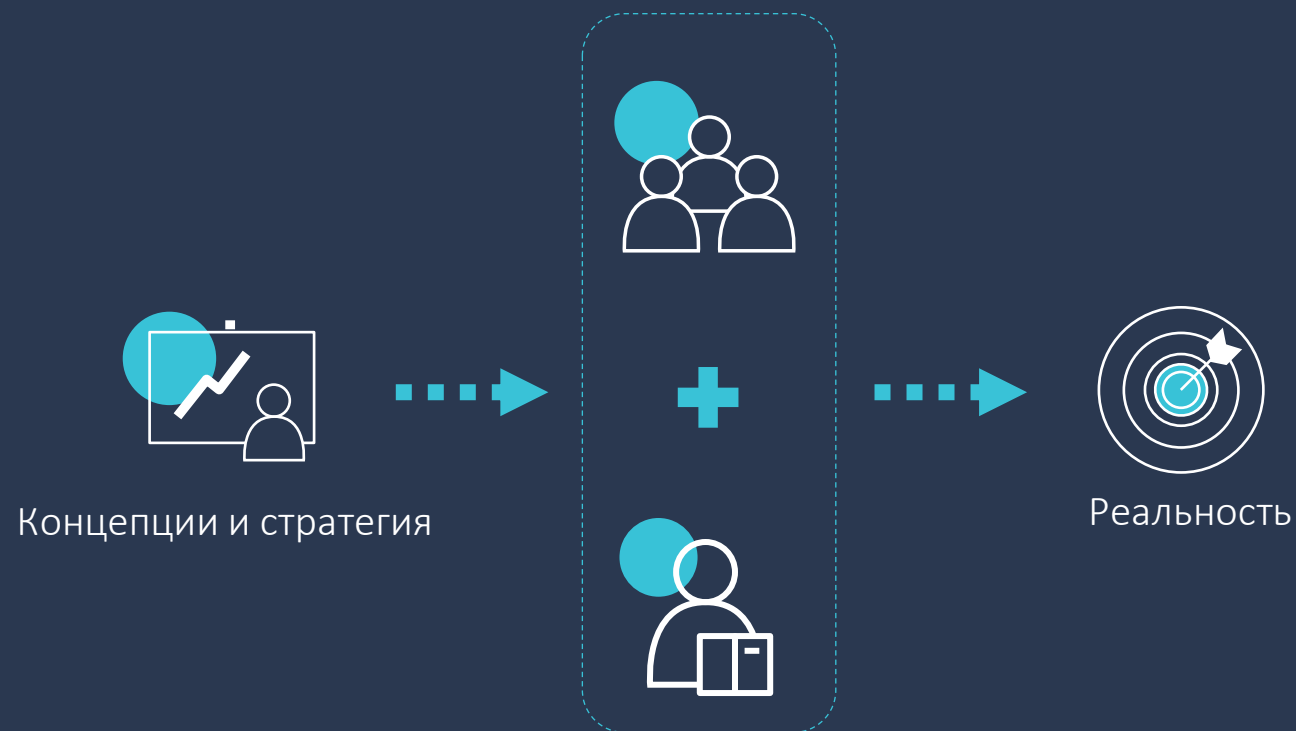
«Устаревший» продукт 15-летней давности объявлен «флагманом» Сиско

Подведём итоги: Agile умер. Да здравствует Agile 360°!

- В узком смысле, Agile, как фреймворк, или Agile, как способ мышления, - мертв!
- Да здравствует **Agile 360°**, как многомерный подход к построению операционной модели и организационной культуры, для достижения адаптивности!



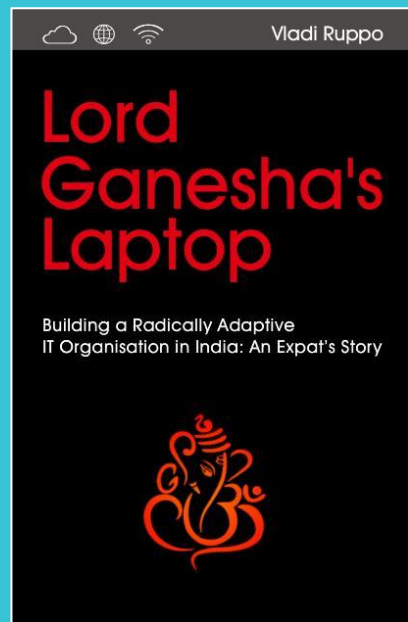
ЕРАМ предлагает: EngX360 лаборатория трансформации



СПАСИБО ВСЕМ
С НОВЫМ ГОДОМ 😊



https://ridero.ru/books/noutbuk_ganeshi_ili_ya_uvolil_stalina/



<https://www.amazon.com/Lord-Ganeshas-Laptop-Radically-Organization-ebook/dp/B08M8YWF5R>

Пишите!

Vladi.Ruppo@mail.ru: книга

Vladi_Ruppo@epam.com: консалтинг

7.

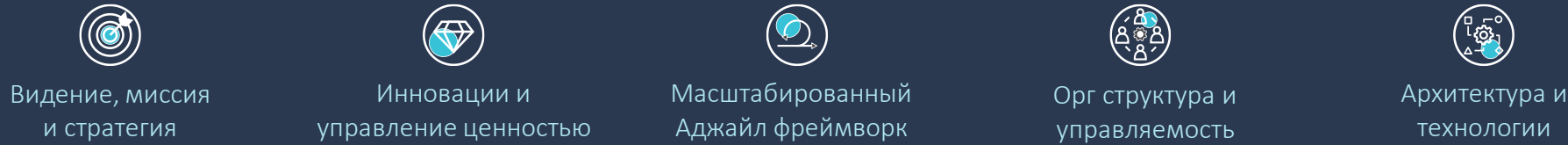
Методология EngX360

Улучшаем [операционную модель и культуру] инженерных экосистем

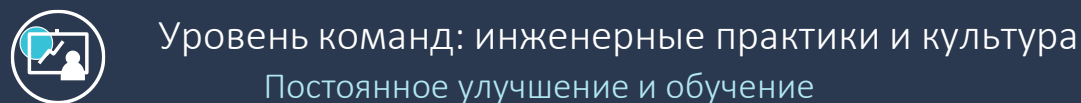
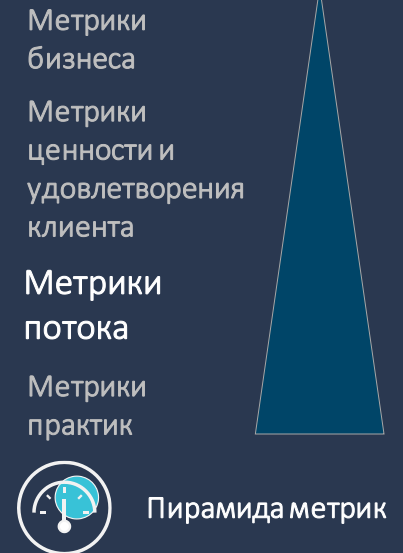
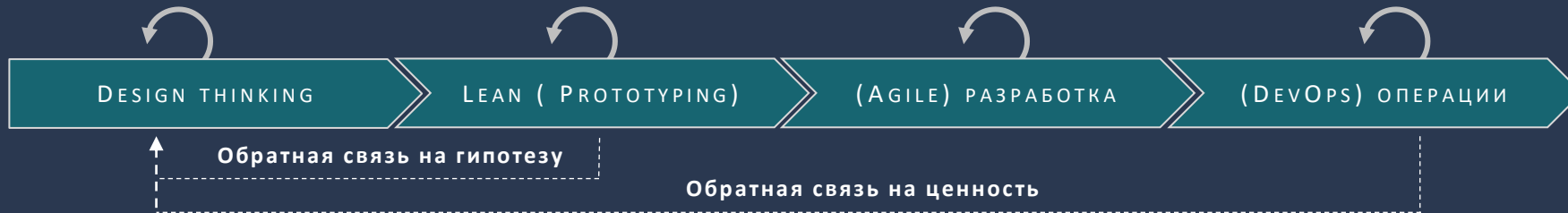
EngX360 модель адаптивной организации

Постоянная обратная связь.
Быстрая реакция.
Минимальное воздействие.

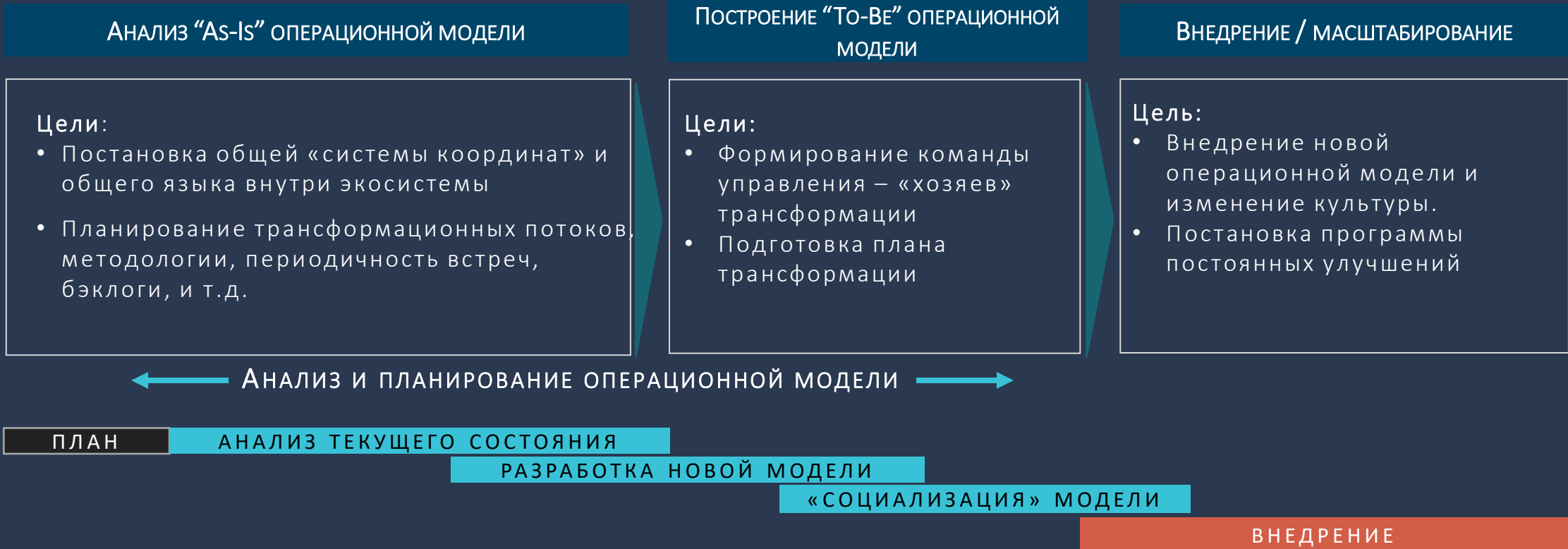
Уровень предприятия: адаптивная операционная модель



ЭФФЕКТИВНОСТЬ ПОТОКА ЦЕННОСТЕЙ И КОНВЕЙЕРА ПОСТАВОК




EngX360 лаборатория трансформации: подход



- Трансформацию не делегируют; делегируют рутину «business as usual».
- Собираем команду топ-менеджеров, лично ответственных за результаты.
- Каждый член команды ведёт один из трансформационных потоков.
- С помощью коуча команда определяет методологию работы (напр. Канбан), ритм и формат встреч и тд.

EngX360 обследование / анализ


Определение объема работ

 Число продуктов / потоков ценности

 Инженерные практики

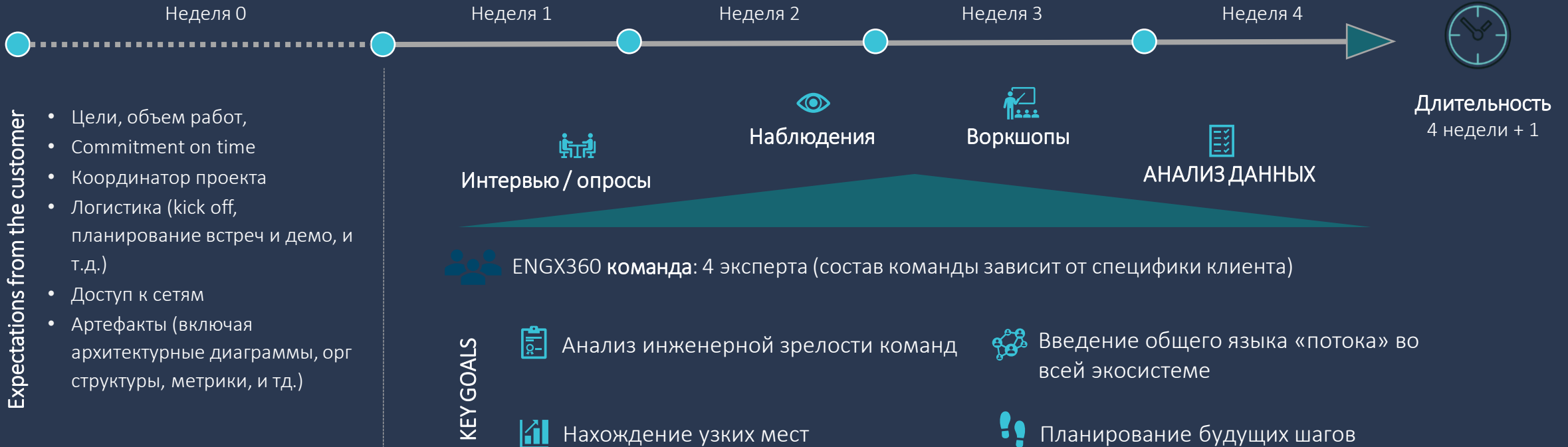
Итеративная работа

 Еженедельные встречи / ретроспективы

 Еженедельные отчеты (что сделано, что запланировано, какие проблемы): обратная связь от клиента и изменения курса

Представление конечных результатов:

- 1 OKR и пирамида метрик
- 2 Поток ценности и конвейер поставки
- 3 Тепловая карта инженерных практик
- 4 Орг. структура и цикл разработки



Календарь трансформации (иллюстративный)



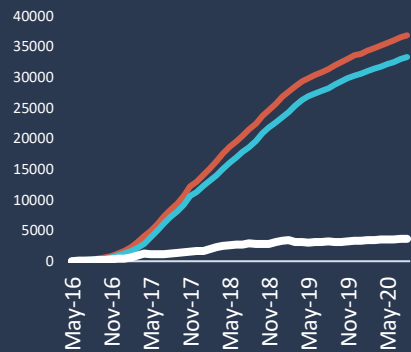
Пример обследования



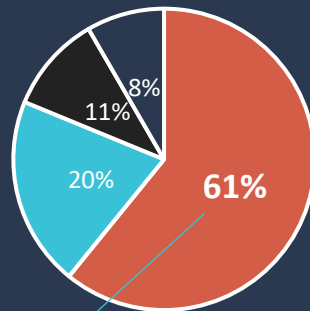
Успешная американская компания с центром разработки в России
Руководство считает, что компания работает по Скраму



Результаты EngX360 обследования:



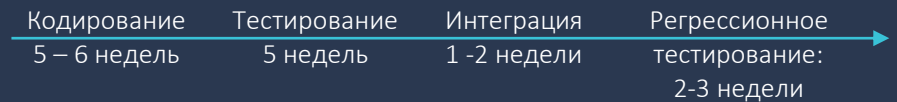
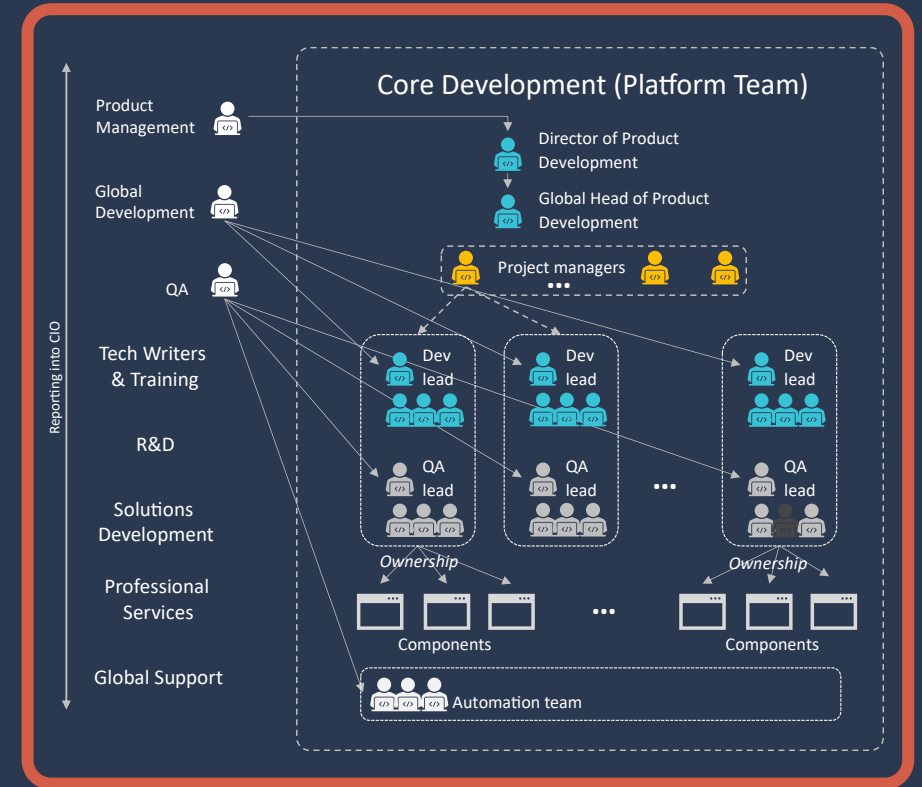
Общее количество багов растёт



Переделки и исправление багов (rework)



Пирамида (?) тестирования



Эффективность процесса – 24%

Пример (пирамиды) метрик



Метрики бизнеса и ценности

- ↑ Выручка / Прибыль
- ↑ Доля рынка
- ↑ Индекс удовлетворенности клиентов (CSAT / NPS)
- ↑ Степень лояльности сотрудников (ESAT)
- ↑ Уменьшение затрат (Cost Saves / Avoidance)
- ↑ Запуск новых сервисов/продуктов
- ↑ Выручка от новых сервисов/продуктов
- ↑ Эффективность конверсии
- ↑ Уровень удержания клиентов
- ↑ Процент фич/гипотез, на которые получена обратная связь по ценности.



Метрики потока

- ↑ Частота релизов
- ↑ Производительность (результат-проведенная работа)
- ↓ Скорость реализации функционала (backlog to production)
- ↓ Скорость внедрения кода (code change to done/production)
- ↓ Затраты на дефекты (% исправления ошибок vs разработка новой функциональности/время)
- ↓ Частота отказов (% изменений вызывающих P1/P2 инциденты)
- ↓ % дефектов, найденных в продакшн
- ↓ Среднее время на восстановление после сбоя (общее время неработоспособности/количество сбоев)



Метрики практик

Scrum

- ↑ Скорость работы команды, % (нормализованная, story point)
- ↑ Выполнение целей спринта, % (запланировано vs реализовано, в story points) – целевое < 80%. Тренд на стабилизацию
- ↓ Увеличение объемов спринта (добавление story point в ходе работ), %

Разработка

- ↓ Технический долг (часы)
- ↑ Процент удачных сборок (%)
- ↓ Количество изменений (к-во строк кода/merge-реквестов)
- ↑ Частота коммитов
- ↓ Концентрация дефектов
- ↓ Цикломатическая сложность кода
- ↑ Покрытие тестами (Unit/интеграционные/e2e)
- ↑ Процент код-ревью (%)
- ↓ Скорость изменения кода (Code churn) (< 30%)

Тестирование

- ↑ % Покрытие кода тестами
- ↑ % Покрытие функциональности автотестами
- ↓ Кумулятивный поток дефектов (открытые и закрытые)

EngX360 лаборатория трансформации

