



Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Гибкие методологии управления проектами

Александр Колесников

к.т.н., PMP (PMI), RTE, KMP, PSM I ([scrum.org](https://www.scrum.org)), руководитель подразделения разработки категорийного продукта для пользователей в «Яндекс Маркет»

Цель курса

- изменить образ мышления
- показать, что такое гибкие методологии управления проектами (ГМУП)
- изучить основной инструментарий и подходы
- оценить целесообразность использования ГМУП с учетом специфики его работы

Содержание курса

1. Предпосылки
2. Основы
3. Организационная гибкость
4. Scrum
5. Lean
6. Приоритеты и оценки задач и проектов; метрики процесса и продукта в гибком подходе
7. Гибкие команды и управление людьми
8. Как начать применять гибкие практики на уровне организации
9. Целеполагание

Основная литература

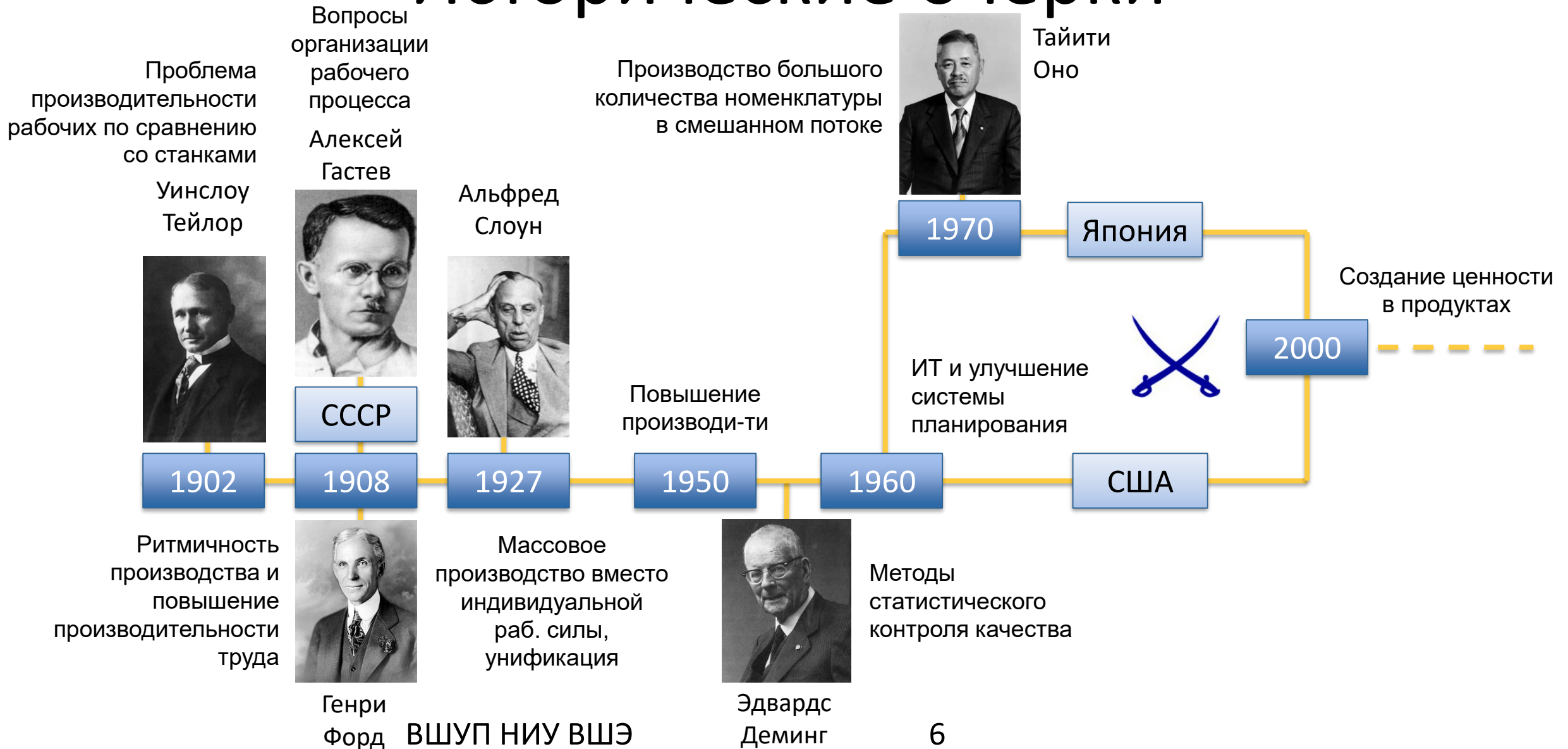
1. Scrum guide <https://www.scrum.org/resources/scrum-guide>
2. Джефф Сазерленд. Scrum Революционный метод управления проектами
3. Kanban guide <http://leankanban.com/guide/>
4. Дэвид Андерсон. Канбан. Альтернативный путь в Agile
5. The Agile Samurai: How Agile Masters Deliver Great Software, Jonathan Rasmusson
6. Agile manufacturing transitional strategies, Ian Christian and others, 2001
7. Быстрая разработка программ. Принципы, примеры, практика, Роберт К. Мартин, Джеймс В. Ньюкирк, Роберт С. Косс
8. Peopleware: Productive Projects and Teams, Tom DeMarco, Timothy Lister
9. Succeeding with Agile, Mike Cohn, 2009
10. Agile Product Management with Scrum, Roman Pilcher, 2010
11. Agile Estimation and Planning, Mike Cohn, 2005
12. Lean Analytics: Use Data to Build a Better Startup Faster, Croll, Alistair and Yoskovitz, Benjamin, 2013
13. Lean from the Trenches: Managing Large-Scale Projects with Kanban, Kniberg, Henrik, 2012



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Тема 1. Предпосылки

Исторические очерки



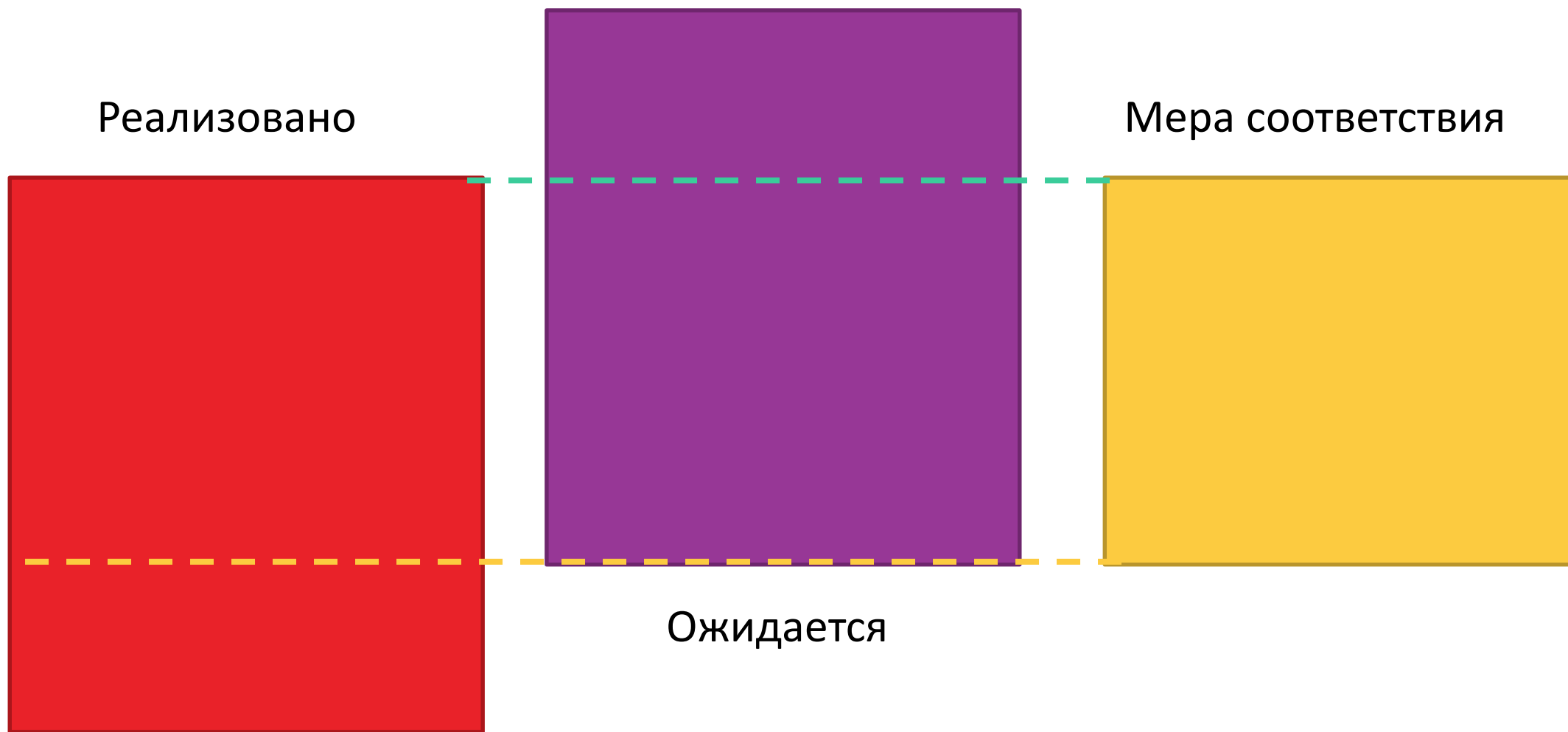
Другими словами

- От осознания проблем в управленческих процессах, через стандартизацию, оптимизацию, проектный подход, статистическое управление и т.д., мы подходим к управлению продуктами и созданию в них ценности
- ГМУП – это о том, как делать продукты, создавать дополнительную ценность

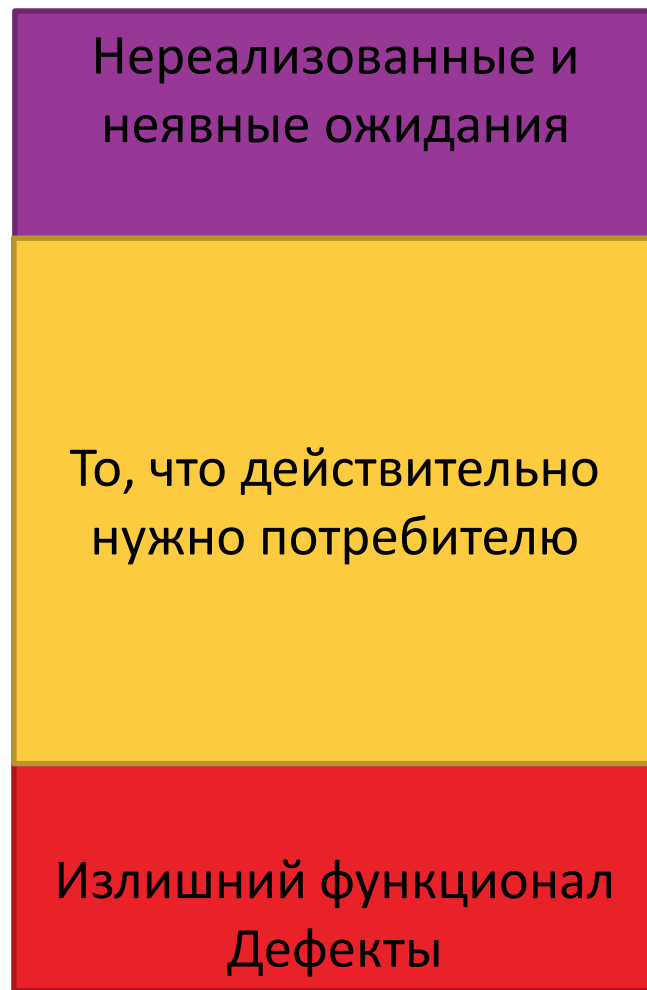
Что такое создавать ценность в продукте?

- Создание любого продукта – это не вещь в себе, это не работа ради работы, это не технологические операции сами по себе
- Это удовлетворение потребности в чем-то.
 - В быстрой езде
 - В точных расчетах
 - Во вкусной еде
 - В защите данных
 - ...
- Это продукт, совокупность процессов и артефактов вокруг этого продукта, направленные на удовлетворение той или иной потребности потребителя этого продукта

На сколько качественно мы можем удовлетворить потребности?



На сколько качественно мы можем удовлетворить потребности?



Что такое качество?

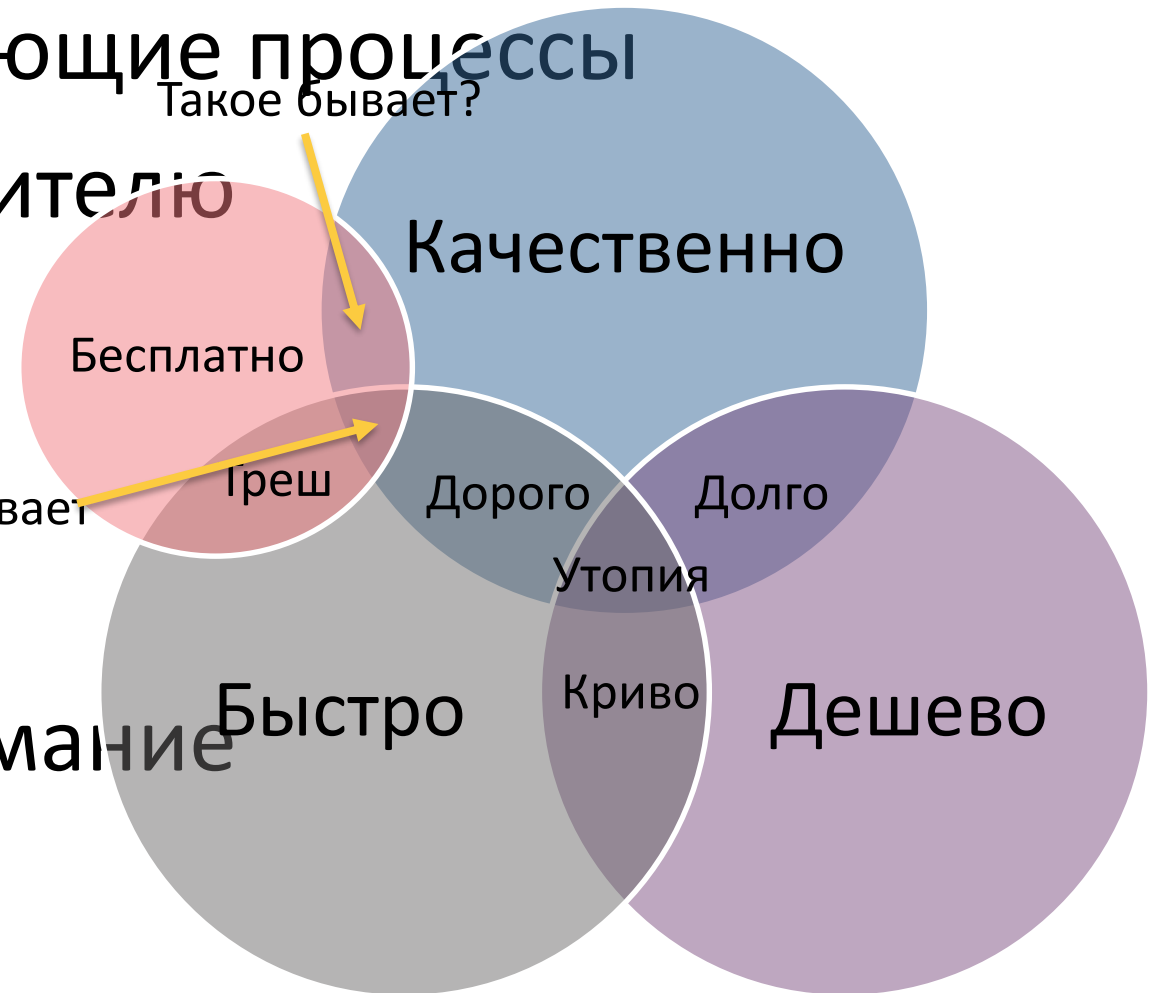
- Это степень удовлетворенности потребителя и соответствия свойств продукта его ожиданиям. (ISO 9000)



Пирамида качества Гойко Аджича

Что нужно для создания качественного продукта?

- Организовать соответствующие процессы
- Понять, что нужно потребителю
- Вписаться в бюджет
- Вписаться в сроки
- Понимать происходящее
- И уметь донести это понимание

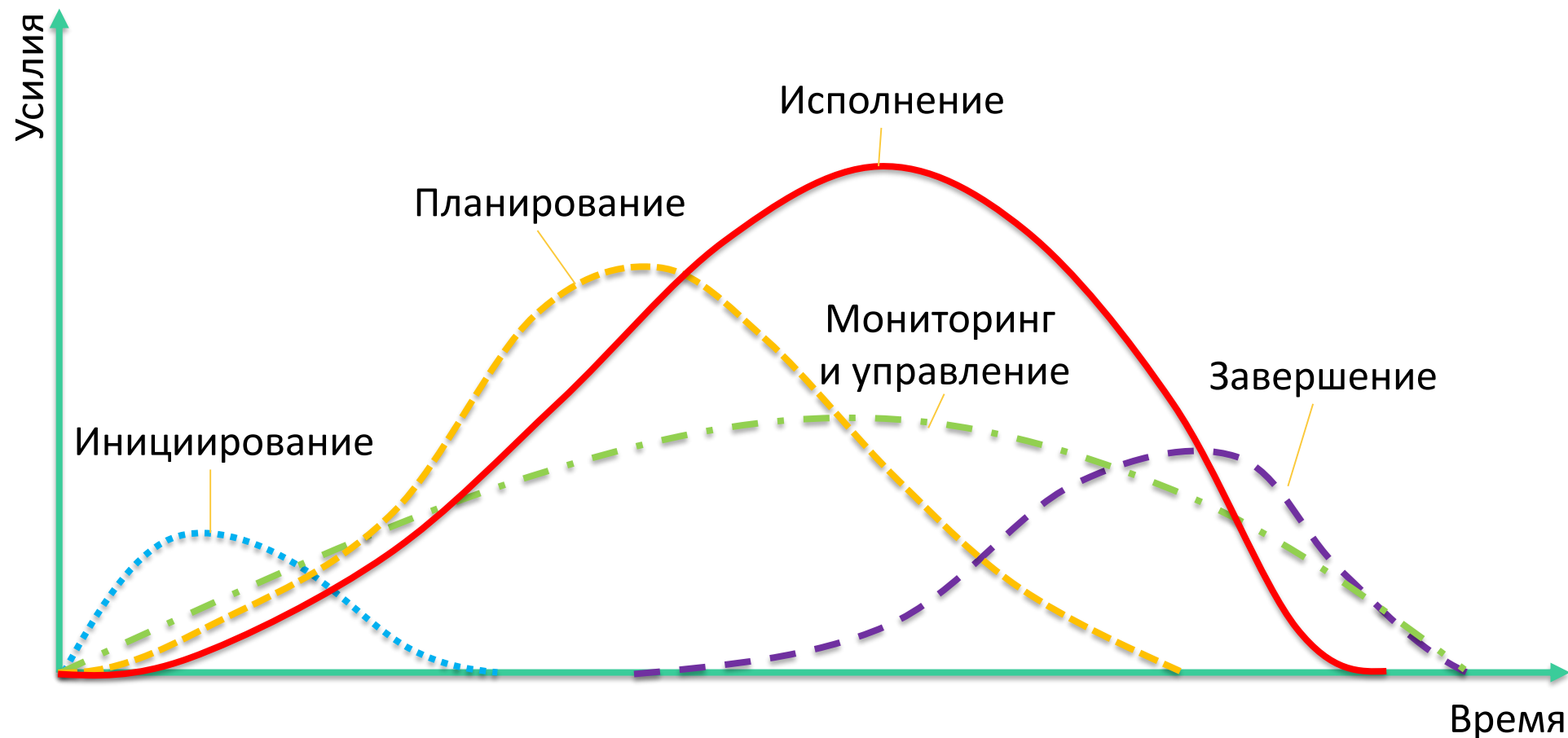


Почему это сложно?

- Потребитель не знает, что он хочет
- Требования могут меняться в любой момент
- У исполнителей нет понимания того, что они делают
- С расписанием может произойти все, что угодно
- С командой тоже
- Да и с продуктом
- Лучшее враг хорошего
- Каждой практике должно быть свое место и время
- И, вообще, все труднопредсказуемо

Какие процессы есть?

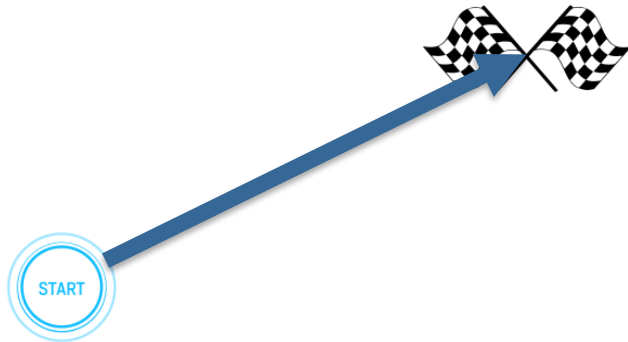
С точки зрения управления проектами



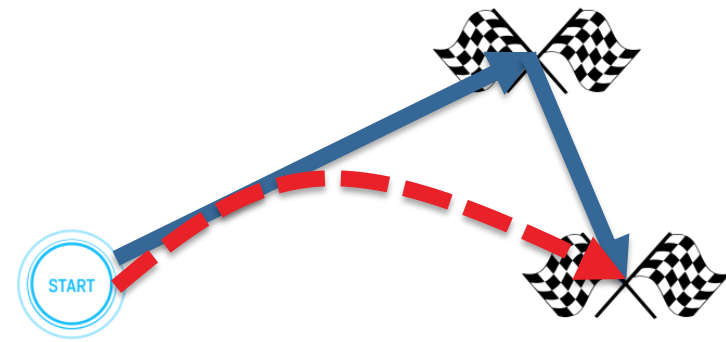
Движение к цели

с точки зрения бизнеса

Идеальный мир

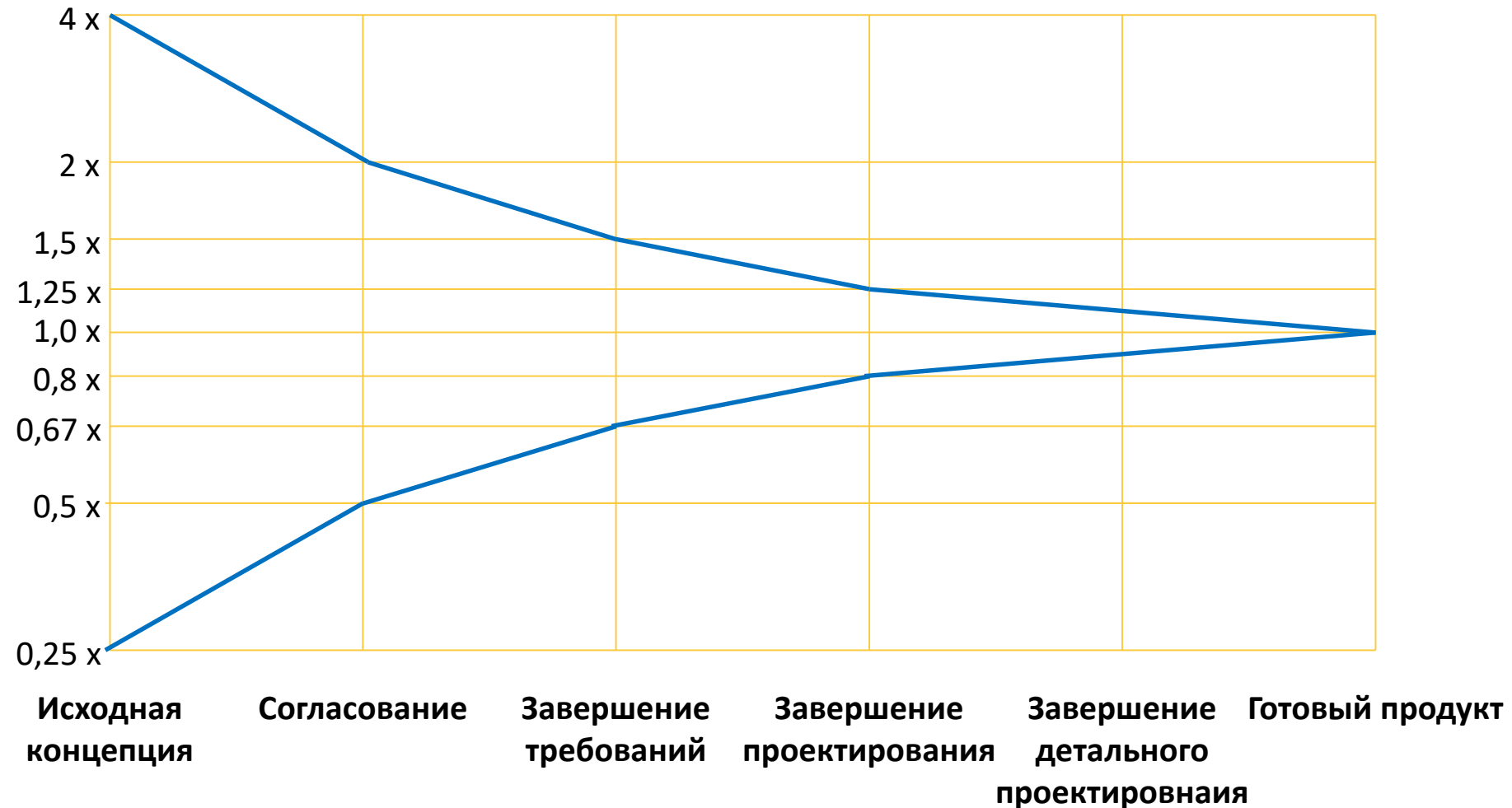


Реальный мир



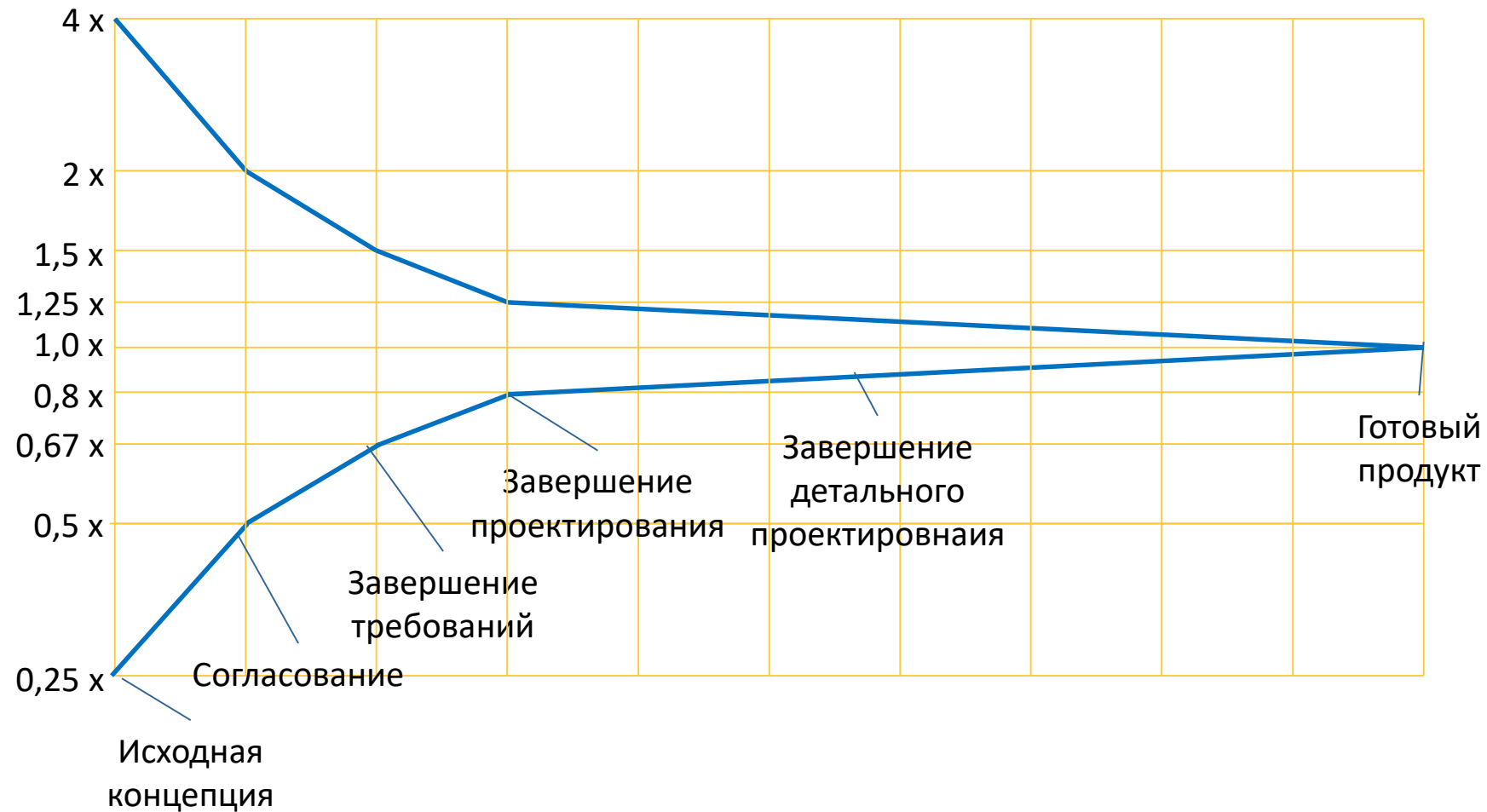
Конус неопределенности

(с точки зрения изготовления продукта по фазам)



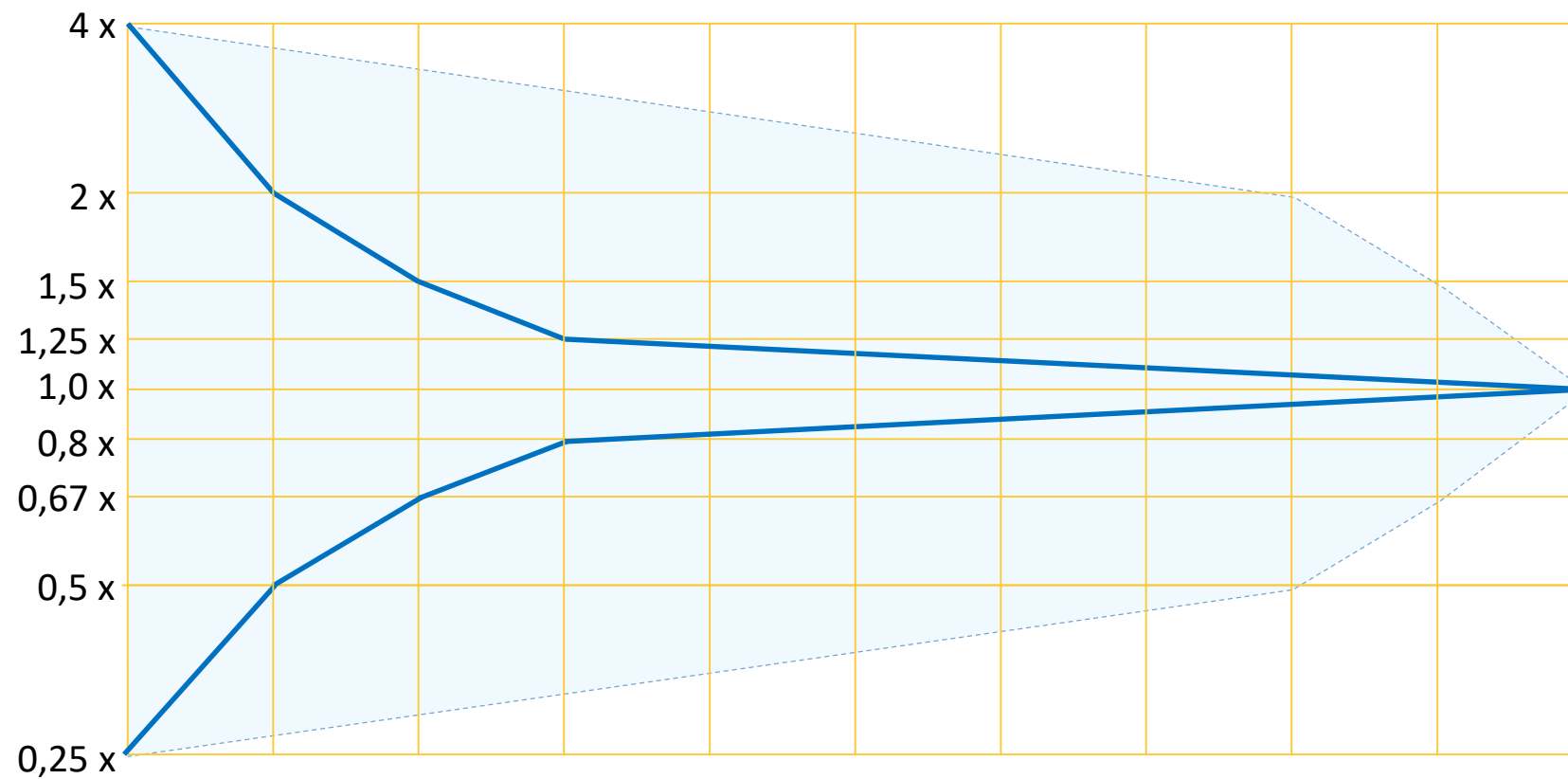
Конус неопределенности

(по времени)



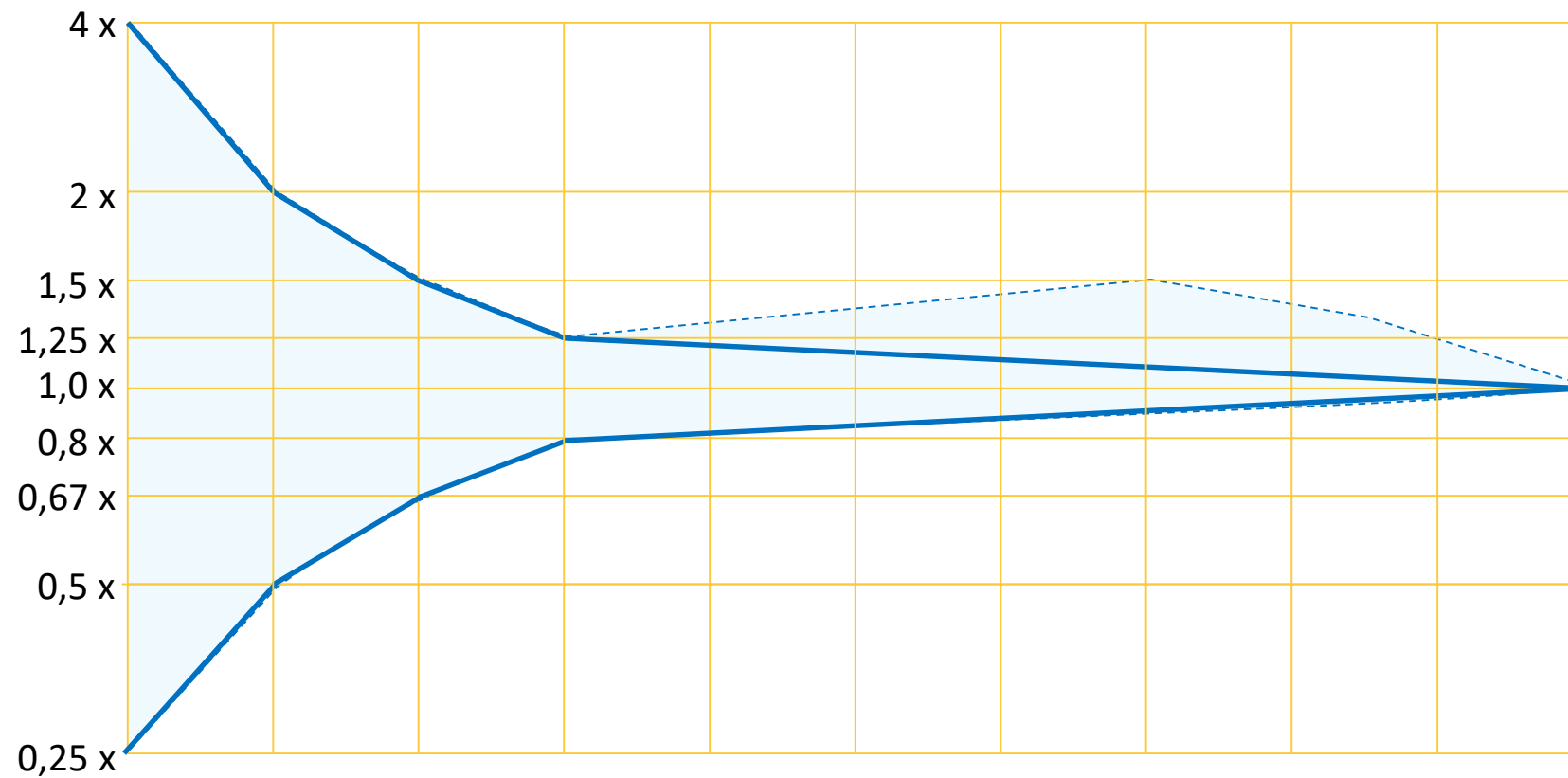
Конус неопределенности

(не работали с требованиями)



Конус неопределенности

(пришли новые требования)



Agile – это рынок покупателя

Себестоимость + Прибыль = Отпускная цена

- Монополизация рынка
- Серийное или массовое производство
- Рынок поставщика

До 60-х годов
XX века

Производительность

Прибыль = Отпускная цена - Себестоимость

- Высокая конкуренция
- Индивидуальный подход
- Рынок заказчика

После 60-х
годов XX века

Издержки

Факторы

- Разнообразие

Получаю, что я хочу

- Оперативность

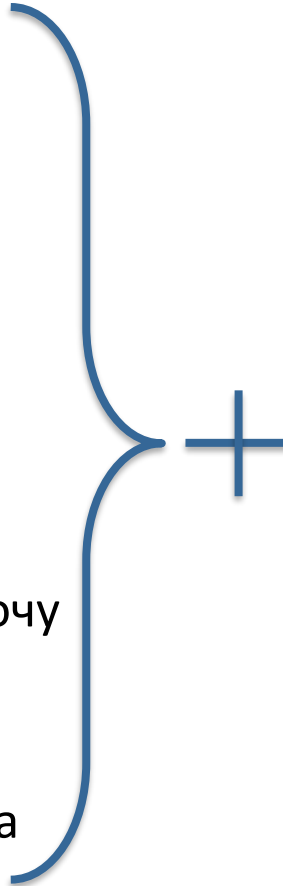
Получаю, когда я хочу

- Эффективность

Получаю, именно то, что я хочу

- Качество

Получаю без потери качества



Минимальное расстояние
в пространстве и времени
от «Купить» до «Получить»

Примеры

Банковское кредитование

- Что хочет получить потребитель?

Деньги

- Что для него издержки?

Все остальное: бумаги, поездки



ИТ (заказной софт)

- Что хочет получить потребитель?

Работающую программу или модуль

- Что для него издержки?

**Все, кроме труда программистов,
которые пишут программу**





**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Тема 2. Основы

Что влияет на создаваемые продукты?

- Все зависит от:
 - Модели
 - Методологии
 - Парадигмы
 - Язык (терминология)
 - Используемых приемов и шаблонов
- Важно понимать разницу между моделями, методологиями и парадигмами:
 - Применимость в данных условиях
 - Зрелость и готовность
 - Достоинства и недостатки тех или иных моделей
 - ...

Какие методологии есть?

- Прогнозирующие (традиционные)
 - ISO, PMP, Prince2, OPM, CMM, DSDM, ...
- Эмпирические, адаптивные(гибкие, бережливые)
 - Agile, Lean, SCRUM, Kanban, XP, TDD, BDD,...
- Смешанные (совмещающие традиционные и гибкие подходы)
 - Свои собственные, PMP, RUP, MSF, ...

СММ/СММІ

- 5-ти уровневая модель зрелости для больших организаций
- Много документации
- «Тяжелая» всеохватывающая методология
- Работа «по стандарту»



RUP/MSF/...

- Каждая большая компания разрабатывает методологию под свои нужды
- В ней стараются соединить лучшие из имеющихся методологий
- Акцентируют внимание на том, что важно для конкретно этой компании
- Если получилось удачно, то оно получает широкое распространение и вне компании
- Обычно это толстая книга, где содержатся указания о том, в каких случаях какие элементы других методологий применять

RUP/MSF/...

- Например, RUP отталкивается от продукта:
 - Итеративная разработка
 - Управление требованиями
 - Управление изменениями
 - Компонентно-ориентированная разработка
 - Использование средств визуального моделирования (UML)
 - Контроль качества ПО
- А MSF концентрируется на процессе:
 - Ролевая модель и зоны ответственности
 - Бизнес-процессы
 - Управление проектом

Agile и Lean

- эти два понятия часто идут рядом, вплоть до смешения
- истоки у них разные
- в дальнейшем, говоря про ГМУП, будет подразумеваться и Agile, и Lean

Agile

- Agile [эджáйл] – перевод с английского: «быстрый», «расторопный», «сообразительный».
- Можно определить, как набор подходов к разработке продуктов, ориентированный на параллельное, адаптивное и итеративное выполнение различных этапов жизненного цикла проекта в результате постоянного взаимодействия самоорганизующихся кроссфункциональных рабочих групп.

Lean

- Lean [лин] – это бережливое производство. Это набор принципов, относящихся к ориентированности на клиента, на качество, на скорость, на постоянное стремление к устранению всех видов потерь.

Другими словами

- Agile – более революционный подход, который дает возможность получать дополнительную ценность (added value).
- Lean – более эволюционный подход, который дает возможность сокращать издержки (reduced debt).

Противостояние традиционных и гибких подходов

- Это миф
- Это ответная реакция на перегруженность на неразумное использование существующих подходов
- Это эволюционное развитие традиционных подходов
- Это органичное дополнение традиционных подходов управления проектами

Agile и PMBOK гармонизированы

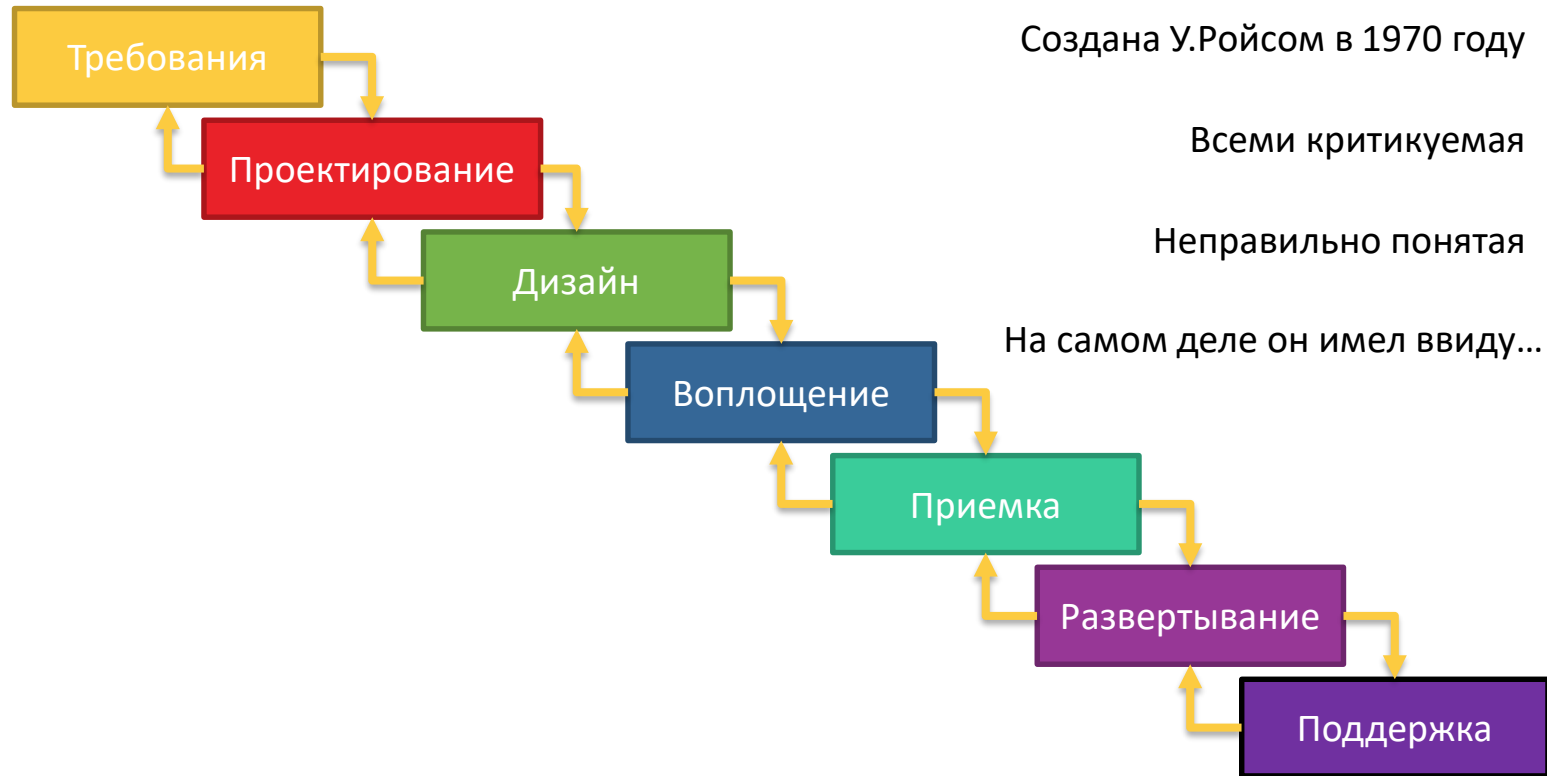


- Уже в шестой стандарт PMBOK вошли соображения для Agile
- Выпущен Agile Practice Guide:
 - Внедрение agile на уровне проекта или команды
 - Описание распространенных agile-подходов
 - Факторы, определяющие, нужен ли вам agile
 - Соотнесение agile с областями знаний и процессами PMBOK
 - Agile за пределами IT
 - Общее руководство, техники и соображения по внедрению agile в организации
 - Определения наиболее распространенных понятий

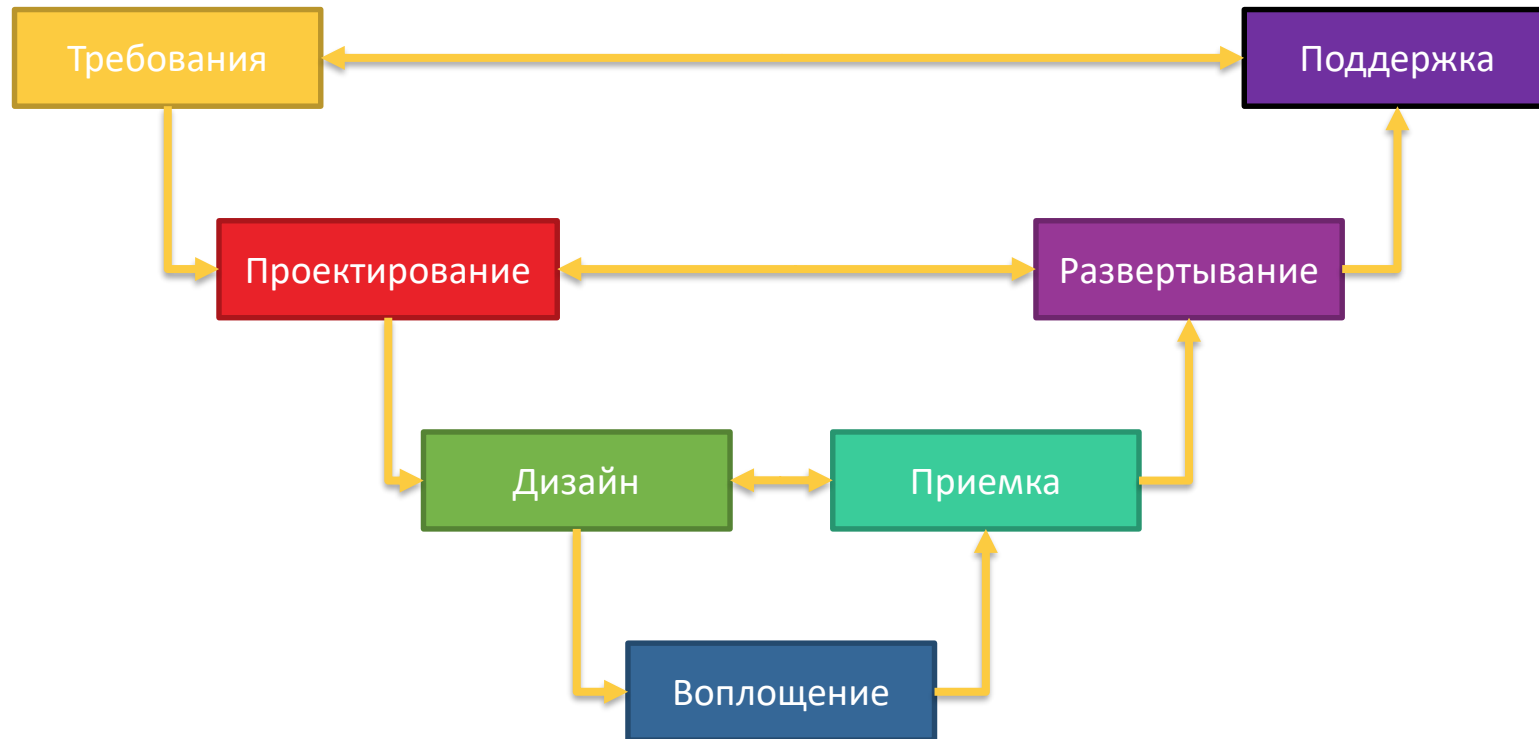
Почему эволюция?

- Каскадная
- V-модель
- Итеративная
- Спиральная
- ...

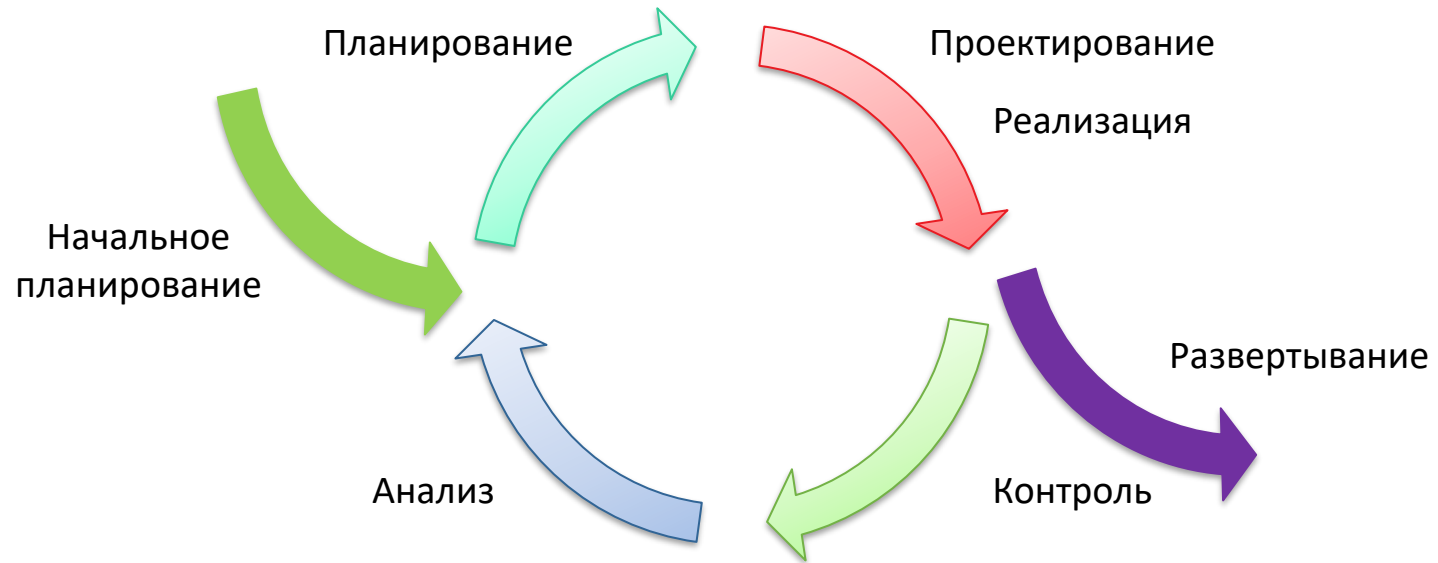
Каскадная модель



V-модель

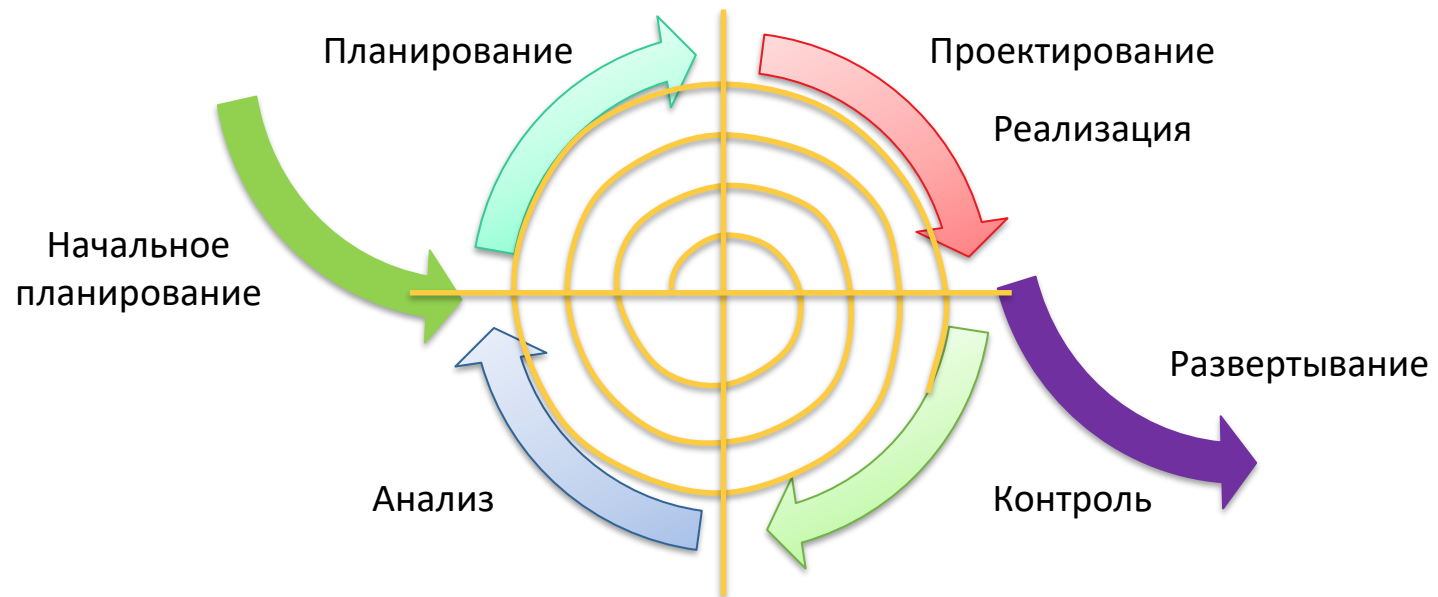


Итеративная модель



Цикл Деминга (PDCA => OODA)

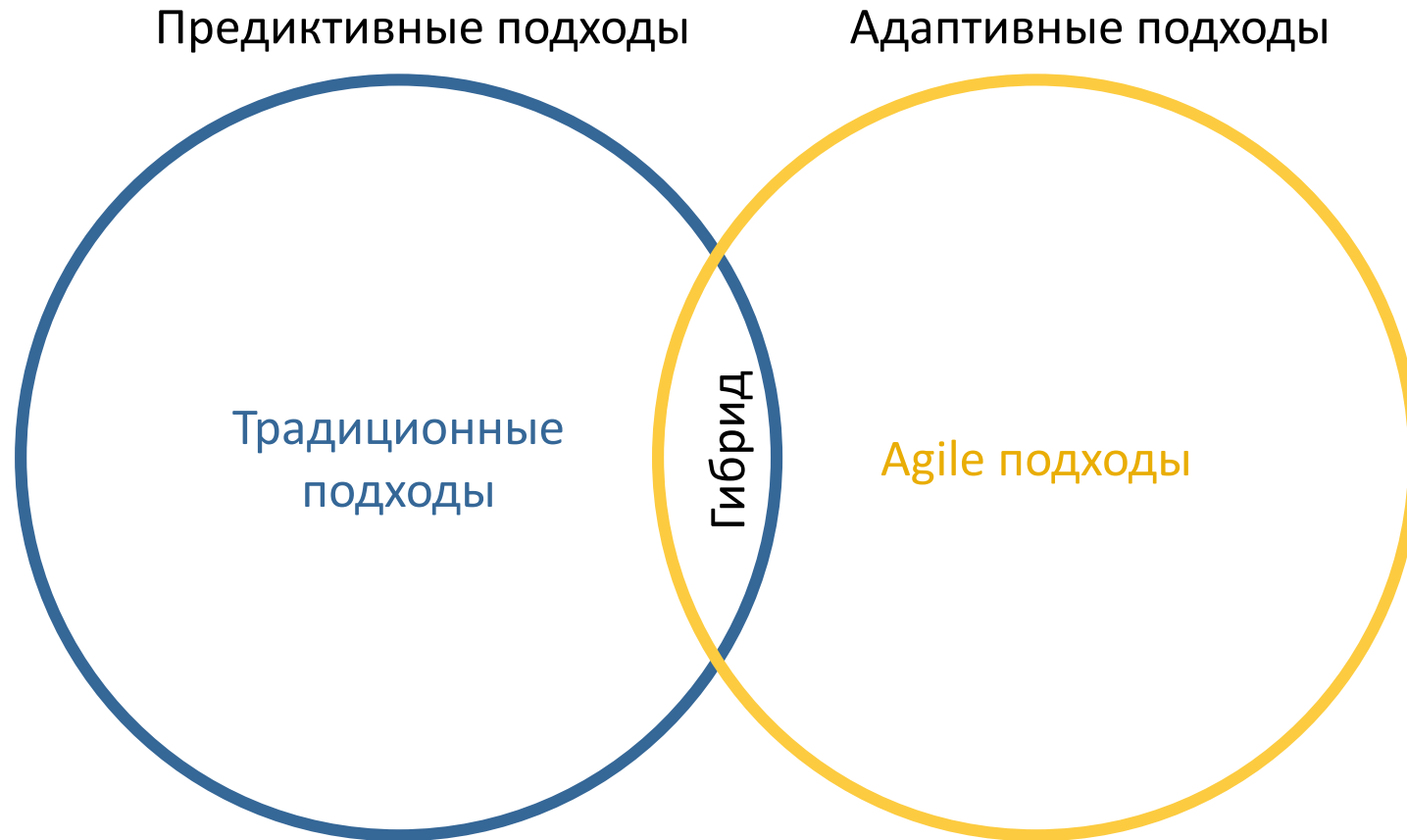
Спиральная модель



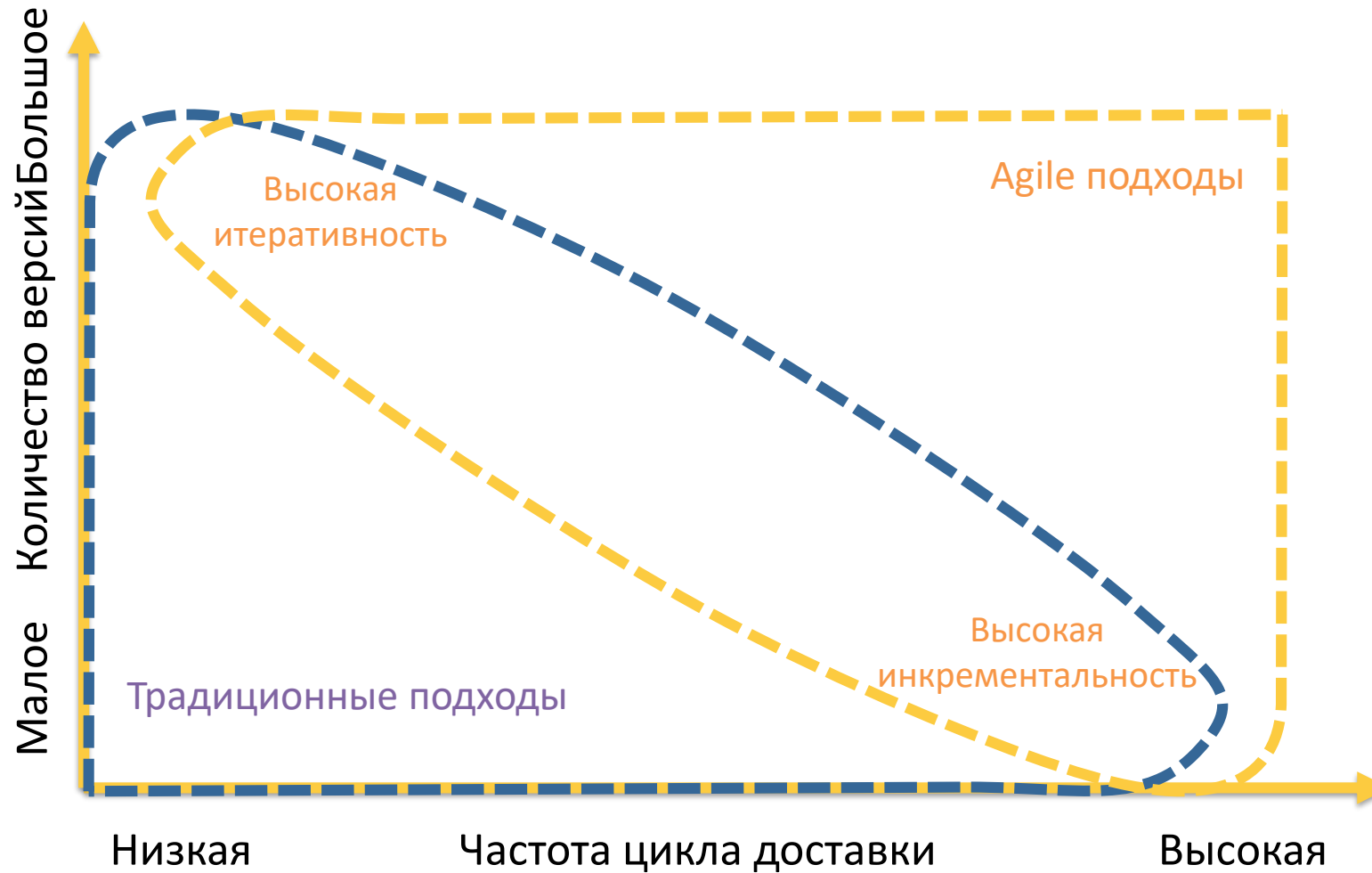
Что в реальности

В реальности идеальных вариантов не бывает

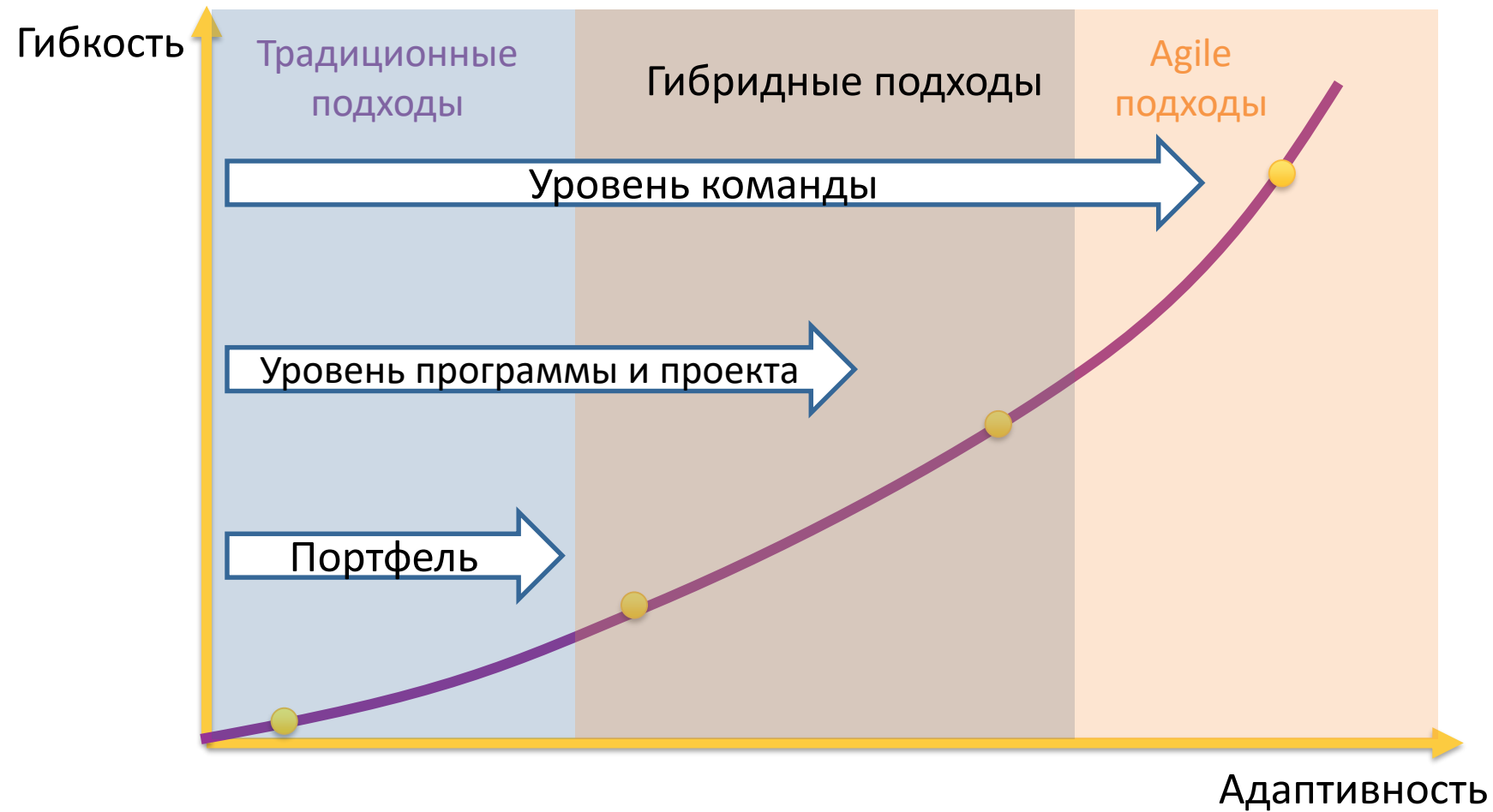
Первое приближение



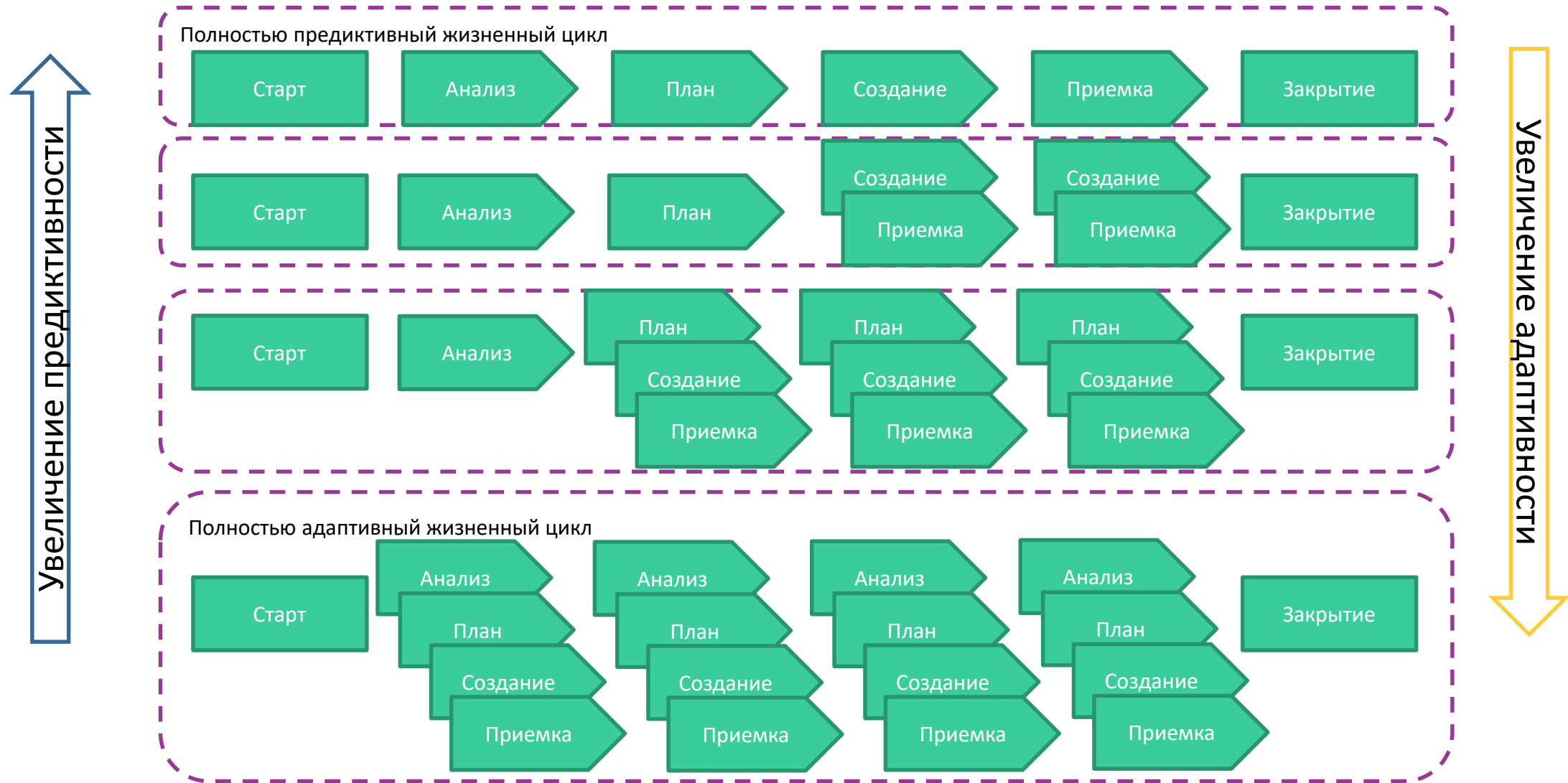
Какова область применения



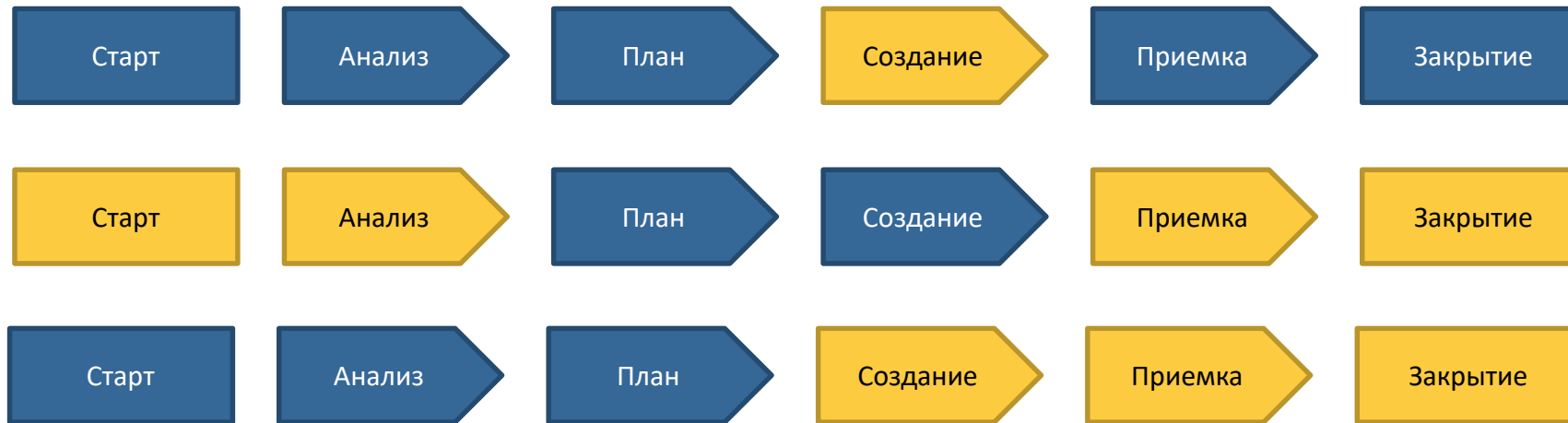
На каком уровне применять (пример)



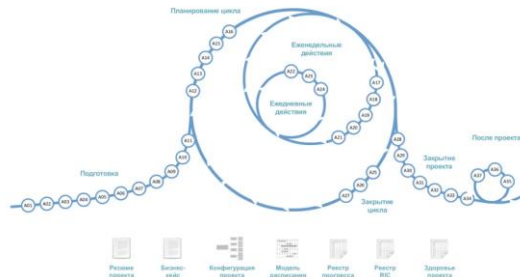
Генерация подходов (общая схема)



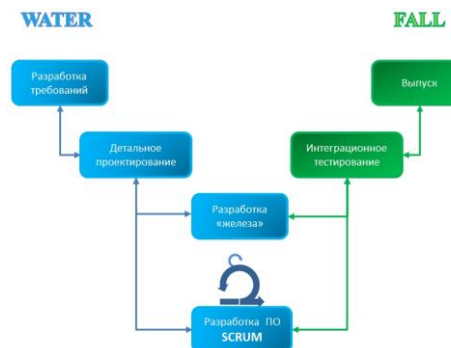
Генерация подходов (как еще может быть)



P3Express

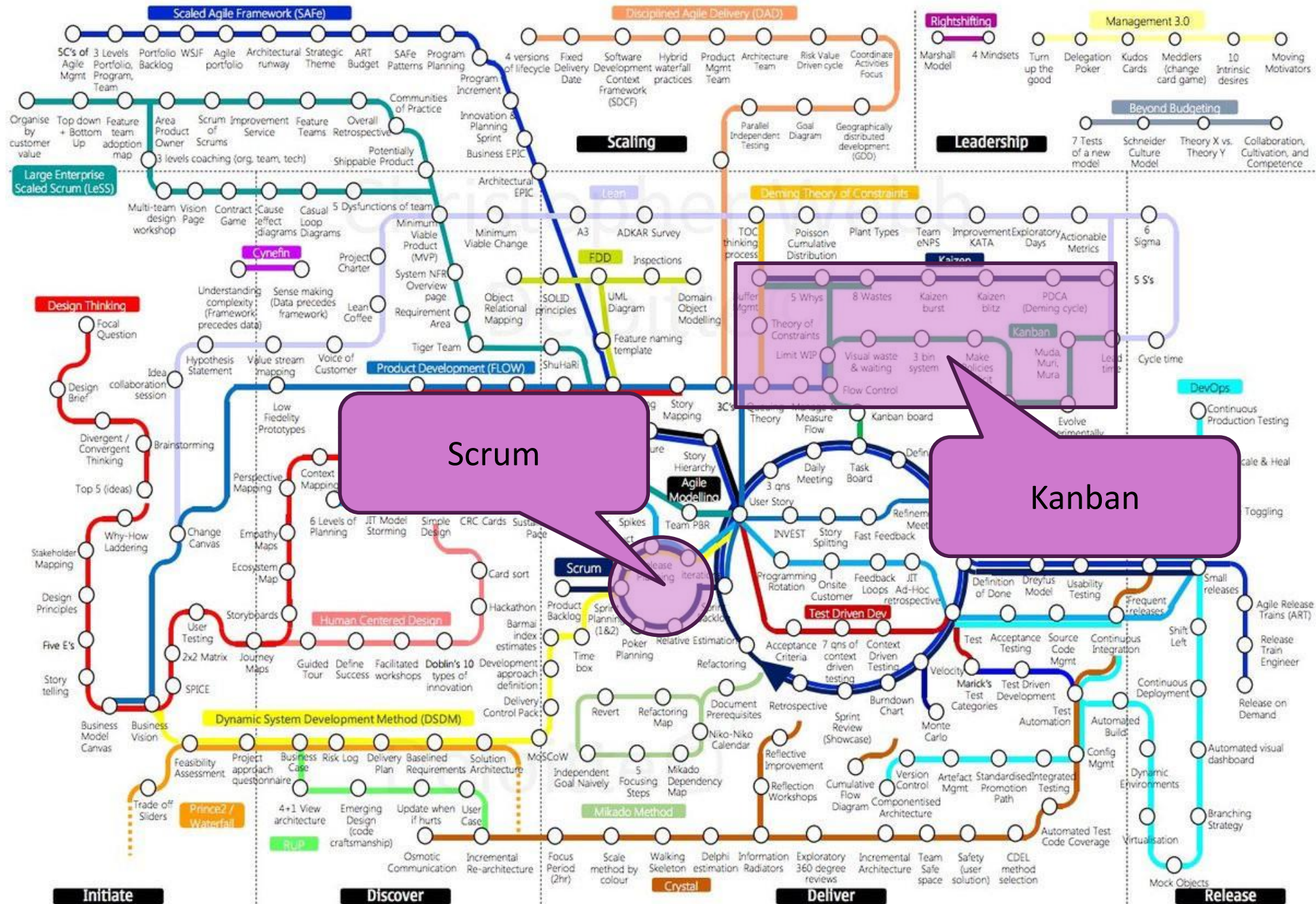


Water scrum fall



Параллельное применение

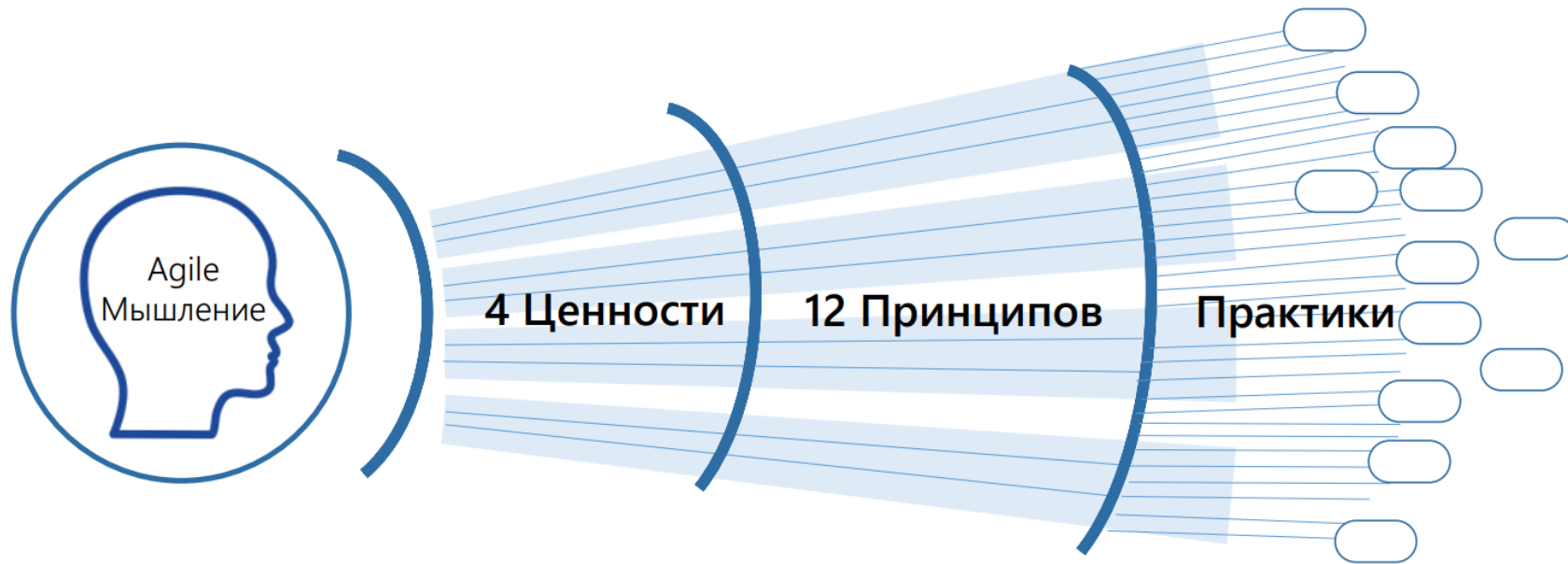
Agile	Agile	Agile
Предиктивный	Предиктивный	Предиктивный



Agile – это образ мышления

<http://agilemanifesto.org/iso/ru/manifesto.html>

<http://agilemanifesto.org/iso/ru/principles.html>



Agile-манифест (2001 г.)

Мы постоянно открываем для себя более совершенные методы разработки программного обеспечения, занимаясь разработкой непосредственно и помогая в этом другим. Благодаря проделанной работе мы смогли осознать, что:

Люди и взаимодействие важнее **процессов и инструментов**
Работающий продукт важнее **исчерпывающей документации**
Сотрудничество с заказчиком важнее **согласования условий контракта**
Готовность к изменениям важнее **следования первоначальному плану**

То есть, не отрицая важности того, что справа, мы всё-таки больше ценим то, что слева.

Принципы Agile

- Наивысшим приоритетом для нас является удовлетворение потребностей заказчика, благодаря регулярной и ранней поставке ценного программного обеспечения.
- Изменение требований приветствуется, даже на поздних стадиях разработки. Agile-процессы позволяют использовать изменения для обеспечения заказчику конкурентного преимущества.
- Работающий продукт следует выпускать как можно чаще, с периодичностью от пары недель до пары месяцев.

Принципы Agile

- На протяжении всего проекта разработчики и представители бизнеса должны ежедневно работать вместе.
- Над проектом должны работать мотивированные профессионалы. Чтобы работа была сделана, создайте условия, обеспечьте поддержку и полностью доверьтесь им.
- Непосредственное общение является наиболее практичным и эффективным способом обмена информацией как с самой командой, так и внутри команды.

Принципы Agile

- Работающий продукт — основной показатель прогресса.
- Инвесторы, разработчики и пользователи должны иметь возможность поддерживать постоянный ритм бесконечно. Agile помогает наладить такой устойчивый процесс разработки.
- Постоянное внимание к техническому совершенству и качеству проектирования повышает гибкость проекта.

Принципы Agile

- Простота — искусство минимизации лишней работы — крайне необходима.
- Самые лучшие требования, архитектурные и технические решения рождаются у самоорганизующихся команд.
- Команда должна систематически анализировать возможные способы улучшения эффективности и соответственно корректировать стиль своей работы.

Не Agile

- Если мы говорим о процессах, документах, встречах, организационных моментах, то это **НЕ** Agile
- Если мы говорим о людях, о фильтрах информации на вход и выход, об «информационном радиаторе», о «тихой гавани», о заказчике, то это Agile

Традиционный и гибкий подходы

	Традиционный подход	Гибкий подход
Метафора	Эстафета	Регби
Фазы	Выполняются последовательно	Перекрываются
Планирование	Детальное	Гибкое
Вовлечение заказчика	Частичное, преимущественно на этапе концептуализации	Полное, в течение всего проекта
Процессы	Стандартизованы	Постоянно пересматриваются
Роли	Четко определенные роли и ответственность	Гибкие роли, делегирование ответственности
Документация	Полная	По необходимости
Обеспечение качества	Контроль для исправления ошибок	Сбор метрик для улучшения процессов



Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Тема 3. Организационная гибкость

Организационная гибкость (Organizational Agility)

Организационная гибкость – способность организации выживать и развиваться в условиях постоянных и непредсказуемых изменений

Потребность в организационной гибкости

- Интенсификация глобальной конкуренции (организация конкурирует с большим числом продуктов)
- Сокращение сроков разработки и жизненного цикла продуктов (требуются иные структуры, ориентированные на потребителя, инновации, быстрое реагирование, высокое качество при более низких затратах)
- Диверсификация спроса (означает не только больший выбор, но и то, что потребители ожидают получить именно то, что они хотят)
- Новые технологии не являются главным двигателем создания конкурентного преимущества. Фокус на новых технологиях, понимании новых технологий и правильном использовании новых технологий.

Признаки гибкой организации

- Постоянно производит измерения эффективности работы персонала, а также ценности продуктов и услуг
- Реагирует на постоянные изменения в потребностях клиентов
- Обучается новому
- Готова реагировать на внезапные изменения и события
- Использует передовые технологии
- Использует новые возможности для улучшения прибыльности и производительности

Составляющие организационной гибкости

- Продукты
- Процессы
- Люди
- Операции
- Организационная структура

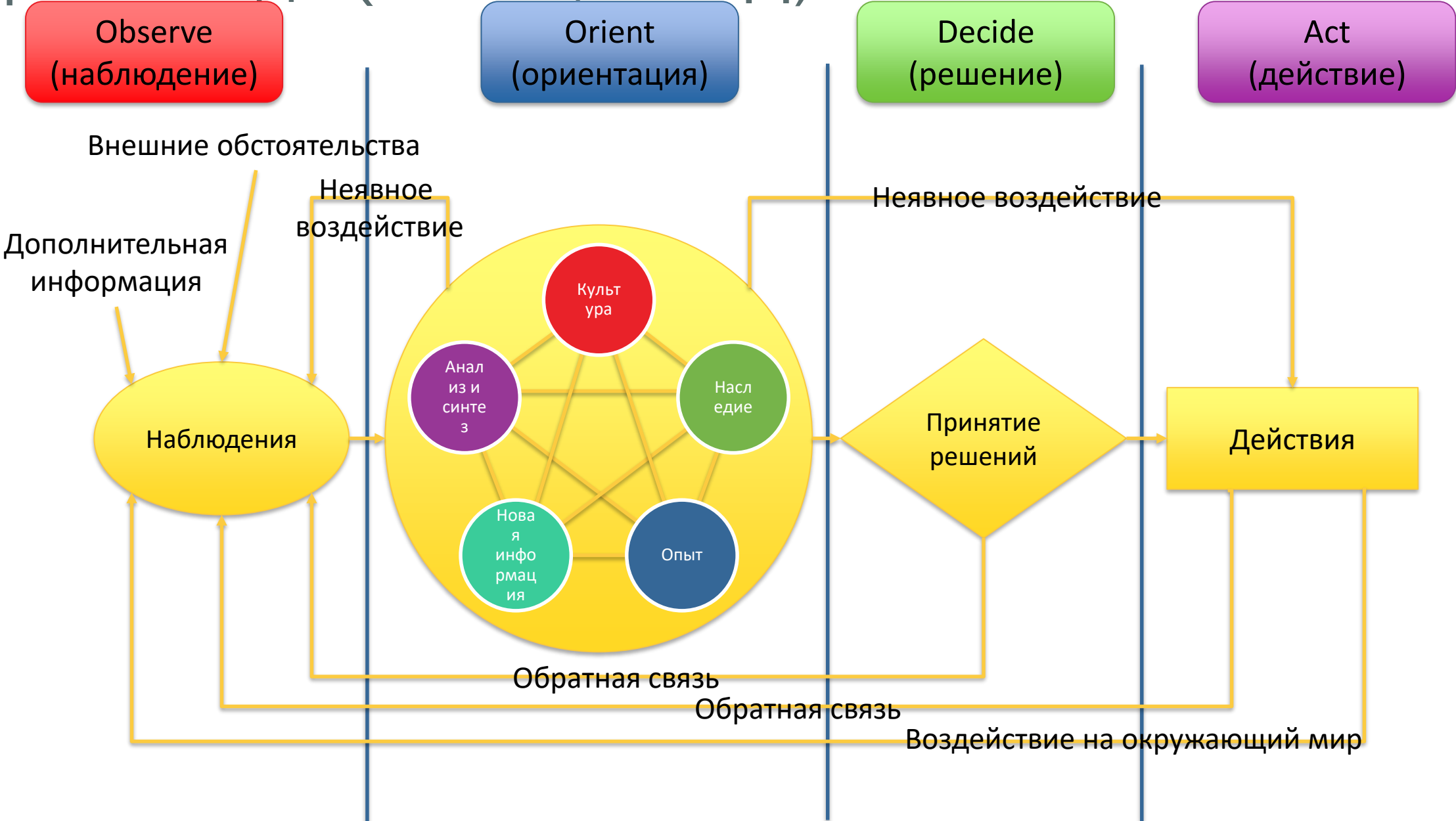
Гибкость продуктов

- Конфигурируемые продукты (возможность производить максимальное число конфигураций при минимальном числе различных компонентов)
- Приспосабливаемые продукты (возможность использовать альтернативные компоненты при возникновении проблем с производством или поставками)

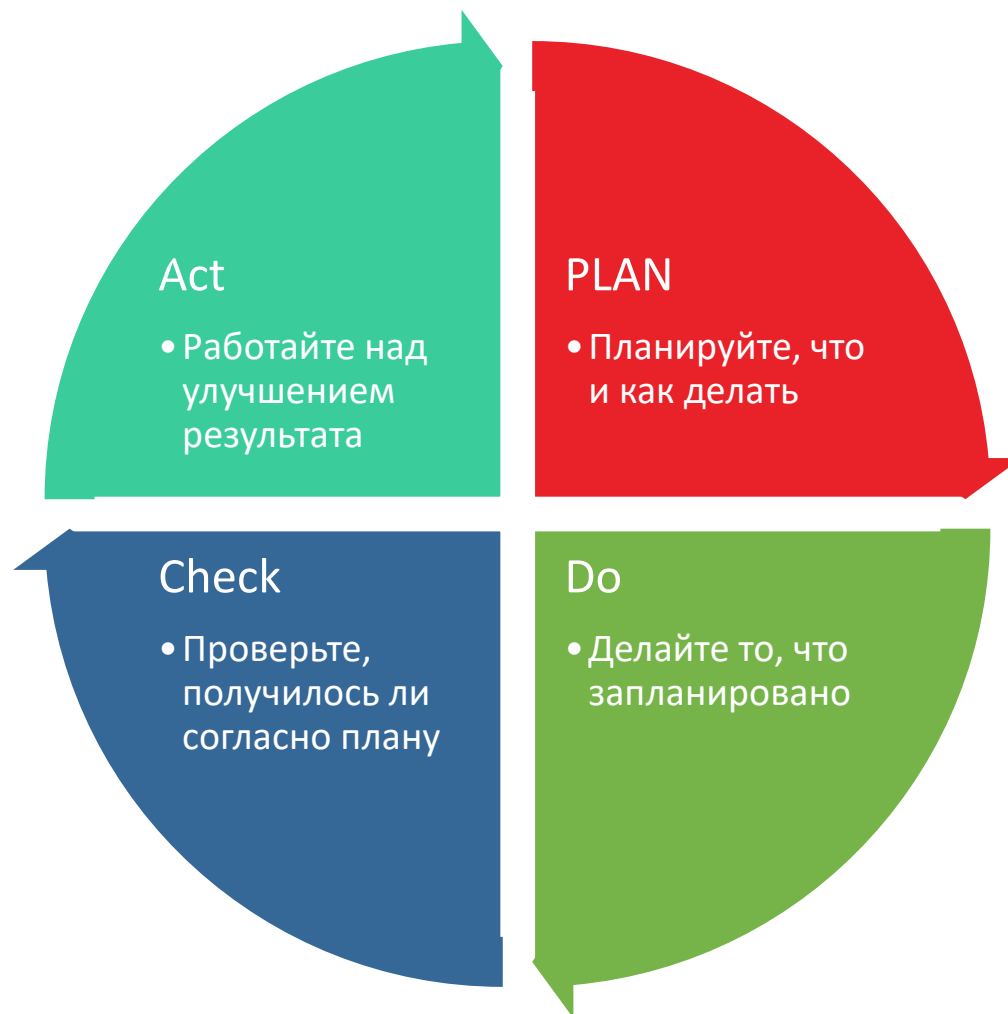
Гибкость процессов

- Разнообразие процессов
- Устойчивость процессов
- Инновации в процессах
- Возможность адаптации процессов

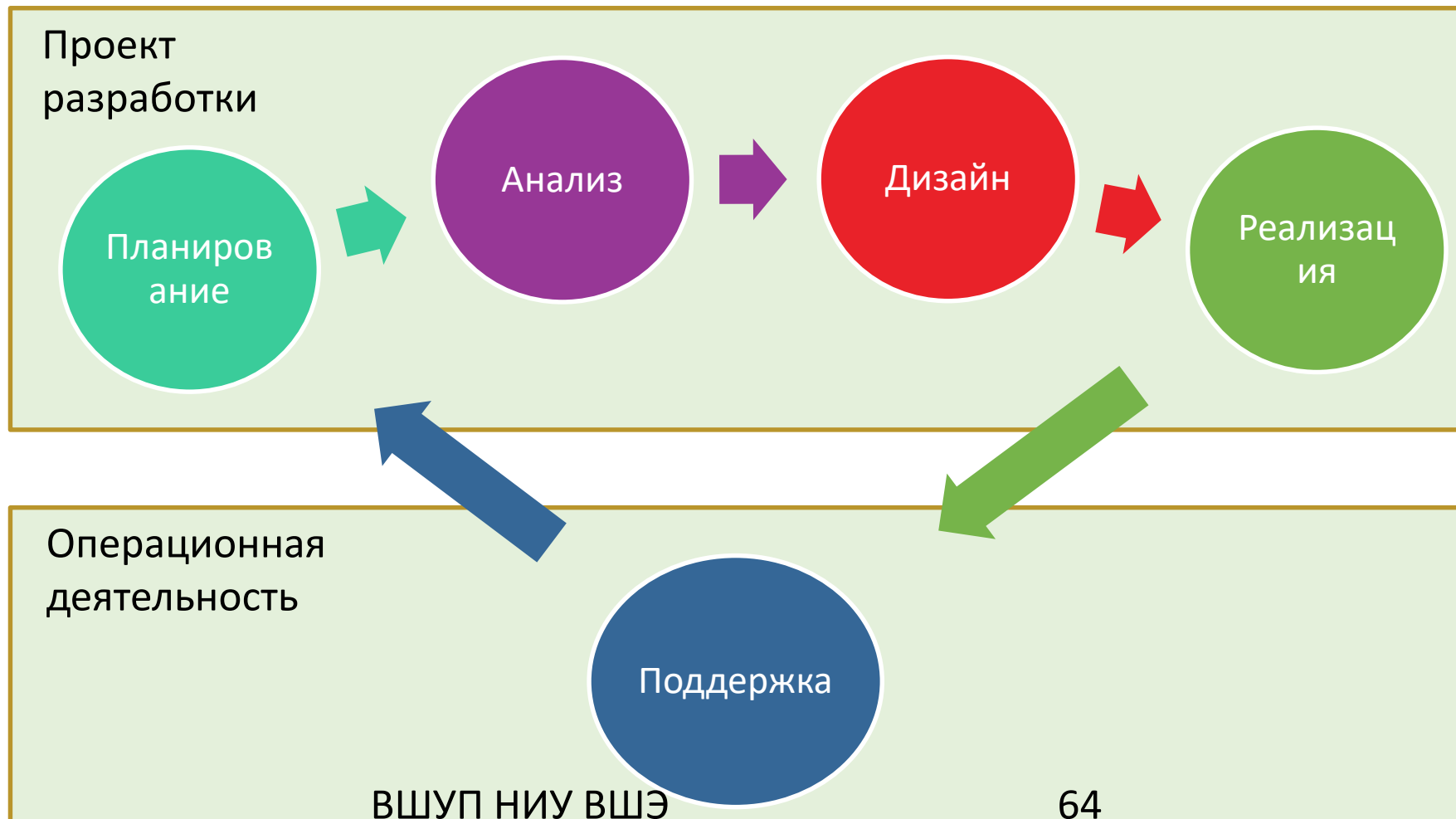
Цикл Бойда (OODA, НОРД)



Цикл Деминга



Жизненный цикл разработки систем (SDLC)



Операционная чувствительность

- Чувствительность технической поддержки
- Чувствительность производства
- Чувствительность закупок
- Чувствительность продвижения новых продуктов

Гибкость человеческих ресурсов

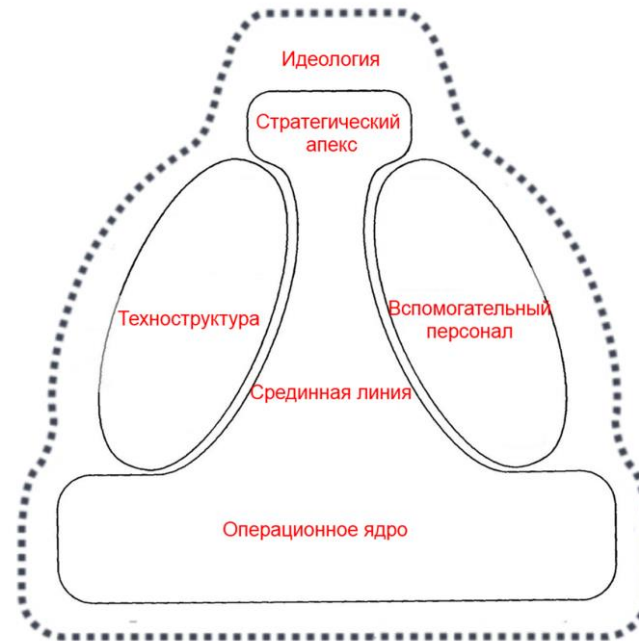
- Персонал с широким спектром навыков
- Готовность к изменениям в зонах ответственности
- Эффективное использование навыков людей

Гибкость организационной структуры

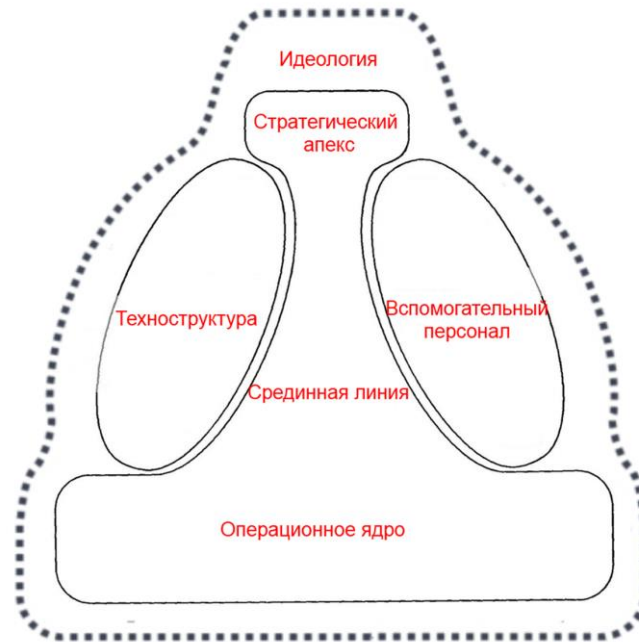
- Стратегическая гибкость
- Эффективность коммуникаций
- Гибкость человеческих ресурсов
- Возможность адаптации структуры

Схема организации

Г. Минцберг, «Структура в кулаке»



Основные типы организационных структур



Как определить, где мы?

Конфигурация	Простая	Механистическая	Профессиональная	Адхократия
Масштаб проекта	Локальный	Локальный	Общекорпоративный	Общекорпоративный
Статус спонсора проекта	Руководитель функционального бизнес подразделения	Руководитель функционального бизнес подразделения	Руководитель из числа первых лиц бизнеса	Руководитель из числа первых лиц бизнеса
Тип лидерства	Один лидер, работа по его прямым поручениям	Работа по четким процедурам	Координация профессионалов с очень высокой квалификацией, профессиональное искусство	Поиск технических решений в условиях жесткой неопределенности
Лояльность	Высокая группе и лидеру, чувствительность с изменениям персонального состава	Низкая, доминируют личные интересы	Профессиональный индивидуализм, приверженность профессиональным стандартам	Командный дух, приверженность целям
Роль менеджера	Политик	Администратор	Профессионал	Партнер-предприниматель
Рядовые сотрудники	Исполнитель-разработчик	Исполнитель-настройщик	Профессионал-аналитик/консультант/архитектор	Команда
Команда	Сплоченная, семья, друзья	Машина	Группа профессионалов	Первопроходцы
Тип коммуникаций	Неформальные	Формальные	Формальные внешние и неформальные внутренние	Неформальные
Передача знаний	Слабая отчуждаемость (давление контекста отношений)	Документированность и формализация	Оппортунизм	Использование и создание НТЗ (хорошая отчуждаемость)

Какие у нас проблемы?

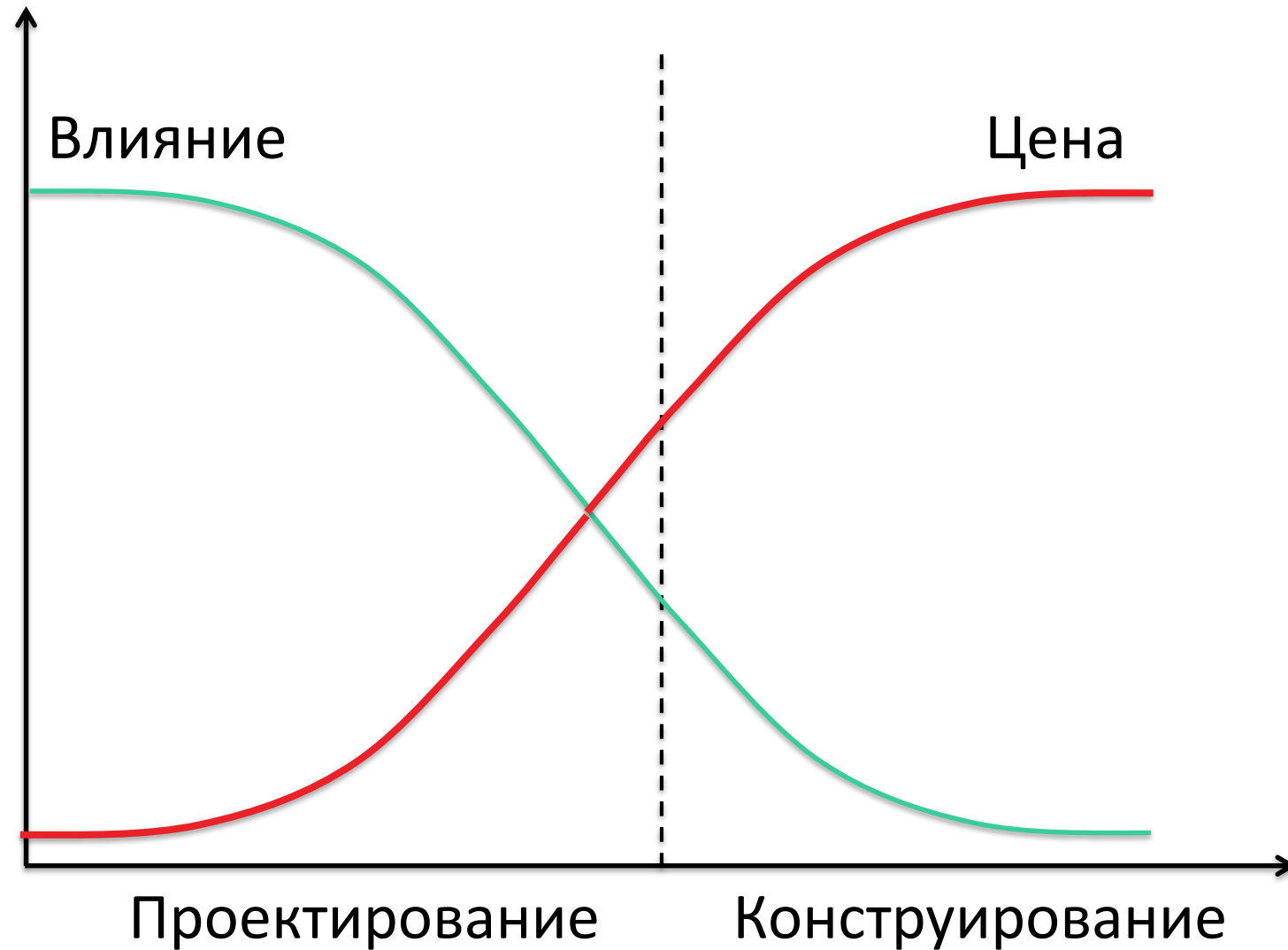
Конфигурация	Простая	Механистическая	Профессиональная	Адхократия
Выявление проблем	Выявление проблем связано с лояльностью	Проблема находит своего адресата путем согласования документов	Проблема находит своего адресата путем согласования документов	Проблема находит своего адресата в результате проявления инициативы
Решение проблем	Решение проблем: прямые поручения лидера и инициатива требует санкции лидера	Решение проблем требует согласования	Решение проблем требует согласования	Решение проблем требует согласования, но оно может быть и неформальным
Узкое место	Перенапряжение лидера	Формализованность отношений	“Навешивание ярлыков”	Большой объем коммуникаций
Отличительная проблема	Примадонны	Медленные решения	Медленные решения	Большой объем коммуникаций
Проблемы информации	Информационная непрозрачность	Горизонтальные связи нелегитимны	Только через “профессиональные центры”	“Изобретение велосипеда”
Пути инноваций	Инновации приходят через лидера	Инновации приходят через техноструктуру	Инновации приходят извне (профессиональное сообщество)	Инновации рождаются в операционном ядре
Отношение к оценке эффекта	У обеих сторон нет стимулов к поиску реальных бизнес эффектов	Оценка снижения затрат	В лучшем случае сравнения с метриками образцов “лучшей практики”	Сильны стимулы к поиску реальных бизнес эффектов

Что необходимо понимать?

- Например, если мы оказались в простой организации, то необходимо не только уговорить ключевых сотрудников, но и получить санкцию лидера на проведение тех или иных изменений без его непосредственного участия, иначе его можно просто перегреть.
- Другими словами, нужно понимать, где мы находимся, чтобы решать действительно имеющиеся проблемы, и не бороться с несуществующими.

Что еще важно?

Стоимость и сила изменений



Почему маленькие команды?

Размер команды	Число коммуникационных связей
3	3
4	6
5	10
6	15
10	45

В больших командах появляется иерархия, усложняется процесс принятия решений, теряется гибкость

Расширение гибких практик для крупных компаний

- Оценка готовности (Readiness Assessment)
- Проектное сообщество (Project Community)
- Разработка устава проекта (Project Chartering)
- Менеджмент через тестирование (Test driven management)
- Ретроспективы (Retrospectives)
- Непрерывное обучение (Continuous learning)

Оценка готовности

- Готовность инфраструктуры
- Доступность экспертов
- Приверженность непрерывному улучшению,
- Организационная культура и организационная структура, поддерживающая изменения и не устанавливающая барьеров для изменений
- Готовность людей

Проектное сообщество

- Заинтересованные стороны проекта
- Активные люди в центре и на периферии разработки, которые могут оказать на нее влияние
- Проектное сообщество всегда больше, чем вы думаете

Разработка устава проекта

- Видение
- Миссия
- Проектное сообщество
- Ценности
- Критерии успеха
- Ключевые события
- Предварительные соглашения

Менеджмент через тестирование

- Определение критериев успеха при инициировании проекта
- Тестируемые критерии успеха должны находиться за границами проекта
- Следует тестировать достижение целей, а не способы их достижения

Ретроспективы

- Цель – непрерывное улучшение
- Проводятся в конце каждой итерации
- Рассматриваются проблемы, события, уроки

Непрерывное обучение

- Получение новых навыков, изучение новых техник, инструментов
- Фокусировка групп и индивидов на развитии технических и нетехнических компетенций, необходимых в работе
- Тренинговые сессии каждые 1-2 недели
- Плохая альтернатива – RBD (Resume Based Development)

Восемь принципов менеджмента качества

1. Ориентация на потребителя
2. Лидерство руководства
3. Вовлечение работников
4. Процессный подход
5. Системный подход к менеджменту
6. Постоянное улучшение
7. Принятие решений, основанное на фактах
8. Взаимовыгодные отношения с поставщиками

Способы применения

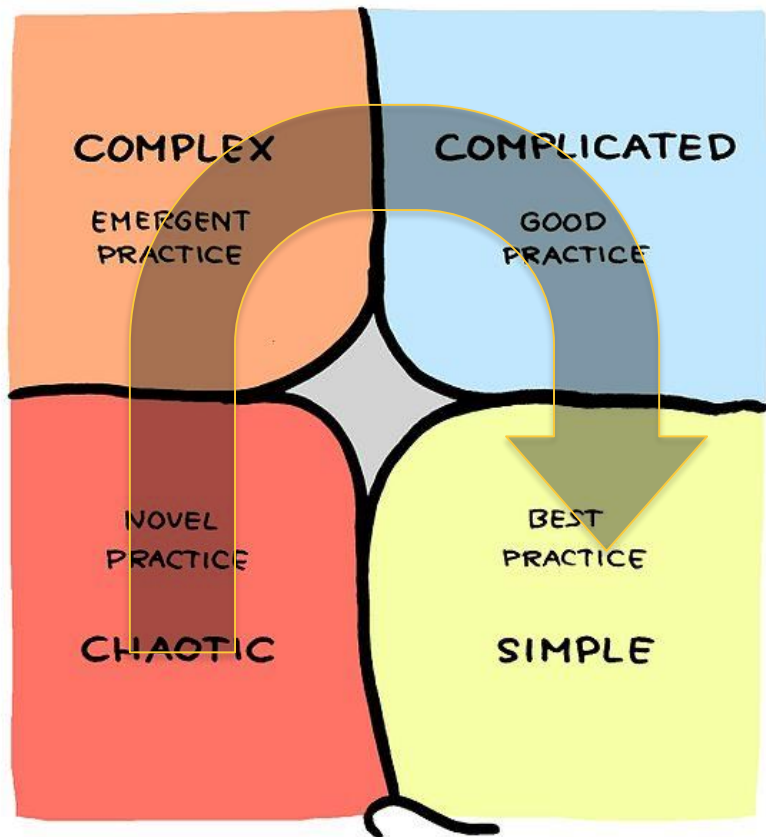
- Agile внутри: заказчик не имеет представления, не может или не хочет работать по гибким методологиям. В этом случае команда может внутри работать по гибким методологиям, а с заказчиком использовать традиционный подход.
- Agile снаружи: заказчик может, хочет работать по гибким методологиям. В этом случае заказчик вовлекается в работу команды проекта

Удобство использования

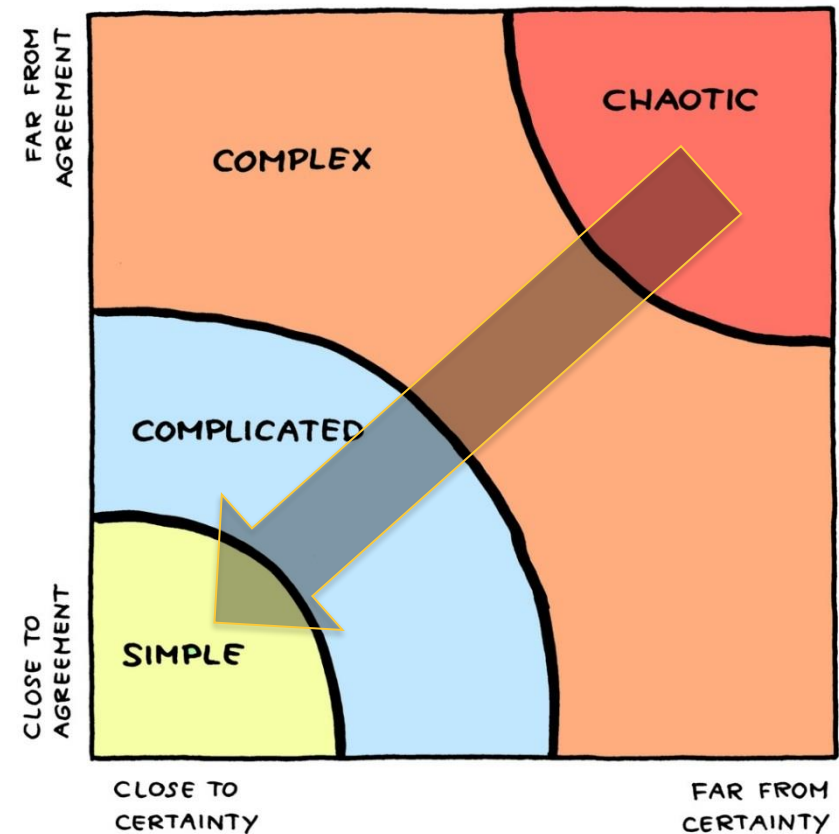
Характеристики проекта	Традиционный подход	Agile внутри	Agile снаружи
Ограничения по цене, времени, требованиям	++	0	-
Внешний заказчик, ограничения по времени	0	++	+
Внутренний заказчик, ограничения по времени	-	+	++

Модель сложности проектов

Cynefin framework

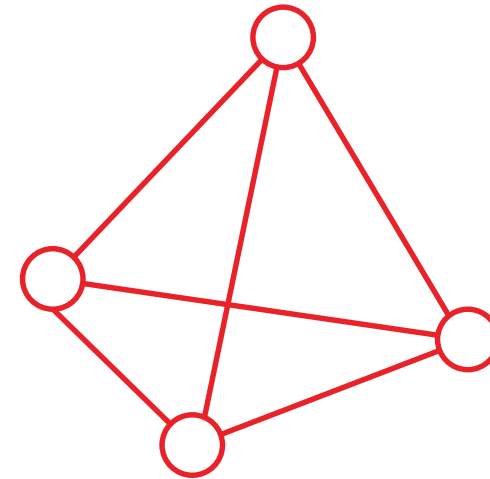


The Stacey complexity model



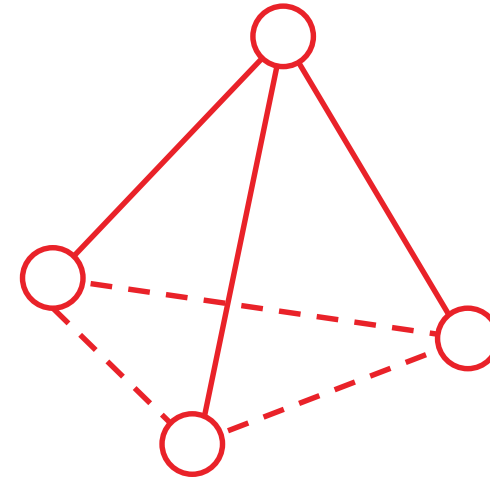
Простые проекты

- Внутреннее устройство известно
- Все причинно-следственные связи осознаны, предсказуемы и повторяемы
- Стандартная операционная деятельность, лучшие практики
- Осмысление -> категоризация -> ответ



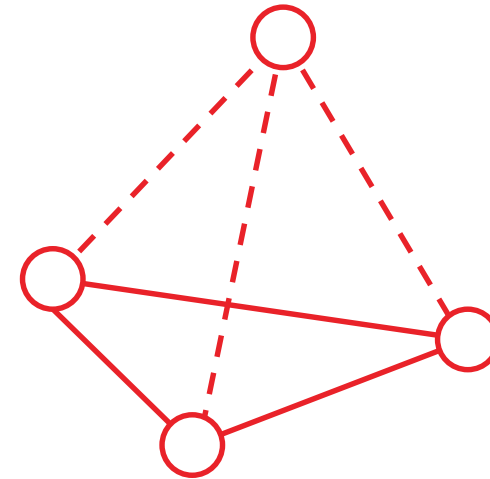
Сложные проекты

- Внутренне устройство потенциально известно
- Все причинно-следственные связи разделены во времени и пространстве
- Экспертное принятие решений, сценарное планирование, системное мышление
- Осмысление -> анализ -> ответ



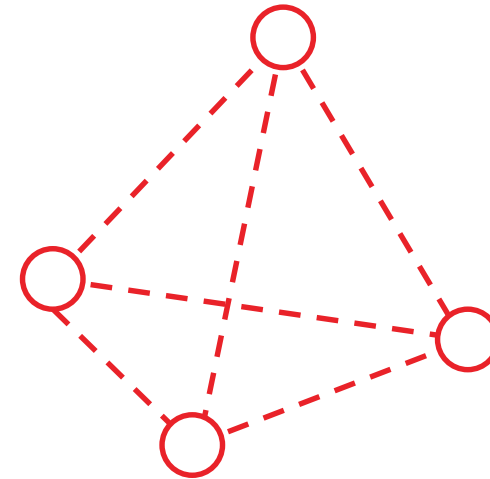
Комплексные проекты

- Ретроспективно когерентно (прослеживаются отдельные зависимости)
- Причинно-следственные связи не повторяемы
- Работа по шаблонам, множественные повторы
- Исследование -> осмысление -> ответ



Хаотические проекты

- Ретроспективно некогерентно
- Причинно-следственные связи невозможно постичь
- Действия, направленные на стабилизацию, кризис-менеджмент
- Действие -> осмысление -> ответ



Стили управления (теория Бланшара)

Поддерживающий (S3)	Наставнический (S2)
Делегирующий (S4)	Директивный (S1)

- D1 - жесткая постановка целей, сроки исполнения, ориентация на результат.
- D2 - обоснование и обсуждение идей, активная работа с командой, решения все равно принимает руководитель.
- D3 - члены команды включаются в организацию рабочего процесса. Руководитель поддерживает подчиненных, но основные технические решения принимают они.
- D4 - принятие решение и ответственность за результат на команде

Зрелость сотрудников (теория Бланшара)

Не мотивирован Профессионален (D3)	Не мотивирован Не профессионален (D2)
Мотивирован Профессионален (D4)	Мотивирован Не профессионален (D1)

- D1 – новичок, которого необходимо вводить в курс дела.
- D2 – либо тот же новичок, который столкнулся с большим количеством проблем, либо профессионал, переброшенный в незнакомую ему предметную область.
- D3 – таким человеком может быть тот, кто заскучал от однообразных задач, либо столкнулся с проблемой, которую не может решить на своем уровне.
- D4 – мечта любого руководителя проекта.

Соответствие стилей и зрелости

- Очевидным является то, что стили управления и уровни мотивации и профессионализма у членов команды и команды, в целом, должны соответствовать друг другу.
- В стилях S1-M1 и S4-M4 в первую очередь важен результат и процесс, поэтому может помочь работа по kanban.
- В стилях S2-M2 и S3-M3 важна работа с людьми, то есть встречи, тут может больше помочь scrum.

Игра: One piece flow simulation ч.1

- 20 монет
 - 4 работника
 - Измеряем время работы каждого работника
 - Измеряем время поступления первой партии на рынок
 - Измеряем общее время работы
- Все 20 монет
 - По 10 монет
 - По 5 монет
 - По 1 монете

Игра: One piece flow simulation ч.2

- Один сотрудник
- 5 менеджеров
- Сотрудник должен написать имя каждого
- Измеряем время написания первого имени
- Измеряем время все работы
- Все действуют наперебой
- Сотрудник пишет по одной букве каждого имени
- Сотрудник по очереди пишет полностью имя каждого



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Тема 4. Scrum

Процессы должны быть!



SCRUM





**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Основные определения

Определение

Скрам – это фреймворк*

*Фреймворк – это набор базовых элементов и правил – своего рода каркас, на котором строится процесс создания решения или продукта.

Основные определения

- Definition of done
- Итерация
- Бэклог

Основные компоненты Scrum-процесса

- Роли
- Документы
- Встречи



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Роли

Свинья идет по дороге. Курица смотрит на нее и говорит: «А давай откроем ресторан!» Свинья смотрит на курицу и отвечает: «Хорошая идея, и как ты хочешь его назвать?» Курица думает и говорит: «Почему бы не назвать „Яичница с беконом“?». «Так не пойдет, — отвечает свинья, — ведь тогда мне придется полностью посвятить себя проекту, а ты будешь вовлечена только частично».

- «СВИНЬИ»

- Скрам мастер (Scrum Master)

- Владелец продукта (Product Owner)

- Скрам-команда (Scrum Team)

- «КУРЫ»

- Пользователи (Users)

- Заинтересованные стороны (Stakeholders)

- Руководители (Managers)

- Эксперты (Consulting Experts)



Scrum Master

- Отвечает за соблюдение практик и процессов
 - Проводит Daily Scrum
 - Ведет Burndown Chart
 - Является модератором собраний
 - Организует коммуникации и свободный обмен информацией
-
- ScrumMaster – это «владелец процесса»



*Кен Бланшар в своей книге “Лидерство к вершинам успеха” ввел термин “лидер-слуга”. Имеется ввиду, что Скрам-Мастер не привилегированный носитель власти, как это происходит в случае с классическим менеджментом, а лидер, который вовлекает участников Скрам-команды в процесс работы, не имея формальной власти.

Product Owner (владелец продукта)

- Формирует видение продукта и принимает решения по продукту
- Определяет приоритеты пользовательских историй
- Разрешает спорные ситуации
- Отвечает за приемку продукта а конце каждой итерации
- Активный член команды, участвует в планировании итераций, ежедневных скрамах и ретроспективах
- Предпочтительно, если находится в одном помещении с командой



Для успешного выполнения обязанностей Владелец Продукта все члены организации должны уважать его решения. Отражением его решений служит содержание и порядок элементов Бэклога Продукта.

Роль руководителя проекта



- решение вопросов размещения проектной команды
- обучение и наставничество проектной команды
- наполнение бэклога продукта вместе с РО
- создание бэклога спринта с проектной командой
- оценка стоимости, бюджетирование и контроль расходования средств, выделенных на проект
- планирование коммуникаций
- управление рисками
- управление закупками

Типовые проблемы с заказчиком

- Владельца продукта невозможно вовлечь в активное участие в проекте
- Заказчик требует пожертвовать качеством



Скрам-Команда (Scrum Team)



Скрам-команда состоит из Владельца Продукта, Команды Разработки и Скрам-мастера.

- ✓ Скрам-команды являются самоорганизующимися и кроссфункциональными.
 - Самоорганизующиеся команды самостоятельно решают, как выполнять свою работу – без внешних указаний.
 - Компетенций кроссфункциональной команды достаточно для выполнения полного объема работ, в т.ч. весь комплекс работ в сочетании с оптимальным способом ее исполнения.
- ✓ Исполнение всех работ предполагает предварительную договоренность команды и представителей бизнеса.
- ✓ Очередная версия готового продукта должна быть доступна в любой момент времени.

Идеальная Скрам-Команда

имеет такие характеристики:

- ✓ **Самоорганизация.** Команда самостоятельно решает, как превратить Бэклог Продукта в Инкремент Продукта – без каких-либо внешних указаний (даже от Скрам-мастера).
- ✓ **Кроссфункциональность.** При этом отдельные члены команды могут обладать различными специализированными навыками и экспертизой.
- ✓ Если мы говорим об IT. "Scrum recognizes no **titles for Development Team members**, regardless of the work being performed by the person." **VS** " **Разработчик – единственная роль** в Команде Разработки, невзирая на тип выполняемых задач." **Правило без исключений.**
- ✓ Отсутствие подкоманд в Команде Разработки. **Правило без исключений.**
- ✓ Коллективная ответственность Команды Разработки за создание Инкремента Продукта наивысшего качества.

Размер Команды Разработки

Не более 9

При подсчете численности Команды Разработки, Владелец Продукта и Скрам-мастер не учитываются, если они сами не выполняют работу из Бэклога Спринта.



Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Встречи

- Инициирование проекта (стартовая колода)
- Собрание планирования итерации (Sprint planning meeting)
- Ежедневное собрание (Daily Scrum)
- Обзор итерации (Sprint Review)
- Ретроспектива итерации (Sprint Retrospective)



Концептуализация и инициирование проекта (стартовая колода)

- Для чего мы собрались?
- Блиц-резюме продукта
- Разработка оформления продукта
- Список того, что мы не собираемся делать
- Встреча с коллегами
- Демонстрация решения
- Что не дает нам покоя?
- Определяемся с длительностью проекта
- Что необходимо сделать?
- Что для этого понадобится?

Разработка стартовой колоды

- На разработку уходит от 2 дней до 2 недель (равноценно шести месяцам по планированию работ)
- Должна пересматриваться при внесении всех крупных изменений в суть или направление движения

Блицрезюме продукта

Для [адресат сообщения]

Который [утверждение о необходимости или представляющейся возможности]

Это [название продукта]

Относится к [категория продукта]

И при этом [основная выгода, убедительная причина приобрести]

В отличие от [основное конкурентное предложение]

Наш продукт [объяснение основного конкурентного отличия]

Возможные параметры пользовательских историй

- Идентификатор
- Название
- Важность
- Предварительная оценка (в чем?)
- Как продемонстрировать
- Категория
- Компоненты, которые затрагивает
- Инициатор запроса
- Примечание

Спринт (The Sprint)

Не более **1** месяца

Спринт состоит из

- Планирования Спринта (Sprint Backlog определяется в начале итерации и не меняется до конца итерации),
- Ежедневного Скрама,
- Разработки (нет историй, выполненных на 10, 50, 95%. История или выполнена или нет),
- Обзора Спринта (в конце спринта – рабочий продукт),
- Ретроспективы Спринта.

Планирование Спринта (Sprint Planning)

8

Не более часов

По результатам Планирования Спринта Скрам-команда получает ответы на вопросы:

- ✓ Цель Спринта* (+ Definition of "Done")
- ✓ Каким будет Инкремент в конце Спринта?
- ✓ Как организовать работу, чтобы получить готовый Инкремент Продукта?
- ✓ Sprint backlog
- ✓ Дата демонстрации
- ✓ Место и время проведения ежедневных встреч

Оценки историй

Основная цель оценки при разработке программ – не предсказать результат проекта, а определить, являются ли цели достаточно реалистичными, чтобы их можно было достичь, контролируя проект (С. Макконнелл, Software Estimation: Demystifying the Black Art)

Методы оценки:

- Используйте относительные оценки
- Триангуляция (три эталонные истории: маленькая, средняя, большая)
- Покер-планирование

Ежедневный Скрам (Daily Scrum)

Не более **15** минут

Каждый участник Команды Разработки отвечает на следующие вопросы:

- ✓ Что я сделал с момента прошлой встречи для того, **чтобы помочь Команде Разработки достичь Цели Спринта?**
- ✓ Что я собираюсь сделать сегодня для того, **чтобы помочь Команде Разработки достичь Цели Спринта?**
- ✓ Какие препятствия замедляют достижение Цели Спринта – для меня и **Команды Разработки?**

Обзор Спринта (Sprint Review)

Не более **4** часов

Ключевые элементы:

- ✓ Участники: Скрам-команда и заинтересованные лица.
- ✓ Владелец Продукта объясняет, что готово и что нет.
- ✓ Команда Разработки обсуждает то, что получилось во время Спринта, какие были проблемы и как эти проблемы были решены.
- ✓ Команда Разработки демонстрирует готовую работу и отвечает на вопросы касательно Инкремента.
- ✓ Владелец Продукта обсуждает Бэклог Продукта.
- ✓ **Все присутствующие** обсуждают то, над чем стоит работать дальше.
- ✓ **Все присутствующие** обсуждают изменения на рынке и их потенциальное влияние на продукт. Определяются следующие действия.

Ретроспектива Спринта (Sprint Retrospective)

Не более **3** часов

Цели Ретроспективы Спринта:

- ✓ Инспекция степени успешности Спринта относительно людей, взаимоотношений между ними, процессов и инструментов.
- ✓ Обнаружение и упорядочение того, что прошло хорошо и того, что нуждается в улучшении.
- ✓ Создание плана внедрения улучшений в процесс работы Скрам-команды.



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Документы

- Критерии готовности (Definition of Done)
- Список пользовательских историй продукта (Product Backlog)
- Список пользовательских историй итерации (Sprint Backlog)
- Обратный график реализации историй (Burndown Chart)

Definition of Done

- Definition of Done («Критерии Готовности», DoD) — это когда все условия или Acceptance Criteria, которым должен соответствовать продукт, выполнены («Done») и готовы к принятию пользователем, клиентом, командой или потребляющей системой.
- Не путаем с критериями приемки. В них требования относятся к конкретным элементам продукта со стороны клиента, а в DoD они относятся к продукту или его инкременту со стороны команды.

Product Backlog

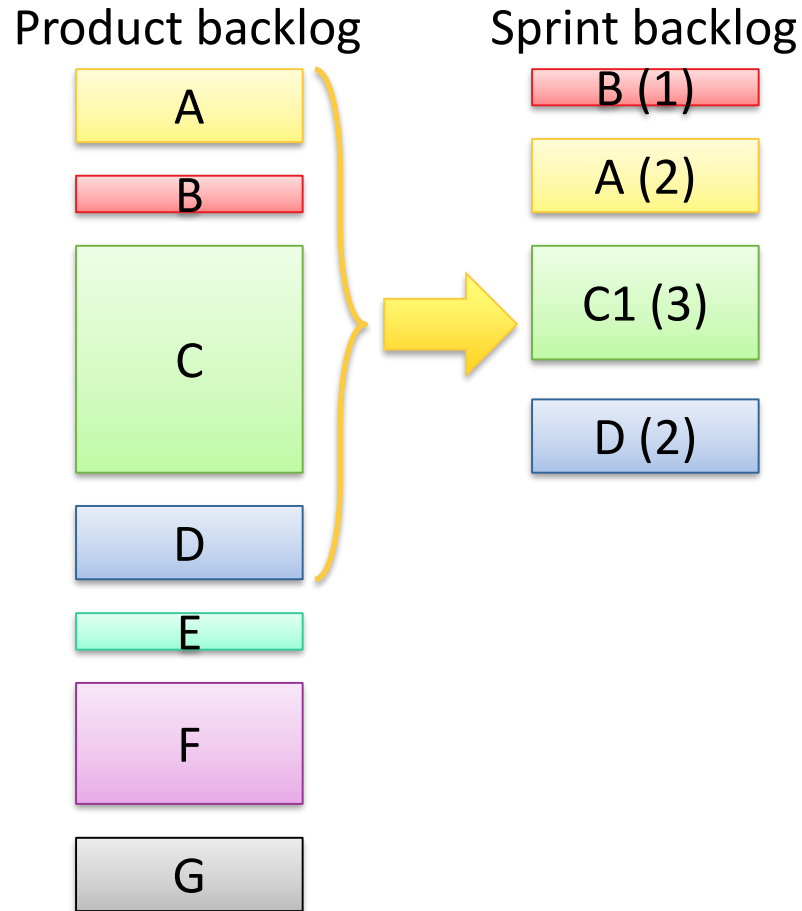
Будет
реализовано

Не будет
реализовано

Не решено

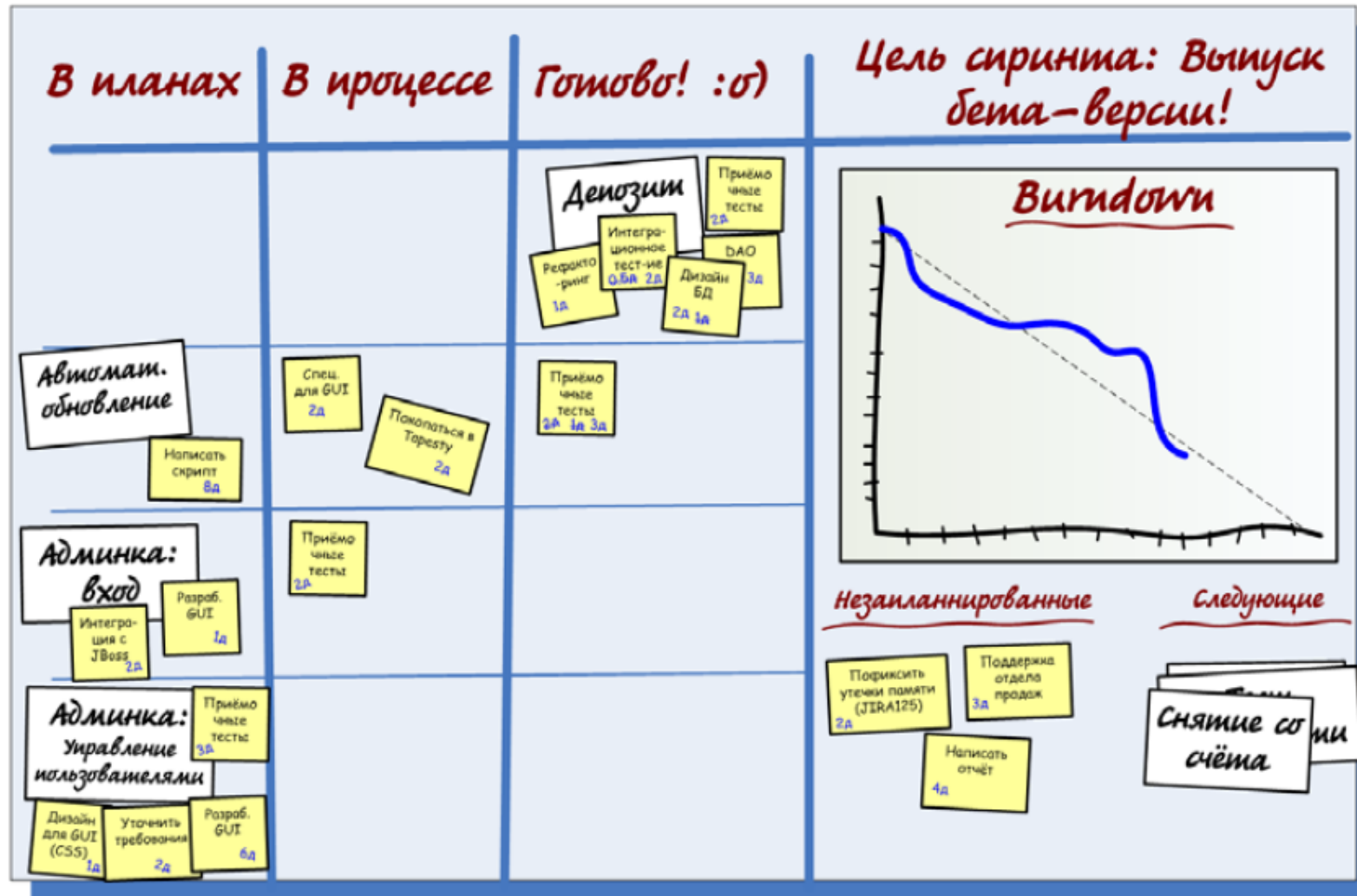
- Описание требований к продукту, сформулированное в виде одного или нескольких предложений на обычном языке
 - Нет двух требований с одинаковым приоритетом, если есть, то команда работает с понравившимся
 - Аналог ТЗ или иерархической структуры работ верхнего уровня
 - Пишутся либо одобряются владельцем продукта (Product Owner)
 - У каждого проекта один Product Backlog и один Product Owner, даже если команд несколько
-

Sprint Backlog



- Часть Product Backlog, которая будет реализована в рамках текущего спринта
 - Более точные формулировки, возможно разбиение на подзадачи достаточной детализации
 - Приоритезация
 - Оценка трудоемкости
-

Скрам Доска (Scrum Board)

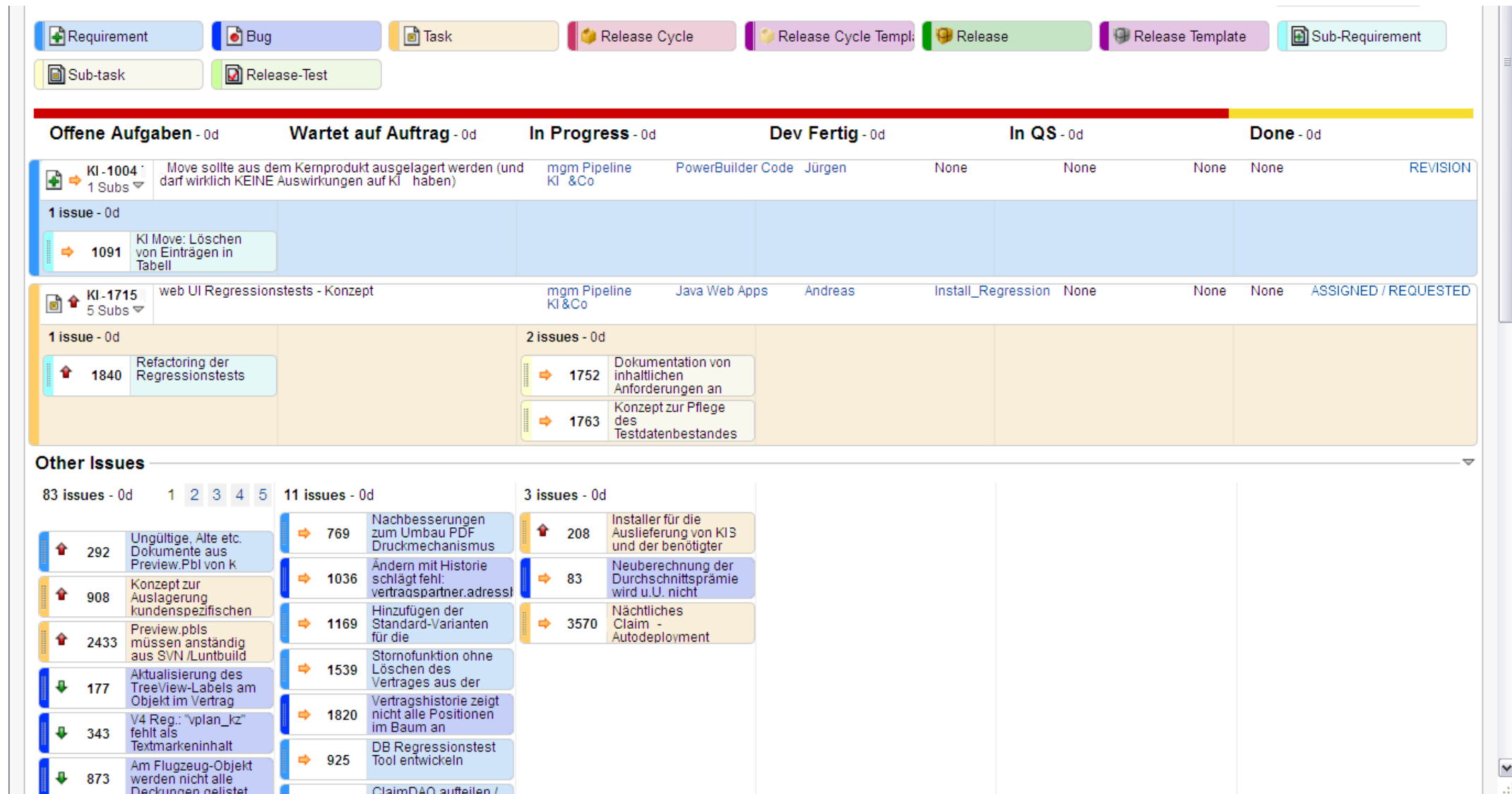


Скрам Доска (Scrum Board)

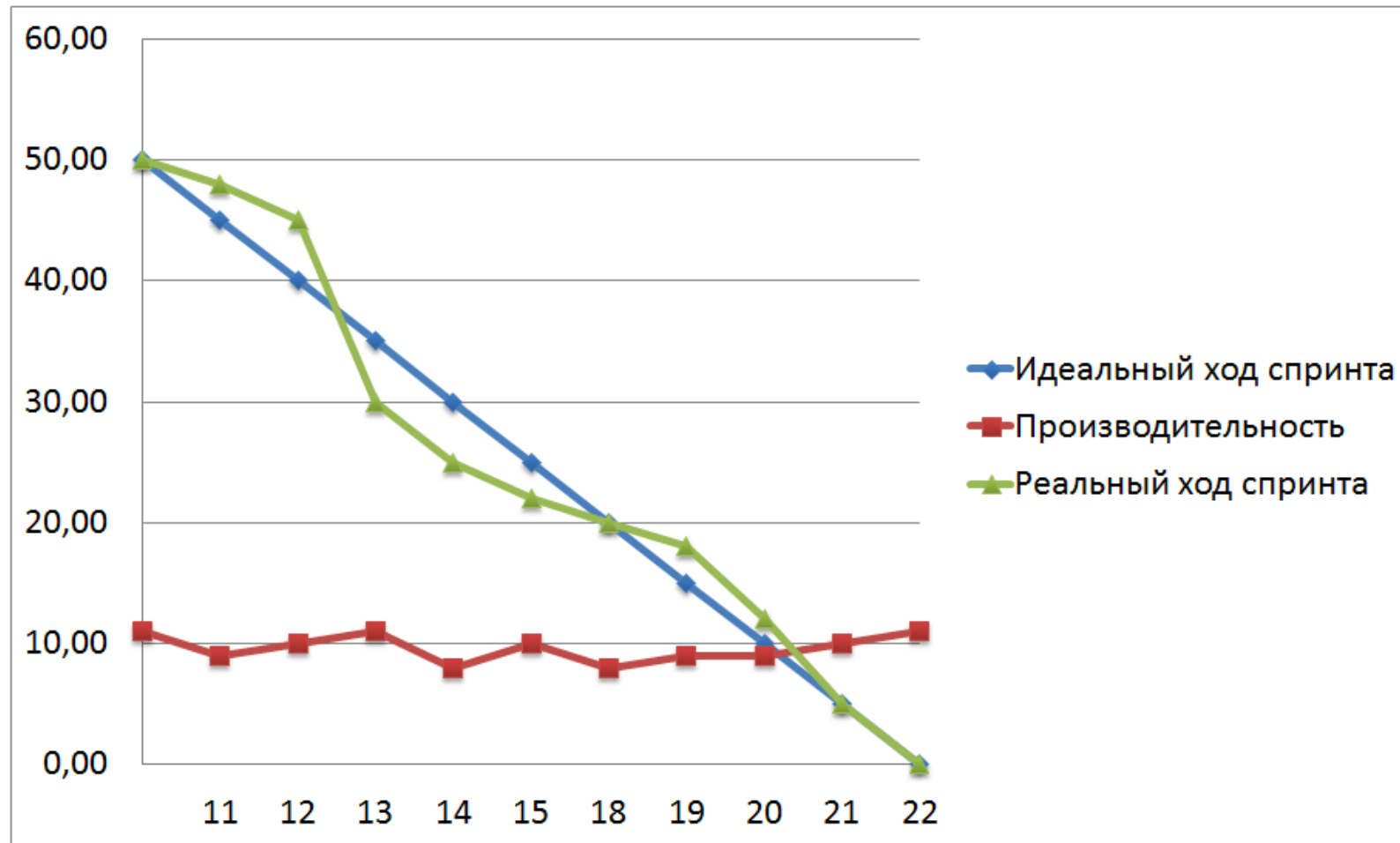
The screenshot displays a Scrum Board interface with three main columns: TO DO, IN PROGRESS, and DONE. Each column has a header bar indicating the total time for tasks in that column. Below the headers, tasks are listed as cards, each containing a task ID, description, version, assignee, and a clock icon. The TO DO column contains three tasks (504, 502, 509) and one task (507) partially visible. The IN PROGRESS column is currently empty. The DONE column is also empty. A mouse cursor is visible over a large yellow rectangular area at the bottom of the IN PROGRESS column.

TO DO - 32h	IN PROGRESS - 0h	DONE - 0h
SNY-418 Validate shared storage clustering option for Alfresco Sprint 3 Planning WCMS Open		
3 issues - 11h	0 issues - 0h	0 issues - 0h
504 418 Perform the tests to validate the setup. Version: Sprint 3 Planning Assignee: Umut Utkan		
502 418 Prepare two images with Alfresco installed and configured. Version: Sprint 3 Planning Assignee: Umut Utkan		
SNY-375 Setup the agreed project structure in Subversion and create all the maven artifacts accordingly. Sprint 3 Planning WCMS Open		
3 issues - 14h	0 issues - 0h	0 issues - 0h
509 375 Documentation of how to start (checkout, common maven goals, etc.) Version: Sprint 3 Planning Assignee: Umut Utkan	508 375 Optimize the POM's and perform necessary clean-up. Version: Sprint 3 Planning Assignee: Umut Utkan	
507 375 Do the migrations for the current project code base to meet the new project structure. Version: Sprint 3 Planning		

Скрам Доска (Scrum Board)



Обратный график реализации историй (Burndown chart)



Инкремент (Increment)

- ✓ Это сумма, как всех элементов Бэклога Продукта, завершенных во время Спринта, так и всех инкрементов предыдущих Спринтов.
- ✓ К концу Спринта Инкремент должен быть Готов, что подразумевает его соответствие **критериям Готовности (Definition of "Done")*** и готовность к использованию.
- ✓ Он должен быть готовым к использованию вне зависимости от решения Владельца Продукта его выпускать или повременить.

*Понимание Критериев Готовности должно быть общим между исполнителями работ и теми, кто ее результаты принимает.

Из заключения «Руководство по Скраму»

- ✓ Роли, артефакты, правила и события Скрама не подлежат изменению.
- ✓ При внедрении отдельных элементов данного фреймворка, полученный результат не может называться Скрамом.
- ✓ Скрам существует только в качестве цельного фреймворка.
- ✓ Он также может служить контейнером для других техник, методологий и практик.

*«**Scrum** имеет лишь косвенное отношение к программному обеспечению.
Зато он напрямую связан с руководством и организацией работы
топ-компаний в мире»*

Хиротака Такеучи и Икуджиро Нонака, «Wise Leadership»



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Дополнительно

Изменение объема работ

- Если запланировано слишком много, следует уменьшить объем поставки, здесь нужен владелец продукта
- Если в ходе работы клиент добавляет новую историю, он должен отказаться от одной из имеющихся

Настройка процесса

Целевые параметры

- Производительность
- Время выполнения задачи
- Качество (уровень дефектности и т.д.)
- Предсказуемость

Управляемые параметры:

- Размер команд
- Число команд
- Число одновременно выполняемых работ (для Kanban)
- Длительность итераций
- Детализация планов
- и т.д.

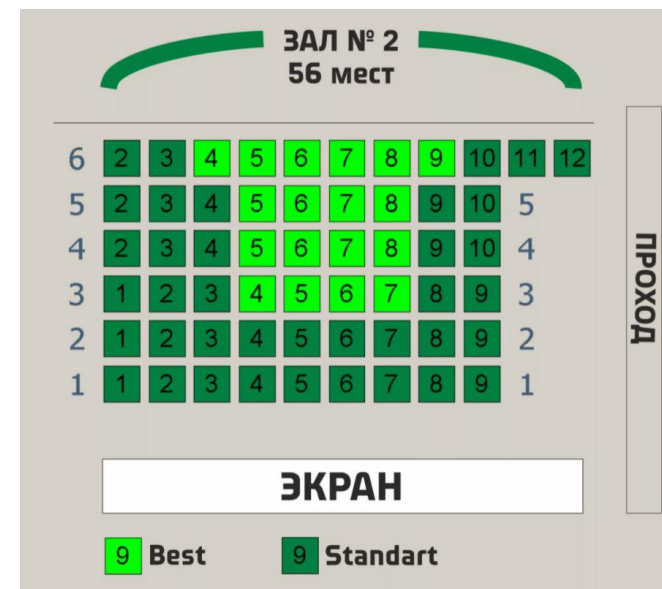
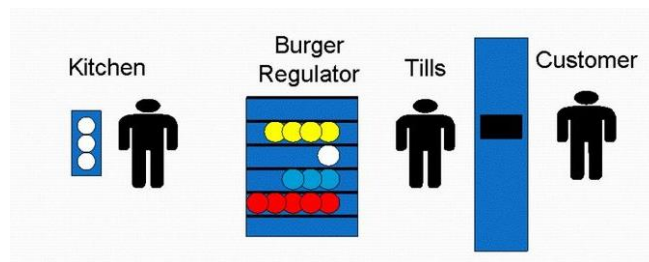


Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Тема 5. Kanban

Виды канбана

- Канбан, как сигнал
- Канбан-система, как система ограничений
- Производственный канбан, как система борьбы с потерями
- Канбан-метод, как метод улучшения сервиса



Бережливое производство (Lean Manufacturing)

Время возникновения – начало 1990-х годов

Предпосылки возникновения:

- Сокращение жизненного цикла продукта
- Ориентация на потребителя (customization)
- Глобальная конкуренция

Основы гибкого производства

- Модульный дизайн (облегчает внесение изменений в продукт)
- Информационные технологии (обеспечивают быстроту реагирования на заказы)
- Партнерство (для ускорения вывода на рынок продуктов)
- Культура, основанная на знаниях (инвестиции в развитие персонала)

Основные принципы lean

- уважение к людям;
- создание ценности;
- постоянное совершенствование, так как потери похожи на гравитацию;
- 90% того, что мы делаем – это потери;
- работать спокойно, а не быстро.

На что жалуются в операционных процессах?

Сложно определить, где находятся «бутылочные горлышки» всех работ

Много жалоб на качество поставляемого продукта

Задачи неравномерно распределены во времени

Частое «выгорание» людей: большая нагрузка, а результатов работы не видно

История канбана

Впервые принципы канбан были задекларированы Таичи Охно еще в 1940х годах в компании Toyota

Вдохновением для них послужили принципы наполнения полок в крупных торговых сетях

Принципиально важным в канбане является использование визуальных сигналов для идентификации состояния системы

Позднее, в 1980х годах послужил основой для идей кайдзен

Канбан – это набор принципов, благодаря которому команды и организации могут визуализировать свою работу, найти и обезвредить «бутылочные горлышки» и добиться драматических операционных улучшений в производительности и качестве



Потери по Т.Оно

- muda – потери, действия, потребляющие время, усилия, ресурсы, пространство, но не создающие ценности для потребителей;
- muri – перегрузка членов команды, рабочих, сотрудников, станков и прочих мощностей при работе с повышенной интенсивностью;
- mura – неравномерность, неритмичность операций;
- дополнительный вид muda – это нерациональное использование талантов, имеющихся в организации

Прямые виды потерь

- потери из-за перепроизводства
- потери времени из-за ожидания
- потери при ненужной транспортировке
- потери из-за лишних этапов обработки
- потери из-за лишних запасов
- потери из-за ненужных перемещений
- потери из-за выпуска дефектной продукции

Перепроизводство

- производство раньше, либо больше, чем необходимо для удовлетворения потребительского спроса:
 - требует хранения;
 - создает дополнительные потери;
 - увеличивает оборачиваемость запасов;
 - продукция может устареть во время хранения;
 - готовая продукция может быть украдена

Транспортировка

- избыточные перемещения незаконченной продукции или людей между этапами обработки, задачами и процессами:
 - это результат нерационального проектирования системы или расположения объектов;
 - может создавать дополнительные риски повреждения при транспортировке;

Ожидание

- недозагрузка команды и оборудования в процессе работы:
 - ожидание на уровне ресурса – это простой;
 - ожидание на уровне клиента – это долгий цикл производства;

Запасы

- количество сырья, незаконченной и готовой продукции в системе:
 - продукт должен проходить в системе без задержек;
 - увеличивает затраты на хранение и снижает оборачиваемость запасов;
 - увеличивает время производства и ожидания результата клиентом;
 - если производится материальная продукция, то она требует дополнительных складских помещений;

Брак

- повторение операций или затраты на исправление в процессе:
 - это несоответствие ожиданию на предъявление качественного результата с первого раза;
 - может быть следствием применяемых материалов, оборудования, инструментов, технологий, навыков людей;
 - требует дополнительных ресурсов для обеспечения непрерывного производства;

Излишняя обработка

- обработка сверх потребности клиента:
 - может быть следствием внутренних регламентов и стандартов, которые не превосходят истинные потребности клиента (клиент и сам может завышать требования;
 - может быть следствием желания достичь лучшего качества и идеального результата в работе;

Перемещения

- избыточные перемещения деталей, инструментов, результатов работы в рамках одной операции:
 - обычно является следствием нерациональной организации рабочего места.

Неполное использование человеческого ресурса

- это нерациональное использование талантов, имеющих в организации:
 - результат специфики менеджмента и отношения к персоналу;
 - может влиять на безопасность труда и травматизм;
 - создание системы постоянного совершенствования становится невозможным без вовлечения всего персонала и раскрытия таланта каждого.

Итого

- Из 7 основных muda 5 являются ресурсно-ориентированными (перепроизводство, транспортировка, брак, излишняя обработка, перемещение), 2 являются процессно-ориентированными (запасы и ожидание)

Что это?

- Это не столько методология, сколько философия
- Делать только то, что действительно нужно
- Уменьшение выполняющейся в данный момент работы
- Визуализация
- Оптимизация

Общие идеи

Ограничение work-in-progress (задач, над которыми идет работа) помогает увидеть, где в системе сосредоточены проблемы

Большая доска с карточками-задачами

Доска всегда представляет актуальный «снимок» текущего статуса проекта

Пытаемся ограничить объем работы на каждой стадии процесса

«Бутылочные горлышки» легко идентифицировать «на глаз» благодаря скопившимся карточкам

Все принципы нацелены на минимизацию простоя и убытков

Работая над операционными процессами – важно не «перегореть» команду. Канбан делает акцент на том, что люди «вытаскивают» задачу из общей очереди, и весь процесс – от ее старта до завершения – виден всем участникам.

Как работает канбан?

Обычно доска выглядит так:

To do, разделенное на 3 категории:

Бэклог – задачи, которые нужно сделать где-то в будущем

В ожидании – ограниченное количество задач,

Готовы – задачи, готовые к тому, чтобы начать по ним работу

In progress – обязательно лимитируется одновременное количество задач, над которыми можно работать

Done – для отчетности и оценки собственной производительности

Основные принципы

Визуализируйте процесс работы – это ключевой фактор успешных изменений. Не понимая особенностей процесса работы, трудно совершить правильные изменения.

Ограничивайте работу в процессе. Ограничение количества задач в работе и одновременная работа с блокерами и задержками ускоряет каждую задачу. Как результат ускоряется весь рабочий цикл.

Течение задач (задержки на определенных этапах, возникающие блокеры) – ваш основной предмет мониторинга, измерений и аналитики

Закон Литтла

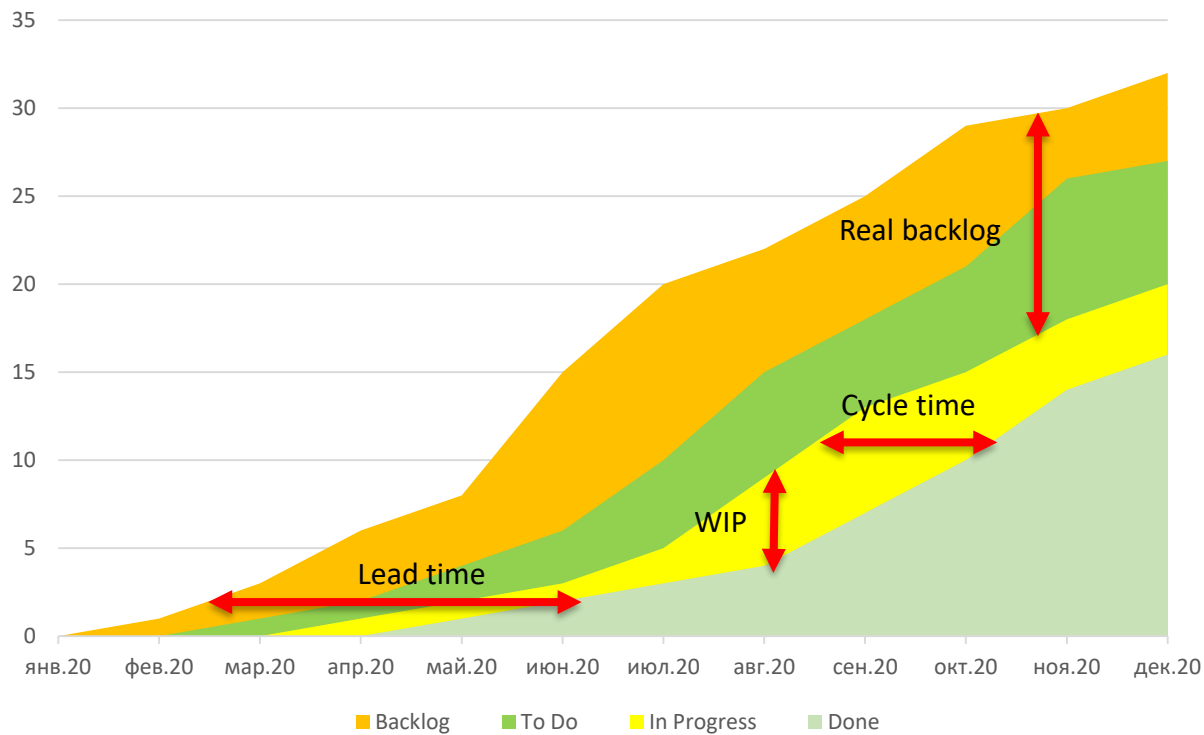
В поточной системе, не имеющей тренда (в которой все элементы выбраны или поставлены), наблюдается простая зависимость между средними значениями показателей за определенный период. Эта зависимость известна как закон Литтла:

$$\overline{\text{Скорость поставки}} = \frac{\overline{WiP}}{\overline{\text{Время производства}}}$$

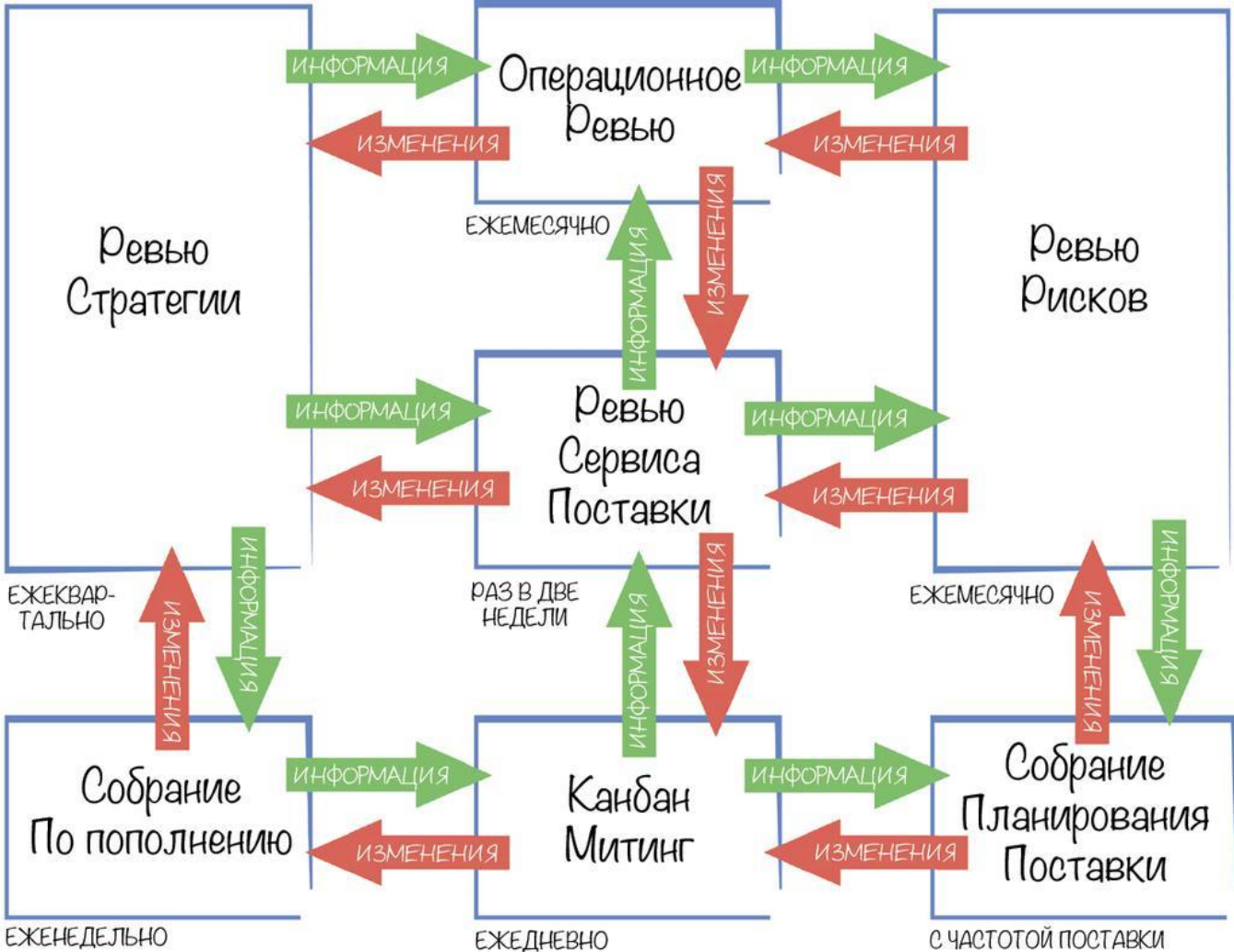
Вместо производства для kanban используется время в процессе:

$$\overline{\text{Пропускная способность}} = \frac{\overline{WiP}}{\overline{TiP}}$$

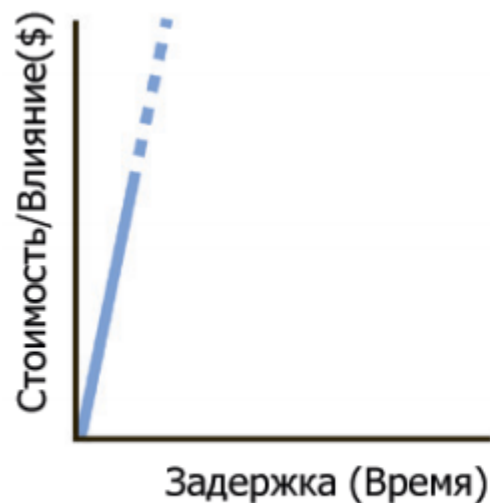
Кумулятивная диаграмма потока (Cumulative flow diagram)



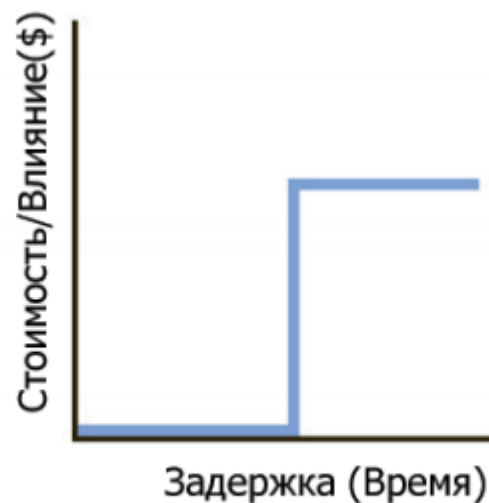
Встречи



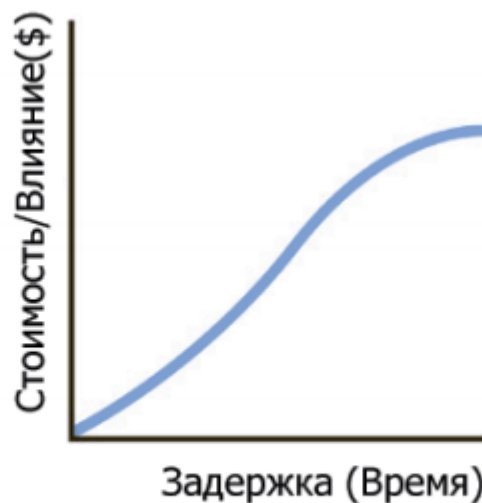
Стоимость задержки



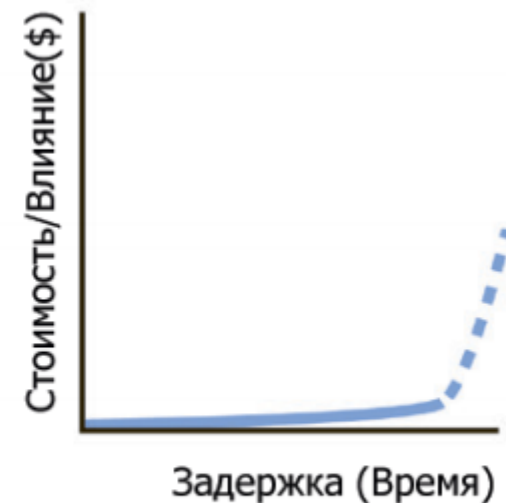
УСКОРЕННЫЙ



ФИКСИРОВАННАЯ ДАТА



СТАНДАРТНЫЙ



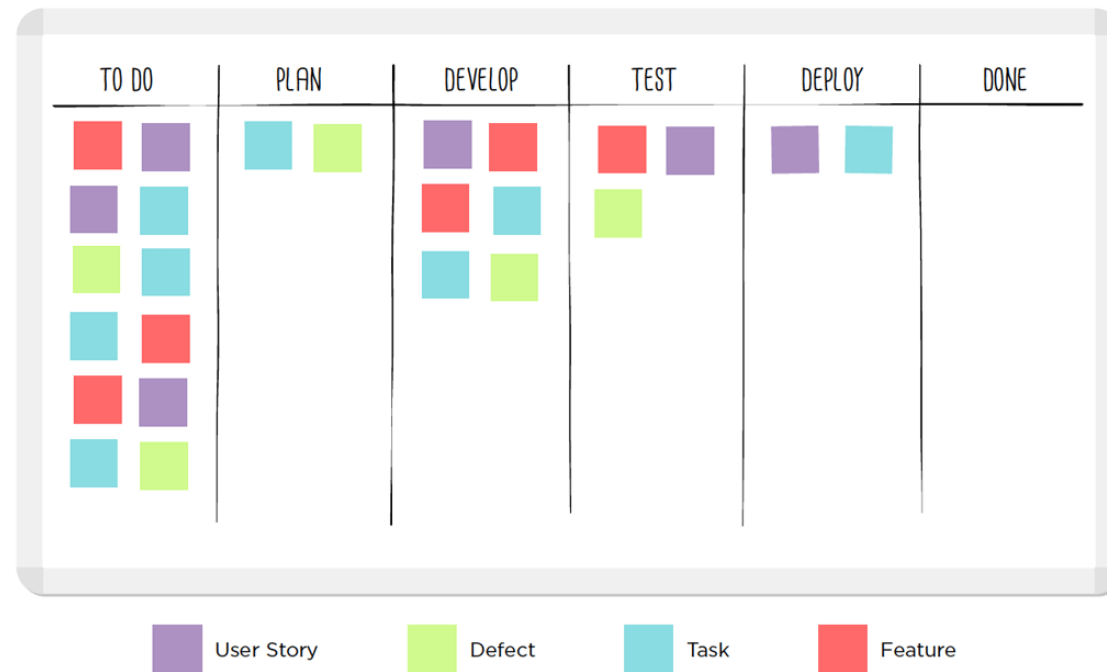
НЕМАТЕРИАЛЬНЫЙ



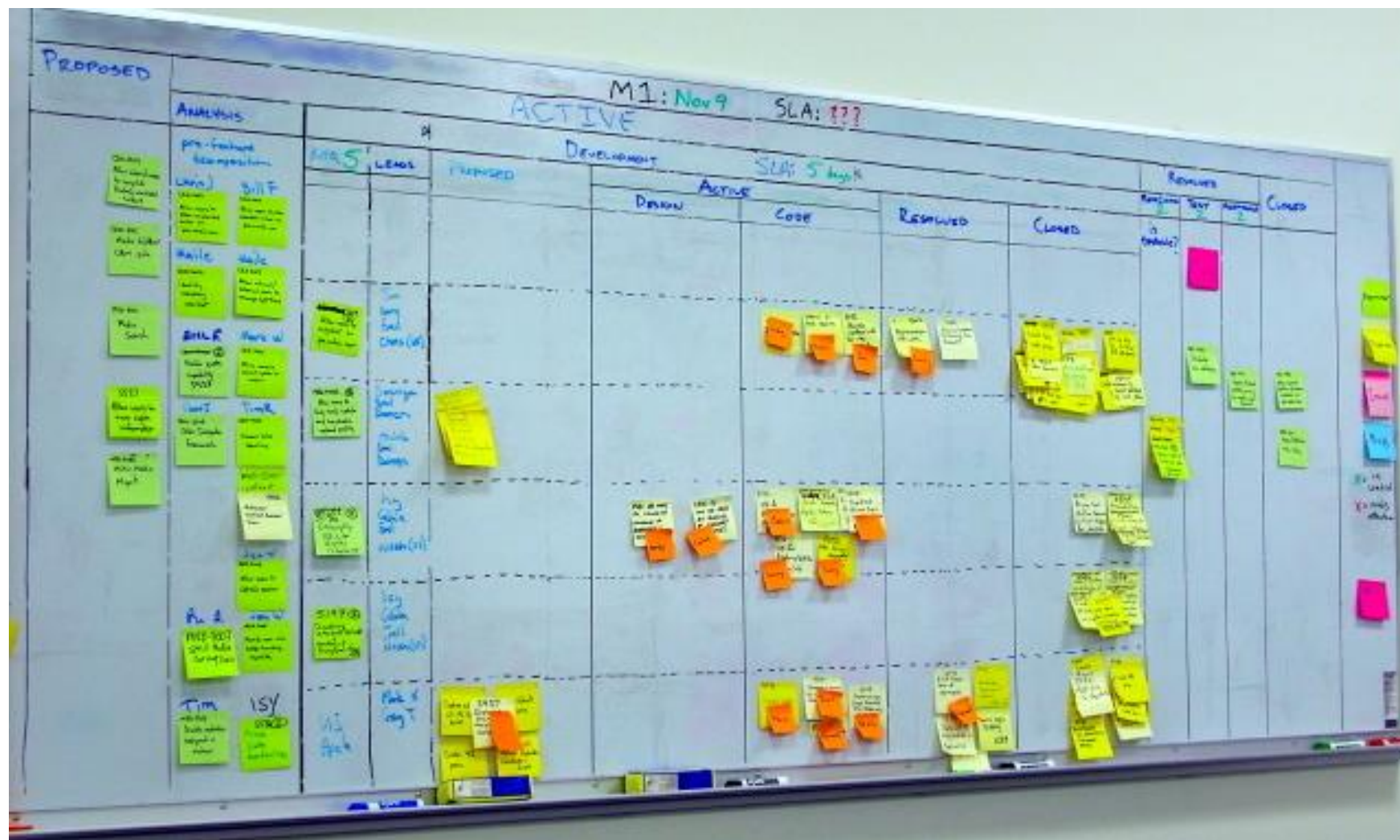
**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Виды досок

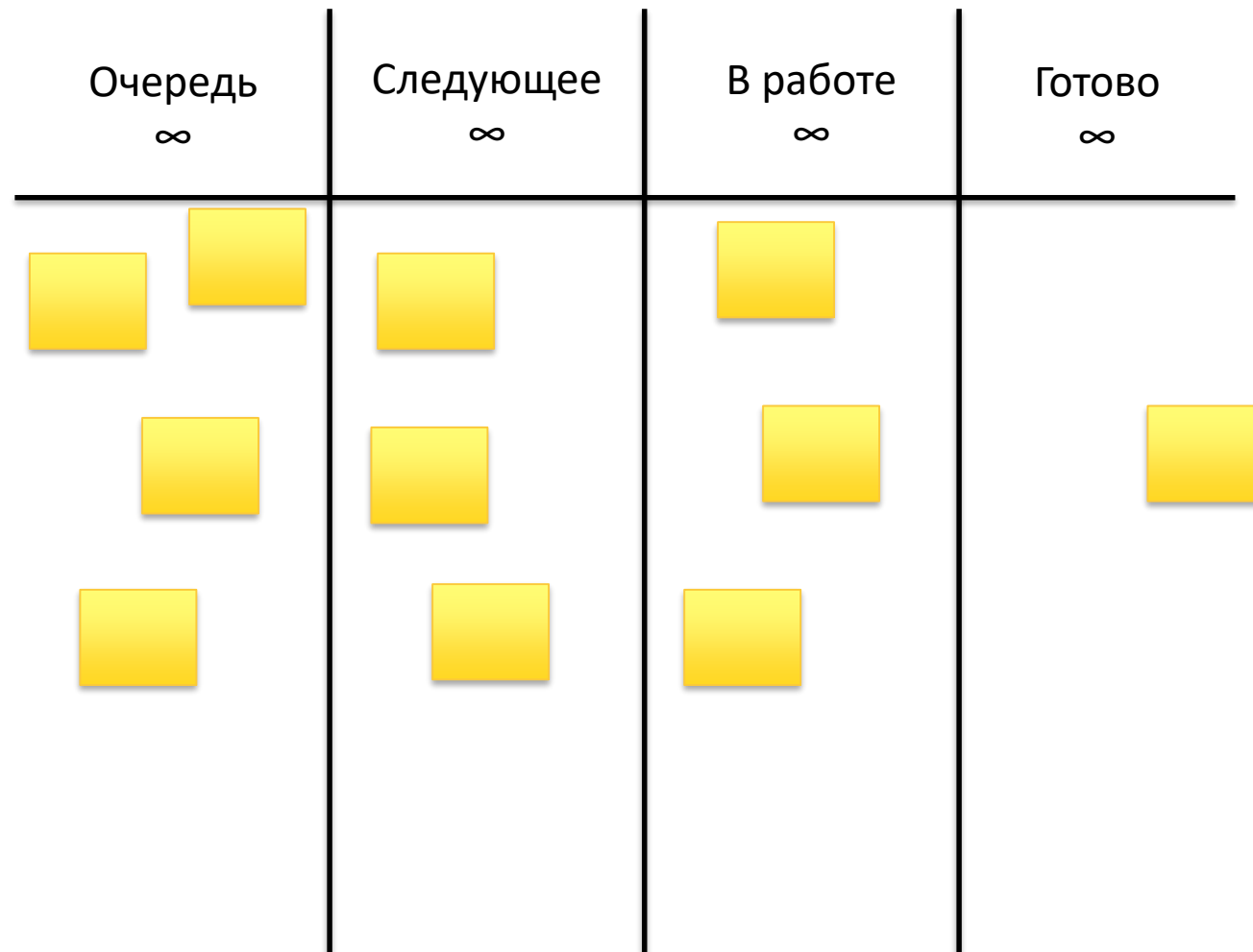
Пример визуализации



Реальная доска



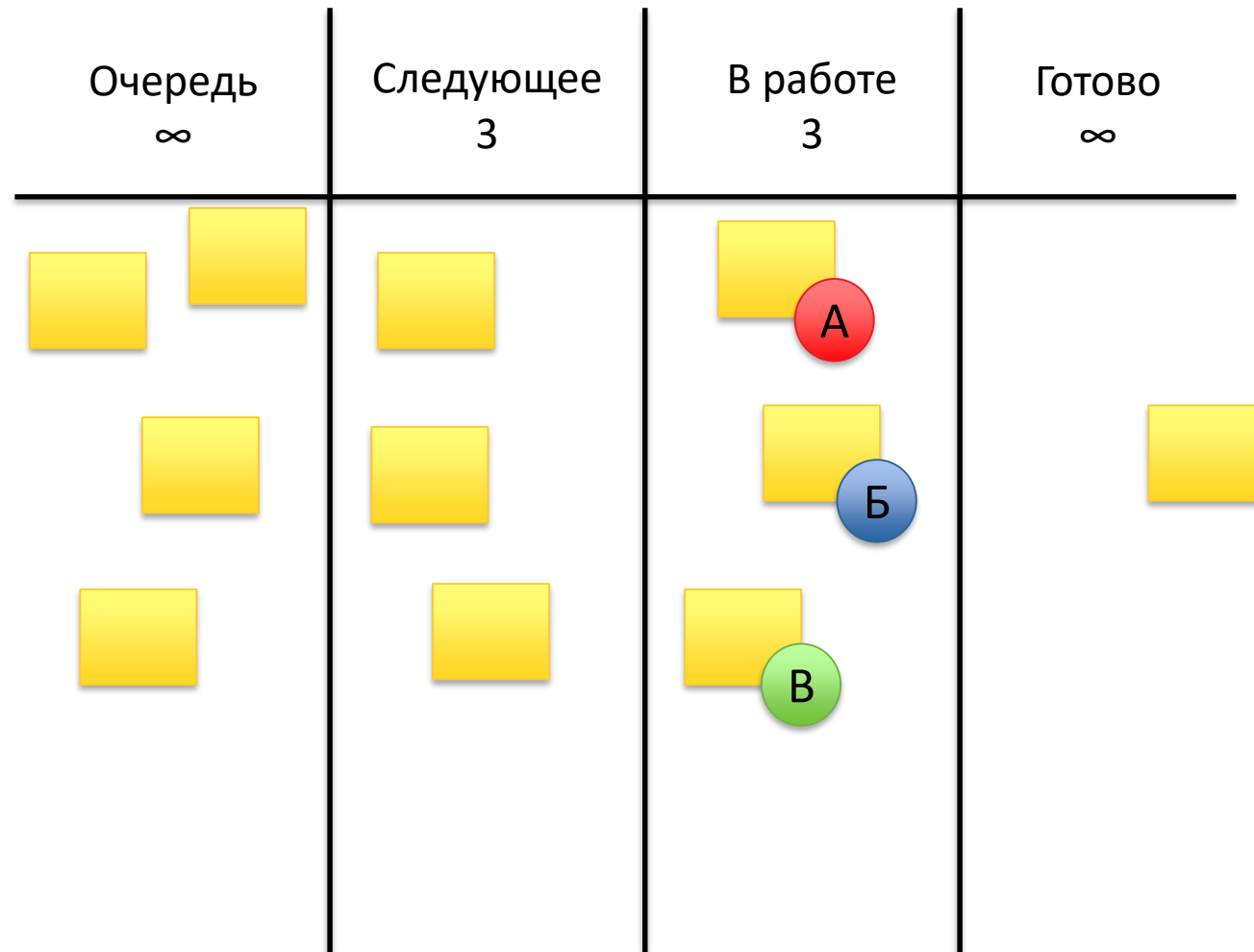
Персональный канбан



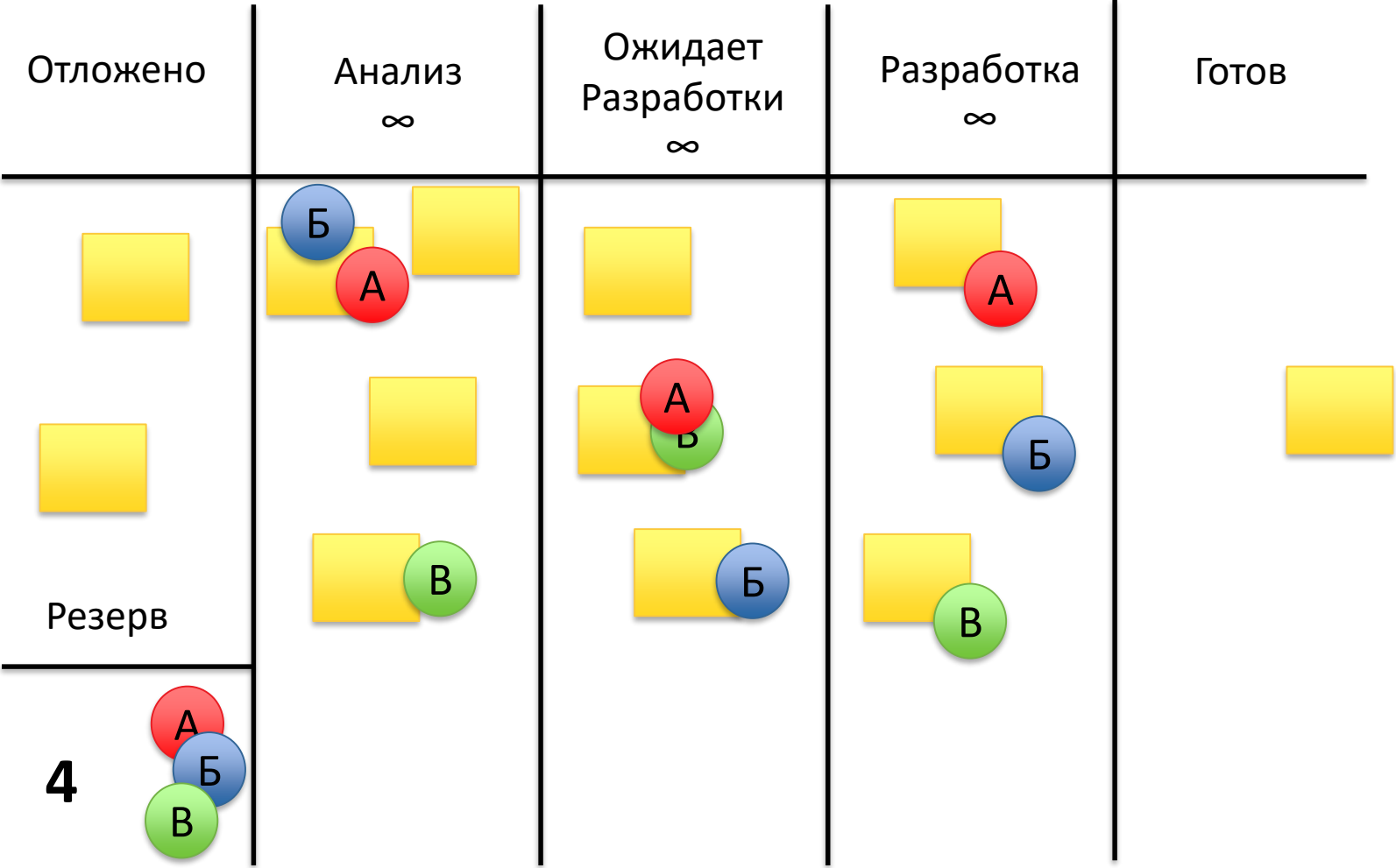
Агрегированный персональный канбан

Член команды	Очередь ∞	Следующее (3 на человека)	В работе (3 на человека)	Готово ∞
Иван	 			
Мария				
Артем				
Елена				

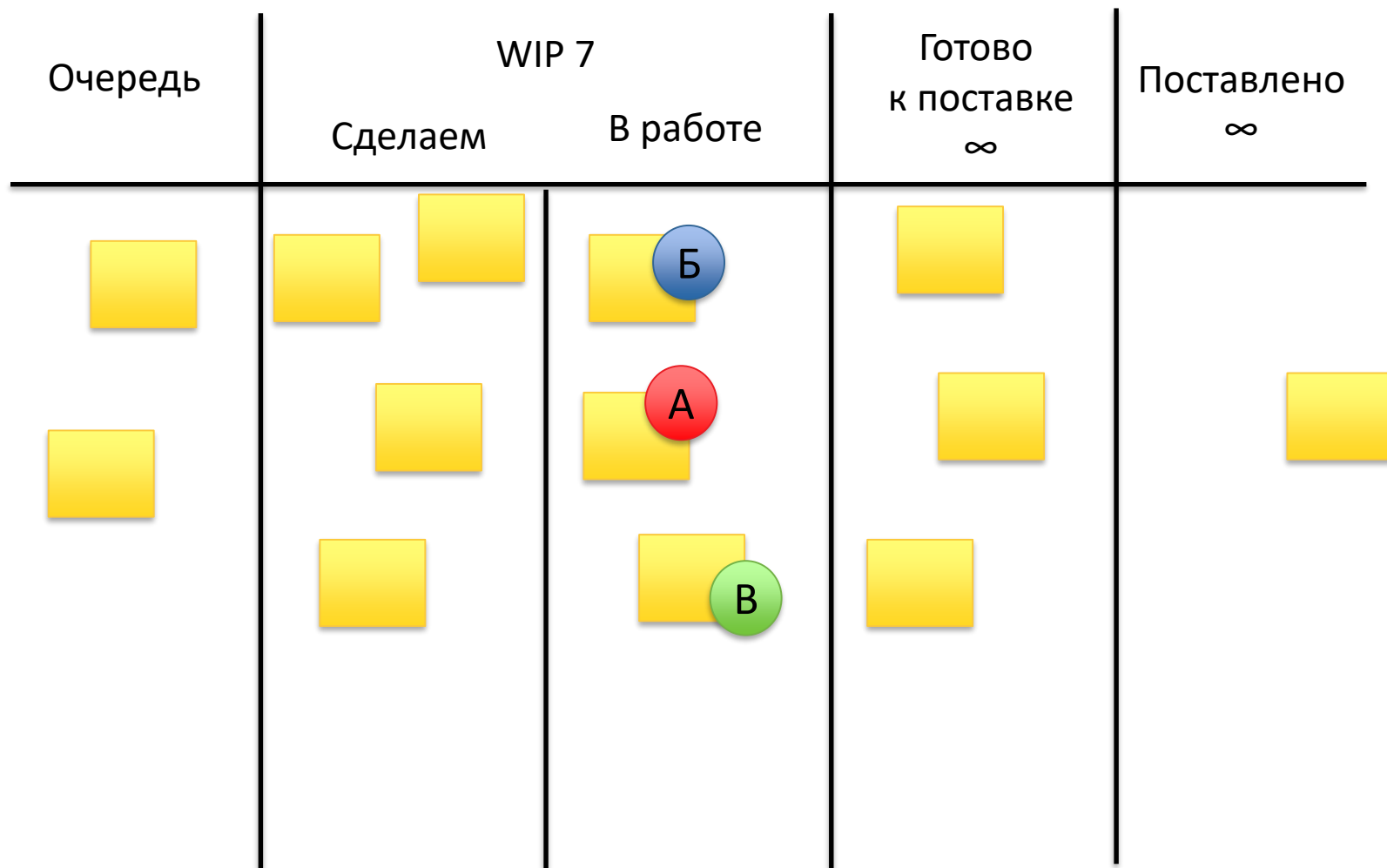
Командный канбан



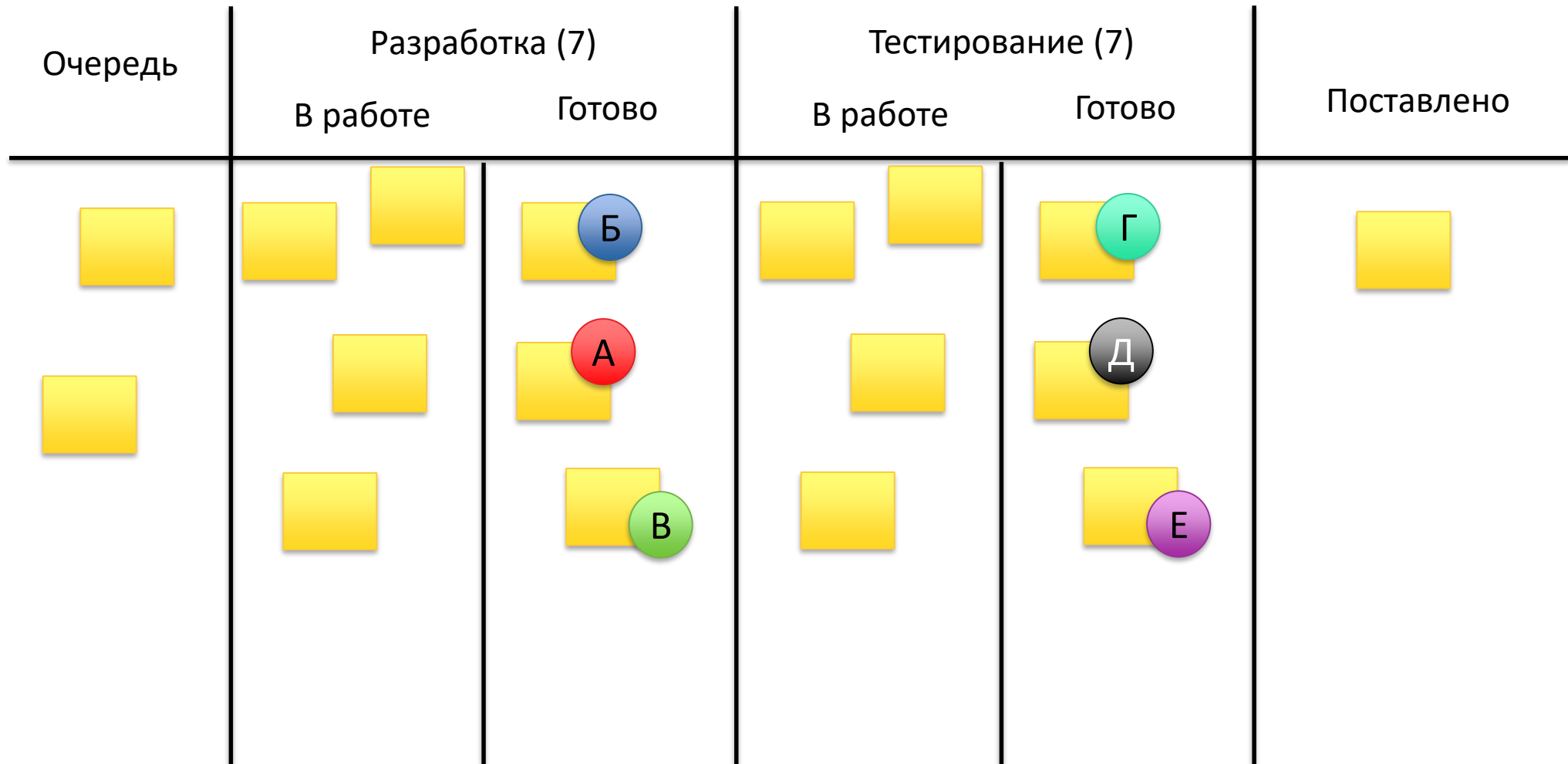
Командный канбан с WIP на человека



Командный канбан с развязанными каденциями и постоянным WIP



Агрегированный командный канбан





**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

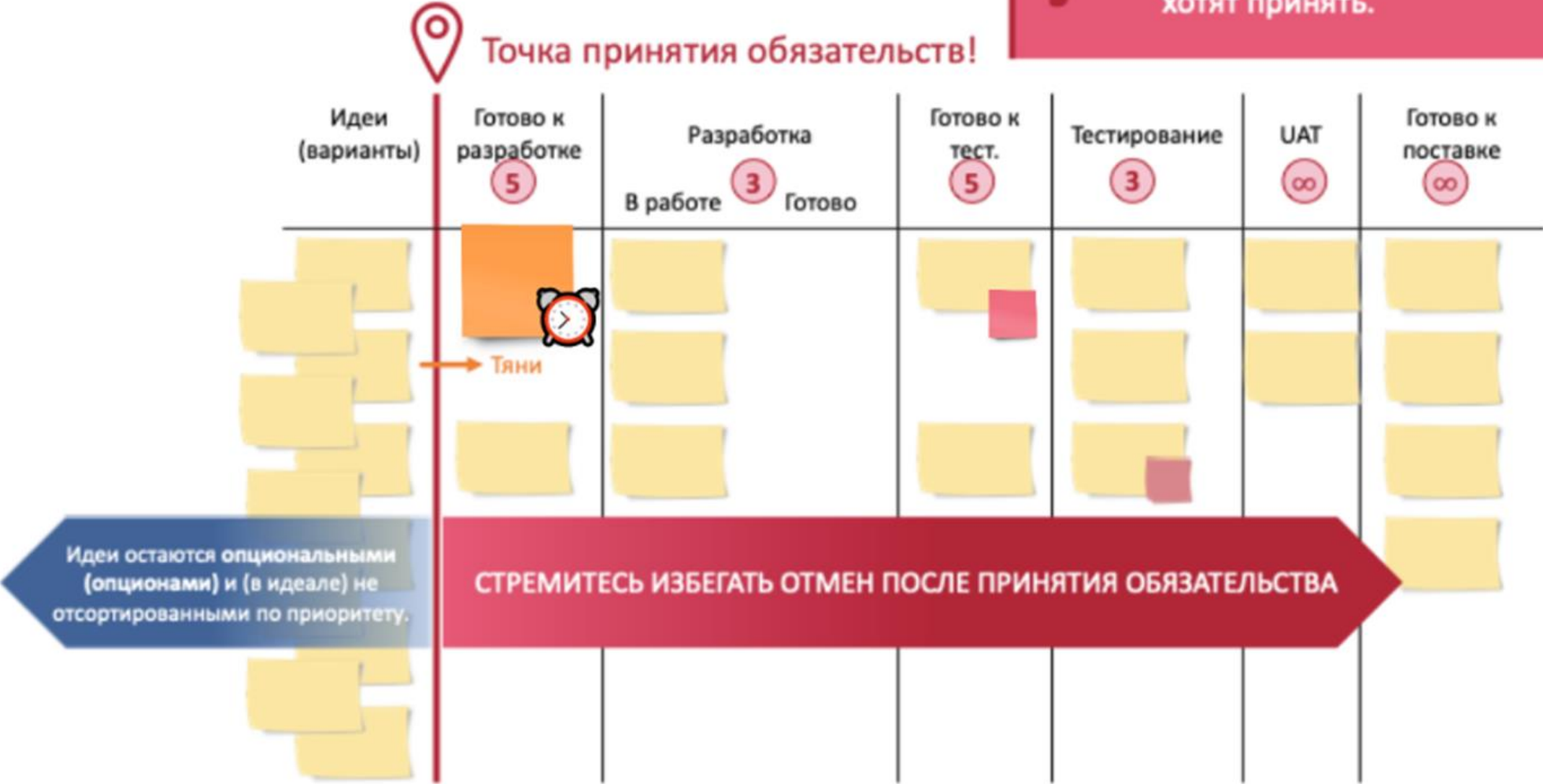
Что еще важно знать

Точка принятия обязательств

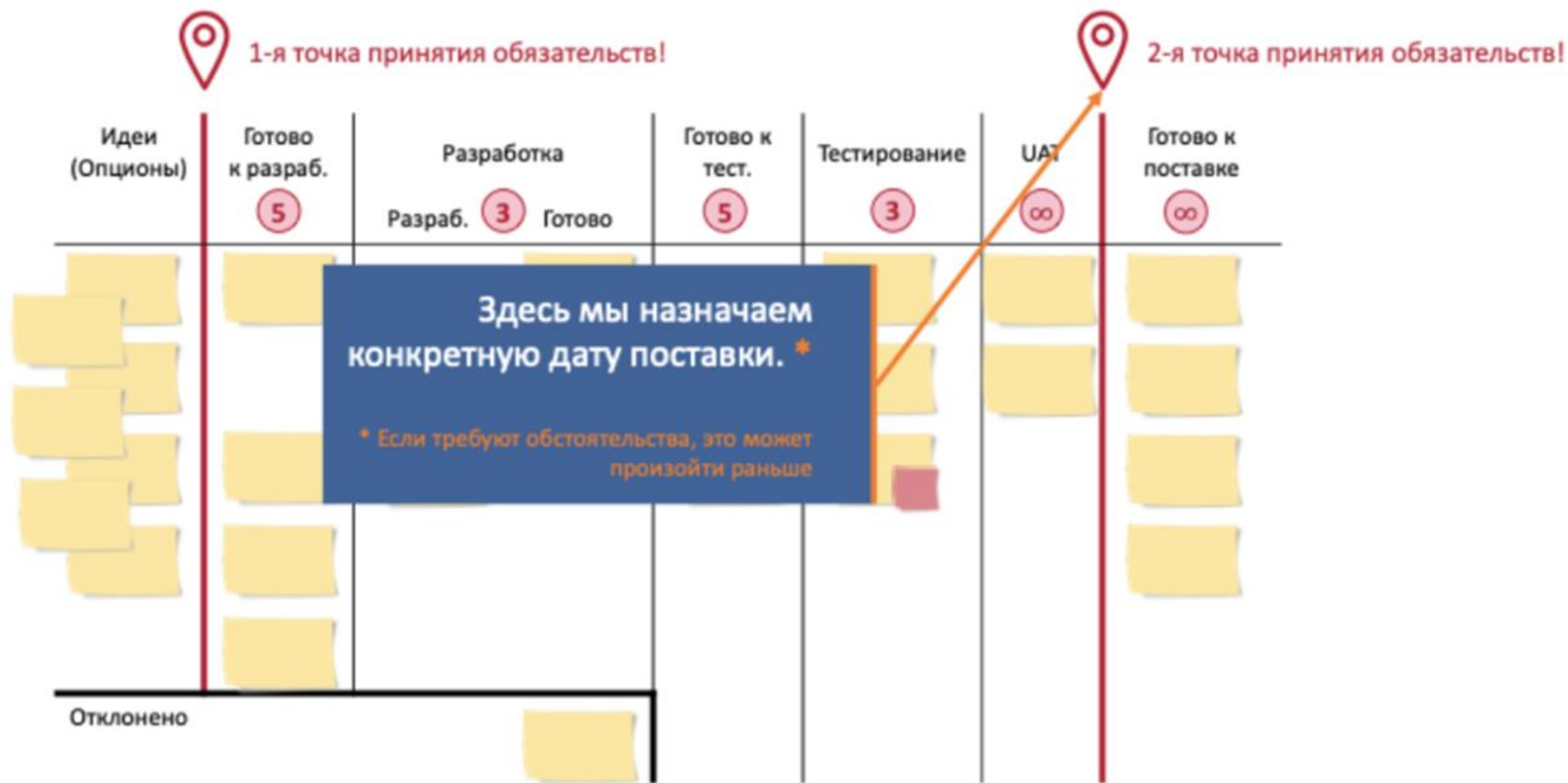


Первая точка принятия обязательств

! Мы принимаем на себя обязательство начать работу. Мы уверены, что доставку хотят принять.



Вторая точка принятия обязательств



Метод STATIK (Systems Thinking Approach to Introducing Kanban)

Сервис:

Менеджер:

Дата:

1 – Источники неудовлетворённости Внутренние Что вас не удовлетворяет внутри сервиса? Внешние Кто неудовлетворен вашей работой вовне? С кем у вас возникают конфликты?						3 – Рабочий поток Нарисуйте рабочий процесс для каждого типа рабочего элемента. Есть ли сходства и различия между ними? Есть ли одновременная и неупорядоченная деятельность? Есть ли внешние зависимости, риски и т. д.?	4 – Классы обслуживания Для каждого типа рабочего элемента укажите текущие классы обслуживания, их политики и ожидания по поставке.																																				
2 – Анализ запросов <table><tr><th>Тип рабочего элемента</th><th>Источник</th><th>Пункт назначения</th><th>Частота поступления</th><th>Природа запроса</th><th>Ожидания клиента</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>							Тип рабочего элемента	Источник	Пункт назначения	Частота поступления	Природа запроса	Ожидания клиента																															5 – Каденции по пополнению и поставке Как часто вы сейчас пополняете очередь? Как часто поставляете? Укажите частоту для каждого типа рабочего элемента.
Тип рабочего элемента	Источник	Пункт назначения	Частота поступления	Природа запроса	Ожидания клиента																																						
						6 – Визуализация Канбан-системы Этот раздел предназначен для простого эскиза, помогающего команде доставки, менеджеру и тренеру определить основные контуры визуальной платы. Они могут включать в себя плавательные дорожки, двухуровневую структуру, использование цвета и т. д. Нет необходимости делать этот раздел миниатюрной копией фактической доски.																																					

Сервис:

Менеджер:

Дата:

Неопределенный

Срочный

Стандартный

Фикс дата

1 – Источники неудовлетворённости

Внутренние

Что вас не удовлетворяет внутри сервиса?

Если заказ висит в разных статусах долгое время, значит что-то не адекватно настроено (сложно)

wip-лимитов нет

Много задач в статусе "готовы к деплою", но они стали не актуальными, а актуальные только "догоняют"

Внешние

Кто неудовлетворен вашей работой вовне? С кем у вас возникают конфликты?

Срок поставки

Качество

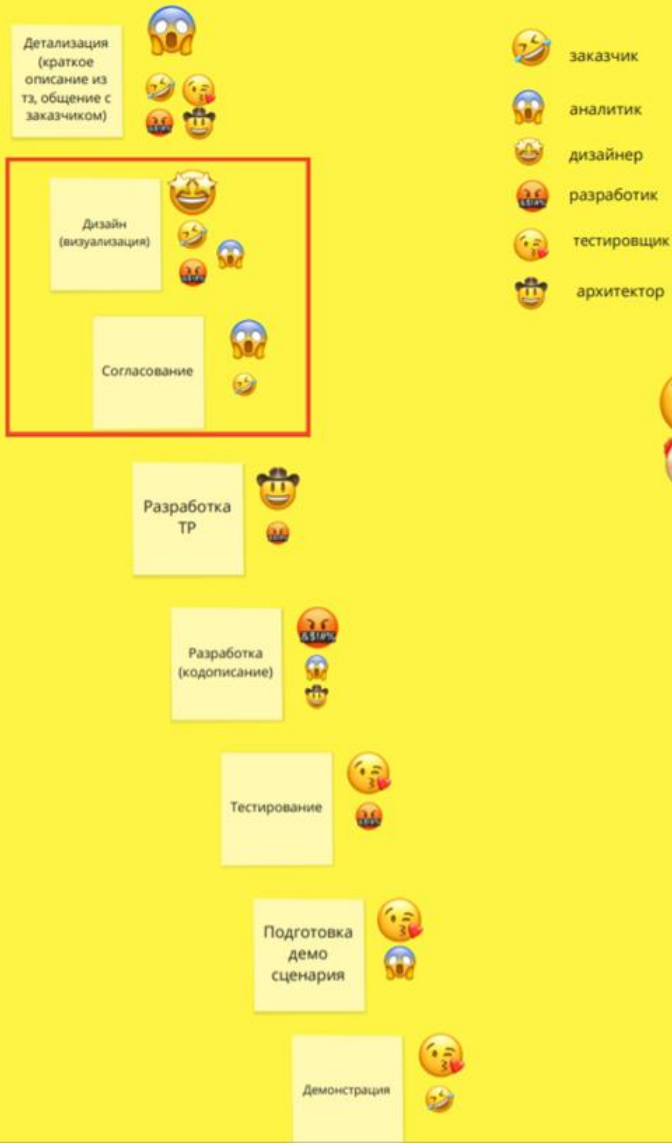
Неудовлетворённость качеством поставок (детализированных описаний задач)

2 – Анализ запросов

Тип рабочего элемента	Источник	Пункт назначения	Частота поступления	Природа запроса	Ожидания клиента
Запрос фичи	Заказчик	Заказчик	3 в неделю	регулярно	3 недели
Запрос изменения	Заказчик	Заказчик	по запросу	рандомно	2 мес
Хотфикс, запрос изменений	Заказчик	Заказчик	по запросу (1 в мес)	рандомно	1 день
	Заказчик	Заказчик	один раз	хаотик	в указанную дату

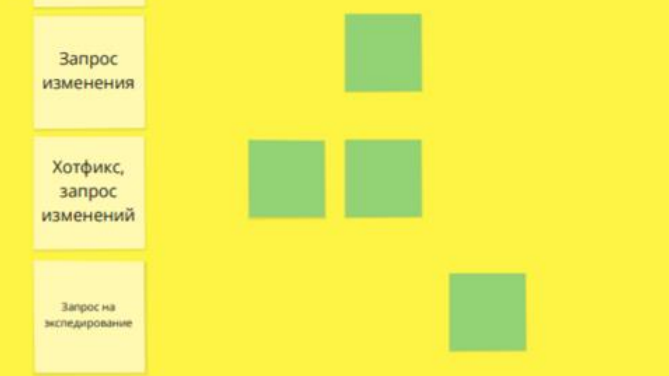
3 – Рабочий поток

Нарисуйте рабочий процесс для каждого типа рабочего элемента. Есть ли сходства и различия между ними? Есть ли одновременная и неупорядоченная деятельность? Есть ли внешние зависимости, риски и т. д.?



4 – Классы обслуживания

Для каждого типа рабочего элемента укажите текущие классы обслуживания, их приоритет и ожидания по срокам



5 – Каденции по пополнению и поставке

Как часто вы сейчас пополняете очередь? Как часто поставляете? Укажите частоту для каждого типа рабочего элемента.

6 – Визуализация Канбан-системы

Этот раздел предназначен для простого эскиза, помогающего команде доставки, менеджеру и тренеру определить основные контуры визуальной платы. Они могут включать в себя плавательные дорожки, двухуровневую структуру, использование цвета и т. д. Нет необходимости делать этот раздел миниатюрной копией фактической доски.

Пример визуализации для бизнеса

Модель Остервальдера

Макет бизнес модели

Предназначен для

Разработан

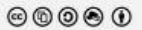
Дата: Месяц: Год:

Итерация №

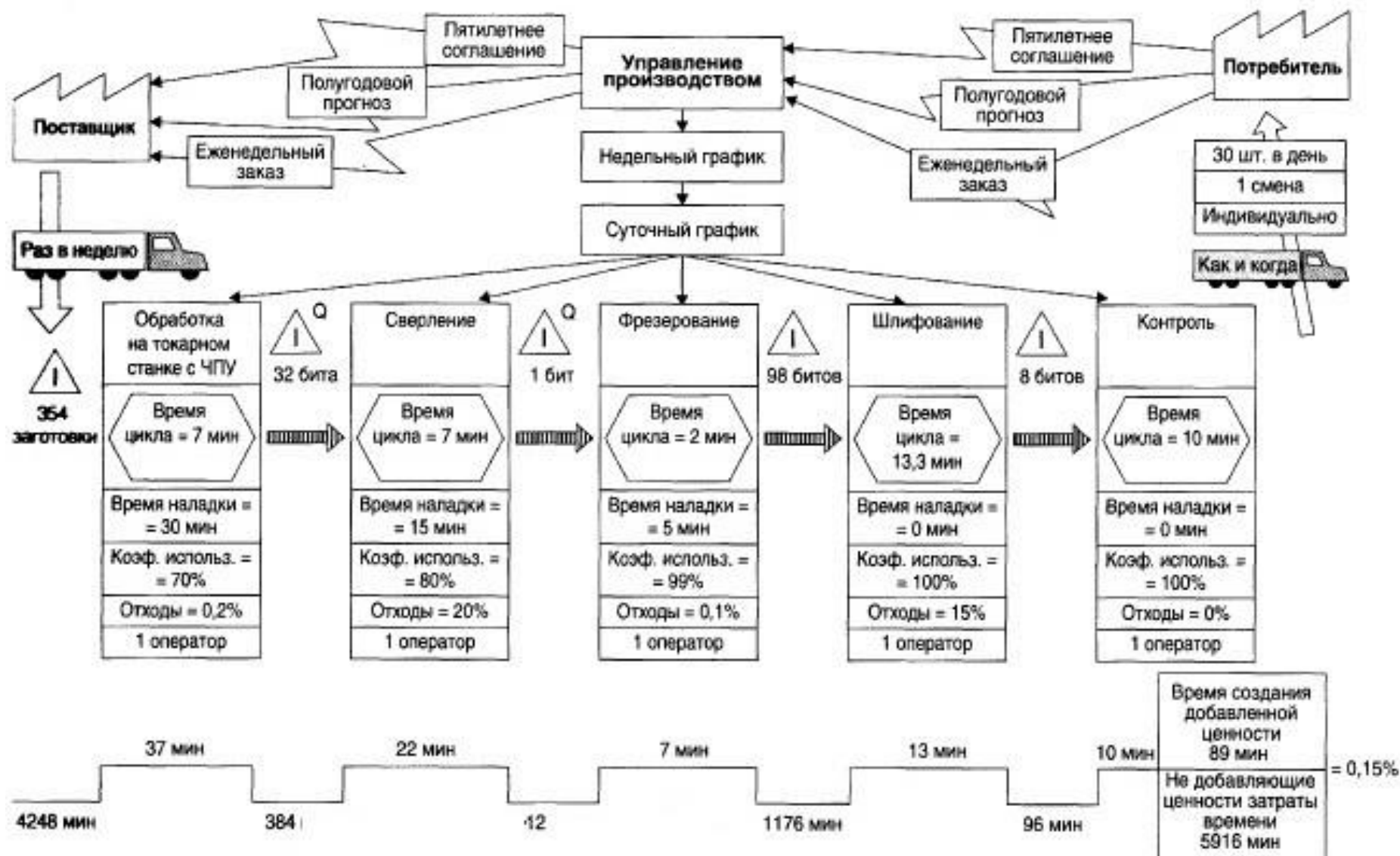
Ключевые партнеры  Кто наши ключевые партнеры? Кто наши ключевые поставщики? Какие ключевые ресурсы мы получаем от партнеров? Какими ключевыми активностями занимаются партнеры? мотивация для сотрудничества: Оптимизация и экономия Уменьшение рисков и неопределенностей Получение конкретного ресурса и активности	Ключевые активности  Какие активности нужны для придания значимости продукту? Для каналов поставок? Для построения отношений с заказчиком? Получения прибыли? категории: Производство Решение проблем Планирование/Установление связей Ключевые ресурсы  Какие ресурсы нужны для придания значимости продукту? Для каналов поставок? Построения взаимоотношений с заказчиком? Получения прибыли? типы ресурсов: Физические Интеллектуальные Человеческие Финансовые	Достоинства предложения  В чем ценность продукта для заказчика? Какие проблемы мы помогаем решать? Какие группы продуктов и сервисов мы предоставляем пользователю сегменту? Какие нужды заказчика мы удовлетворяем? характеристики: Новизна Производительность Поддержка под клиента "под ключ" Долгий Стабильность/Бренд Цена Уменьшение затрат Уменьшение рисков Доступность Удобство	Отношения с заказчиком  Какие взаимоотношения с нами ожидает получить каждый представитель пользовательского сегмента? Какие из них уже установлены? Как они интегрируются с основной бизнес-моделью? Насколько они затратны? примеры: Персональная подборка Удаленная поддержка Самостоятельное Сообщество Совместная разработка Каналы поставки  Через какие каналы будет происходить поставка в пользовательский сегмент? Как происходит это сейчас? Насколько эти каналы интегрированы? Какой работает лучше? Какой наиболее экономически эффективен? Как ими пользуется потребитель? этапы поставок: 1. ознание 2. Оценка 3. Закупка 4. Доставка 5. Постпродажный этап	Пользовательские сегменты  Для кого мы делаем продукт значимым? Кто наш главный потребитель? Массовый рынок Нишевый рынок Сегментированный Разнообразный Многочисленный
Структура затрат  Каковы самые значительные затраты в нашей модели? Какие из ключевых ресурсов самые дорогие? Какие из ключевых активностей самые дорогие? как бизнес работает: зависит от затрат (средняя структура затрат, низкая цена значимости продукта, максимальная автоматизация) зависит от продукта (нацеленность на создание качественной продукции, продукт на первом месте) примеры характеристик: Фиксированные затраты (аренда, аренда, различные платежи) Непостоянные затраты Экономия на различных ресурсах		Источники доходов  За что именно в продукте платят потребители? За что они платят сейчас? Как они оплачивают продукт сейчас? Как бы они хотели оплачивать? Сколько приносят каждый из источников в итоговый доход? типы: продажа активов плата за использование плата за подписку аренда лицензирование Реклама фиксированный расчет цен: цена по прецеденту зависимость от сегмента зависимость от версии динамический расчет цен: негосударственная управление доходами Real-time рынок		

www.businessmodelgeneration.com

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA. Email: info@creativecommons.org



Карта потока ценности

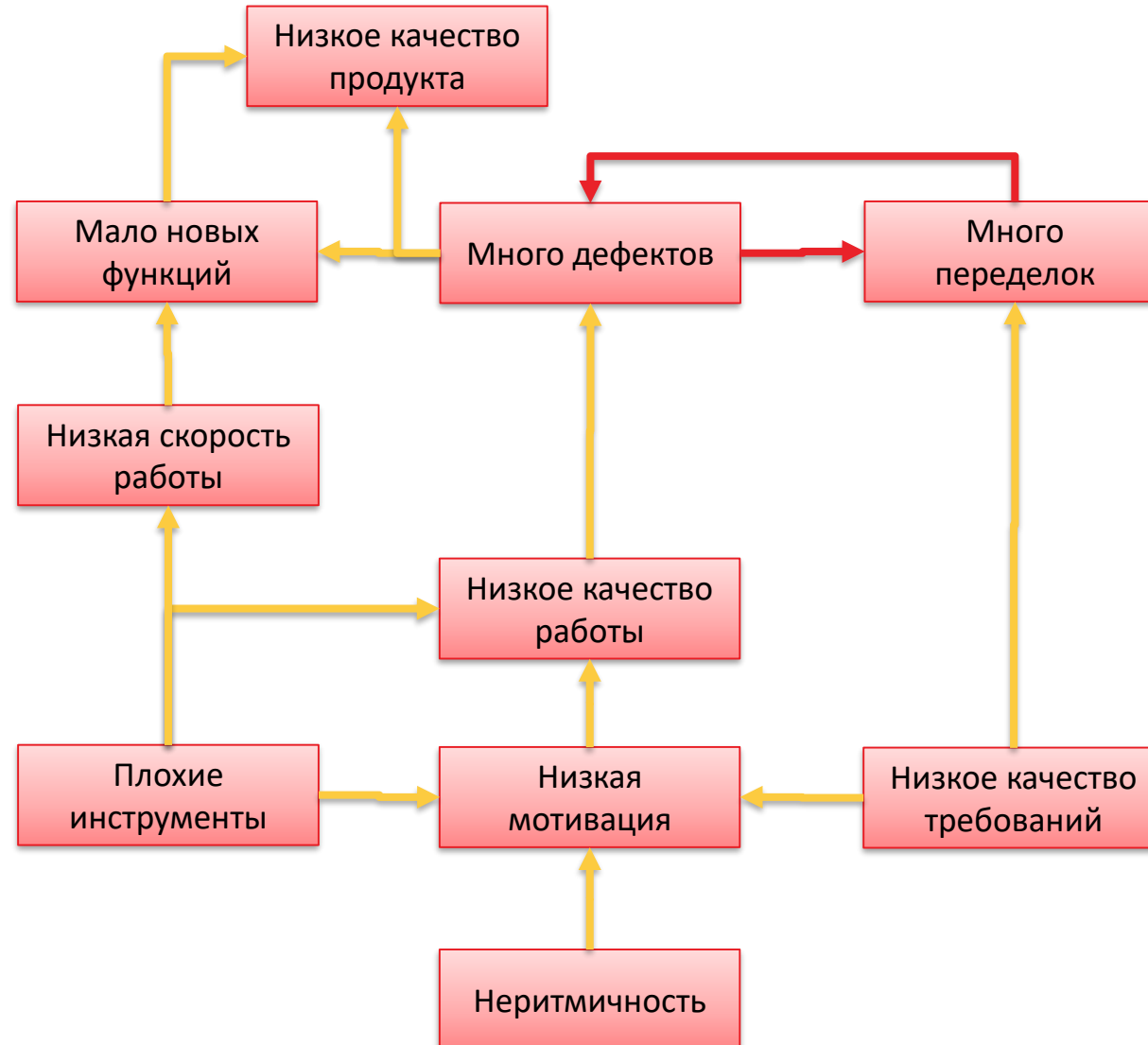


Пример банки колы

	Хранение перед обработкой	Обработка	Хранение после обработки	Скорость обработки	Общее число дней	Общие отходы
Рудник	0	20 мин	2 недели	1000 т/ч	14	0
Обогатительная фабрика	2 недели	30 мин	2 недели	-	28	0
Плавильная печь	3 недели	2 часа	2 недели	-	104	2%
Горячая прокатка	2 недели	1 мин	4 недели	5 шт/мин	42	2%
Холодная прокатка	2 недели	<1 мин	0.039%		42	2%
Изготовление банок	2 недели	1 мин	4 недели	2000 шт/мин	42	14%
Разливочная фабрика	4 дня	1 мин	5 недель	1500 шт/мин	39	4%
Региональный склад	0	0	3 дня	-	3	4%
Магазин	0	0	2 дня	-	2	4%
Дом	3 дня	5 мин	-	-	3	100%
Всего	5 месяцев	3 часа	6 месяцев		319 дней	36%

Пример визуализации ДТР

Модель Голдрата



Как начать?

Начните с личных задач или задач внутри маленькой команды

Договоритесь внедрять все изменения инкрементально и эволюционно

Уважайте текущий процесс, роли и области ответственности

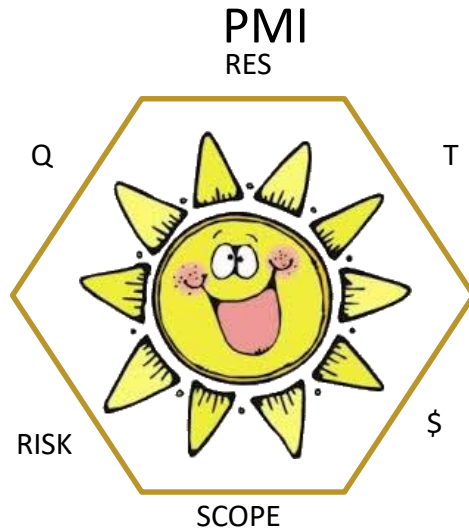
Пропагандируйте лидерство и ответственность на всех уровнях



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Сравнение подходов

Общий взгляд



Удовлетворить стейкхолдеров
Уложиться в ограничения:

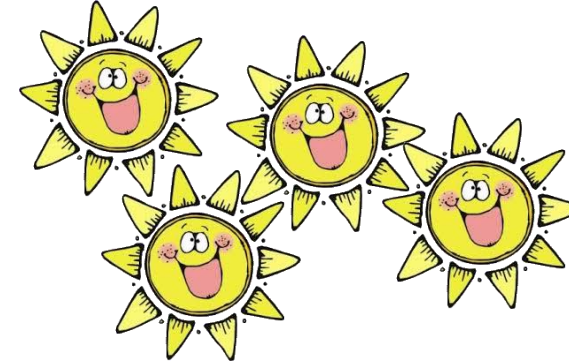
- Сроки
- Бюджет
- Скоуп
- Риски
- Качество
- Ресурсы

Scrum



Удовлетворить
стейкхолдеров

Kanban



- Выстроить процесс
- Сформировать поток
- Непрерывно совершенствовать процесс производства

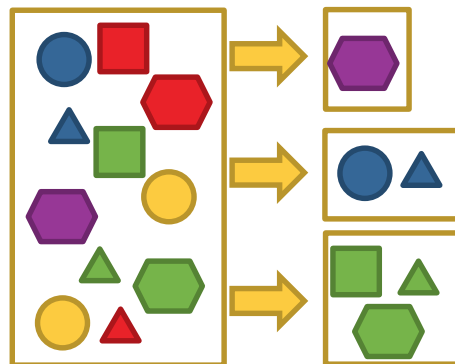
Роли

PMI



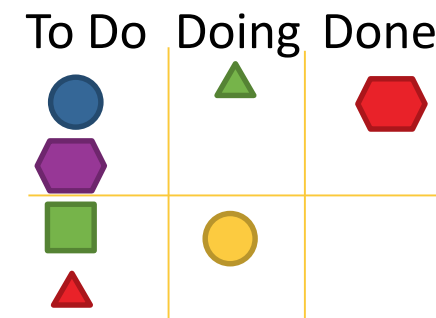
Project manager (менеджер проекта)
Sponsor (спонсор)
Project team (команда проекта)

Scrum



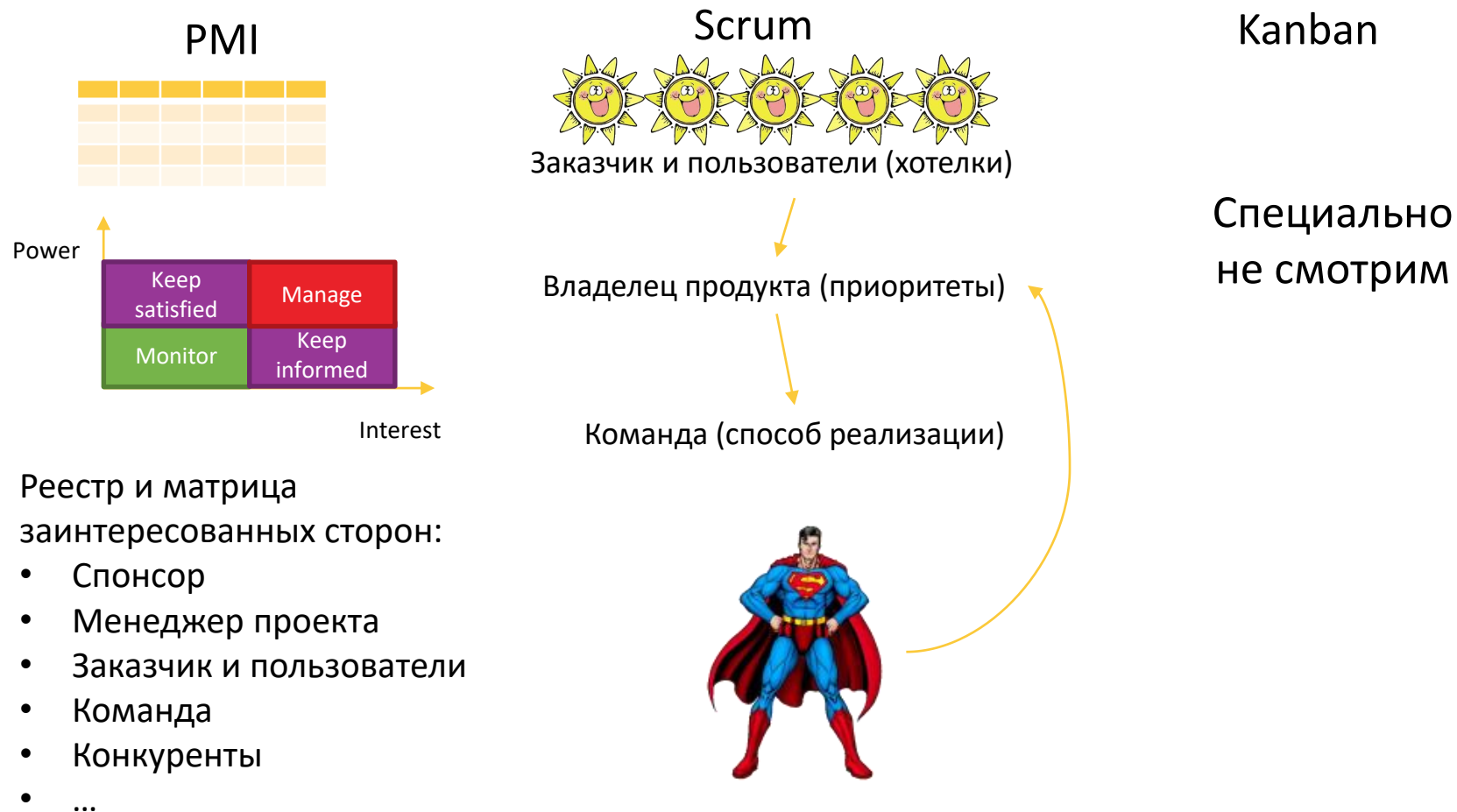
Product owner
(владелец продукта)
Scrum master (скрам-мастер)
Project team (команда проекта) -
кроссфункциональная

Kanban



Project team (команда проекта) – разделение по фазам

Заинтересованные стороны



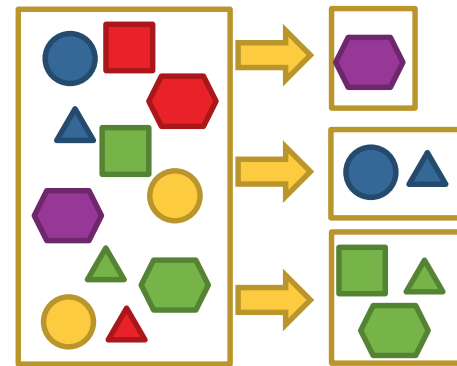
Содержание

PMI



- Устав проекта
- Скоуп проекта
- Сбор требований
- Описание содержания проекта
- ИСР

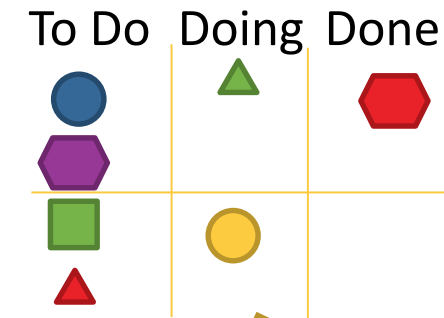
Scrum



Видение
проекта

Кто-то
наполняет
(неважно кто)

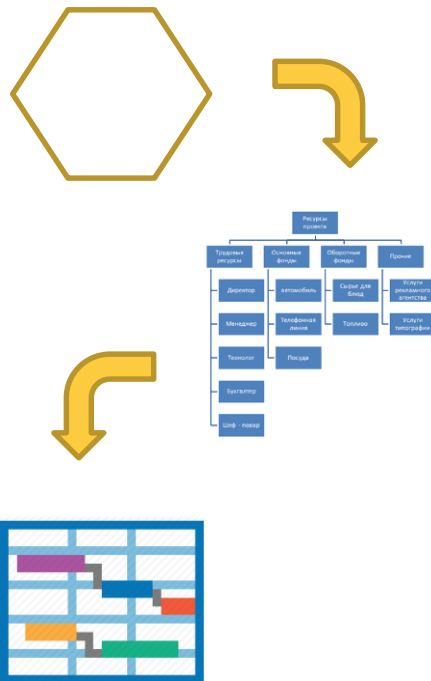
Kanban



Делают
по ролям

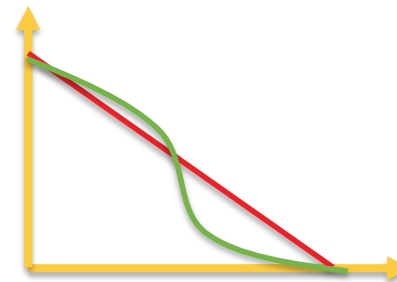
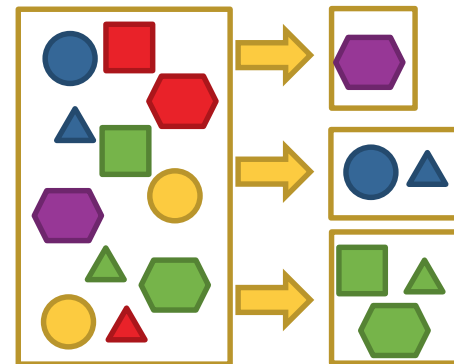
Сроки

PMI



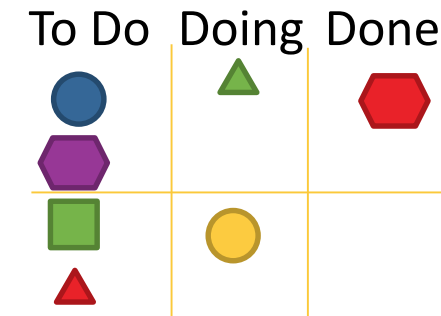
Различные способы оценки
сроков

Scrum



Оценка в storypoint'ax
Burndown chart

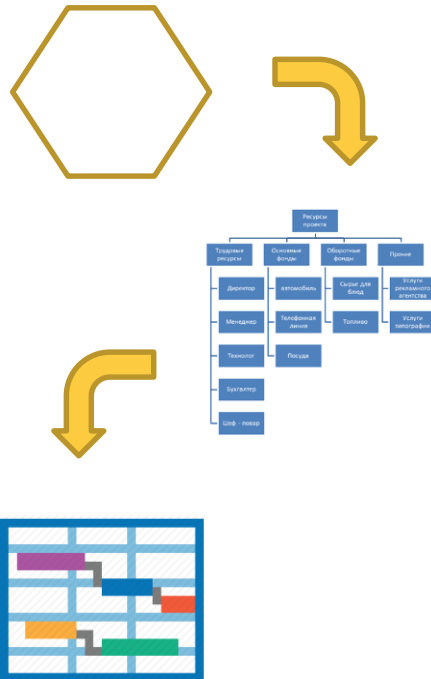
Kanban



TTM (time to market)
LT (lead time)
CT (cycle time)

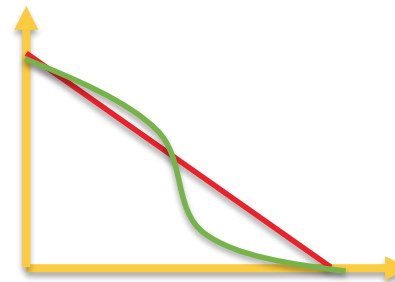
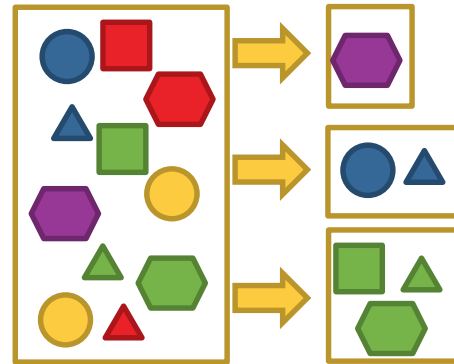
Стоимость

PMI



Различные способы оценки стоимости
Спросить поставщиков
Резервы на риски

Scrum



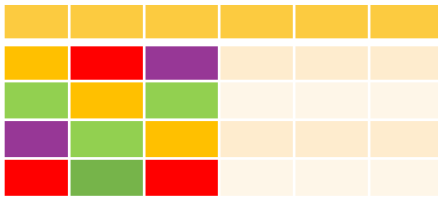
Себестоимость =
зарплата + налоги

Kanban

Специально
не смотрим

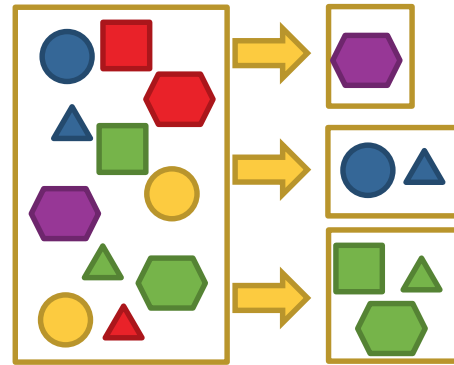
Риски

PMI



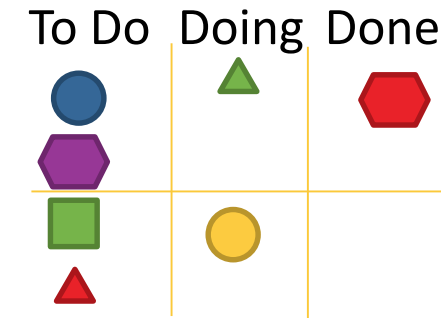
Реестр рисков и возможностей
Идентификация рисков
Качественный анализ
Количественный анализ
Планирование реагирования
Резервы
База знаний

Scrum



Специально не
смотрим, но
есть
ретроспективы

Kanban



Визуализация
узких мест

Качество

PMI

7 инструментов контроля качества:

- Диаграмма Ишикавы
- Диаграмма разбрасывания
- Диаграмма Парето
- Гистограмма
- Чек-лист
- Контрольная карта
- Блок-схема

QA и QC

- Хороши ли наши практики

Ответственность менеджера, вовлечена вся команда (TQM).
Отдельные роли могут быть сосредоточены на проверке результатов в интересах пользователей и заказчиков

Scrum

- Проверка владельцем продукта во время демо
- Ретроспективы похожи на работу с качеством

Командная ответственность + владелец продукта как адвокат пользователя

Kanban

Настройка ролей, входов и выходов

Команда стремится к тотальному обеспечению качества (TQM), менеджер помогает + отдельные роли могут фокусироваться именно на проверке результатов в интересах пользователей и заказчиков

Закупки

PMI

- Планирование закупки
- Осуществление закупки
- Контроль закупки
- Закрытие закупки

Типы контрактов

- Fixed Price
- T&M
- CR

Scrum

- Нужно договориться с внешней командой договориться работать по scrum
- Некоторые закупки не равны вовлечению внешней команды, а требуют долгого и тщательного планирования

Kanban

Не рассматривается

Прогресс

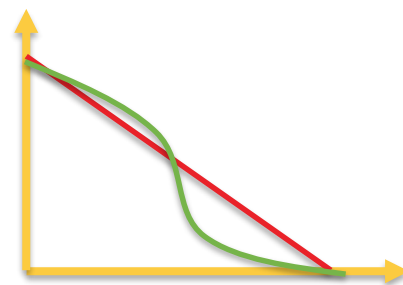
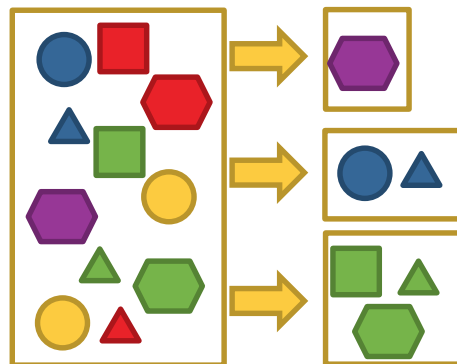
PMI



Различные способы
отслеживания планов

- Отчеты по запросы
- Правила заполнения
- Метод освоенного объема
EVA
 - SPI, SV, CPI, CV

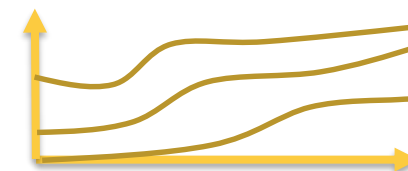
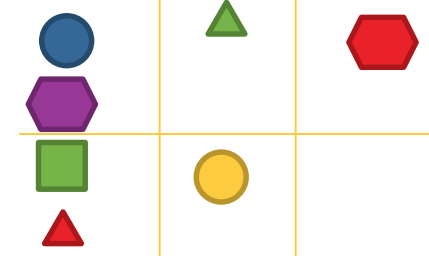
Scrum



Burndown chart
Burnupchart
Agile EVA

Kanban

To Do Doing Done



TTM (time to market)
LT (lead time)
CT (cycle time)
WIP (work in progress)
Кумулятивная
диаграмма потока

Интеграция

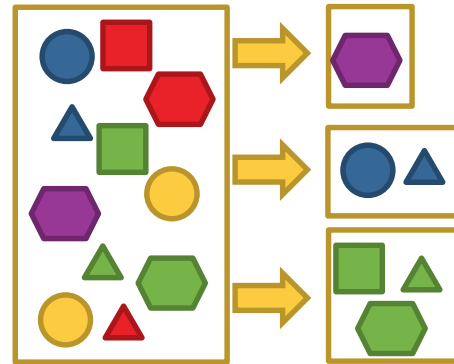
PMI



Менеджер проекта

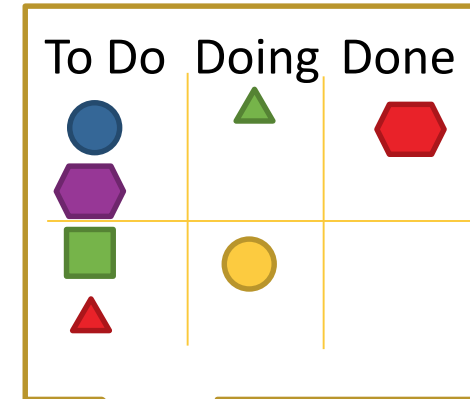
- интегратор
- формирует и удерживает команду специалистов
- контролирует связанность планов, их взаимное влияние и соответствие ограничениям

Scrum



- Команда сама организует работу в итерации согласно приоритетам владельца продукта
- Владелец продукта – интегратор в рамках продукта

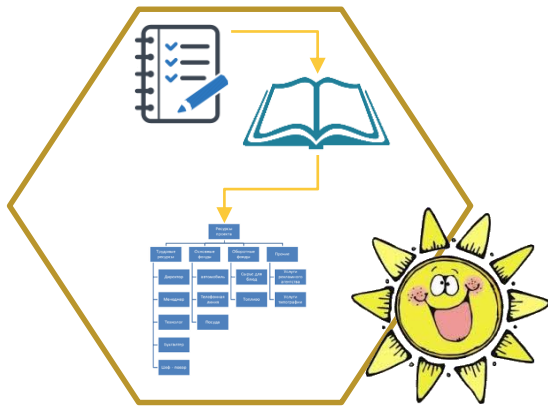
Kanban



Тот, кто совместно с командой придумал и настроил поток задач, тот, кто обеспечивает стык с другими методами и командами, часовщик

Заккрытие

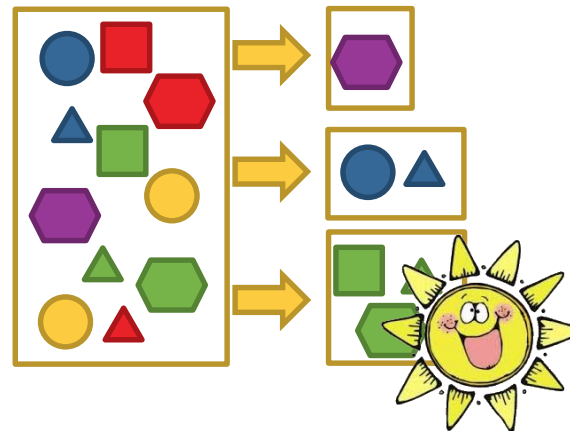
PMI



Сдача-приемка

- На всем протяжении проекта
- Формальное завершение проекта
- Высвобождение ресурсов и команды
- Фиксация Lessons Learned

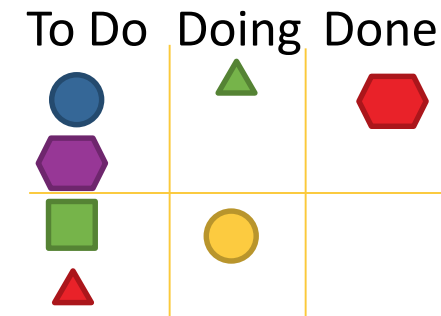
Scrum



После каждой итерации

- Демонстрация
- Ретроспективы

Kanban



Результаты работы по приоритетам

Сравнение Scrum и Kanban

	Scrum	Kanban
Описание задач	Прорабатываются заранее	Может изменяться
Длительность задач	По возможности одинаковые	Могут быть разные
Перенос задач	Задачи должны быть сделаны в рамках итерации	Выполняются столько, сколько требуется
Расписание встреч	Набор обязательных встреч	По необходимости
Ключевая метрика	Скорость выполнения задачи	Время выполнения задачи
Члены команды	Могут быть разрозненны	Сплочённый коллектив
Вид команд	Кросс-функциональные	Кросс- функциональные или профессиональные
Взятие задач в работу	Группой в начале итерации	В любой момент времени
Роли в команде	Жестко определены	Определены условно и по обстоятельствам
Ограничения	Задачи берутся в спринт, Количество не важно	Количество одновременно выполняемой работы на этапе ограничено
Этапы	Сделать, в работе, выполнено	Определяются для проекта индивидуально
Дедлайны	Соблюдаются на уровне задач в итерации	Большая ориентация на процесс

Для больших команд и организаций (справочно)

<http://www.agilescaling.org/ask-matrix.html>

<https://scrumtrek.ru/blog/enterprise-agility/1140/ask-matrix/>

Aspects / Criteria	Scrum-of-Scrums (SoS) PO meta-scrum	Large Scale Scrum (LeSS) Larman/Vodde	Scaled Agile Framework (SAFe) Leffingwell	Disciplined Agile Delivery (DAD) + Agility at Scale Ambler/Lines	Spotify "model" (Tribes, Squads, Chapters & Guilds) Kniberg	DSDM Drive Strategy Deliver More	Recipes for Agile Governance (RAGE)	Nexus / Scaled Professional Scrum Scrum.org	Scrum at Scale Sutherland/Brown
Description	An important mechanism that may be enough for smaller organizations but is not a full scaling approach	Larman / Vodde model as documented in "Scaling Lean & Agile Development"	The method documented by Dean Leffingwell and Scaled Agile, Inc.	Scott Ambler model documented in the book "Disciplined Agile Delivery"	The method used at Spotify, featuring Squads, Tribes, Chapters & Guilds.	DSDM is a robust Agile project management and delivery framework that delivers the right solution at the right time	Traditional-agile hybrid of portfolio-project planning	Scrum-based scaling with an "exoskeleton" called Nexus plus over 40 practices	Scrum Inc.'s modular framework for scaling Scrum
Web Link	guide.agilealliance.org/guide/scrumofscrums.html	less.works/	scaledagileframework.com/	disciplinedagiledelivery.com/	scaling-agile-spotify-11.pdf	www.dsdm.org/	www.cprime.com/tag/agile-governance	www.scrum.org/Resources/The-Nexus-Guide	www.scruminc.com/
What Team level frameworks are supported? (Scrum, Kanban, XP, etc.)	Scrum	Scrum	Scrum / Kanban / specific XP practices "mandated"	Scrum /Lean	Own method though partly Scrum-like	Own Hybrid Agile Scrum method	Scrum /Lean	Scrum	Scrum
Software centric - how often used outside of SW or IT?	Could use anywhere you use Scrum	Focused on Software or SW/HW	Focused on Software or SW/HW	Has been used outside of IT	Spotify only	Has been used outside of IT	Focused on Software	Could use anywhere you use Scrum	Could use anywhere you use Scrum
Big Positives / Key Differentiators	simple, standard Scrum focus on dependencies & resolutions	Good PO scaling; strong principle alignment, Non-prescriptive - gives "suggestions"	The "big picture" and completeness; getting Agile "in the door" at large corporations; actively evolving.	Lots of content; strong in areas such as architecture, design and dev ops; incorporates many good models.	very agile, entrepreneurial, distributed teams, low overhead	Very established following in the UK	Fluid and adaptive	Authored by Ken Schwaber	Lightweight authored by Jeff Sutherland
Key Risks / Concerns	limited scaling, limited documentation, not clearly defined Not likely "sufficient" for large scale; some differences in implementation	A more "radically agile" approach that may be a hard sell in larger traditional orgs with many layers and specializations.	Little info on "how", most need certified SPCs to implement properly; Seen as prescriptive; not "agile enough" in its structures; "quick start and leave" issues some places	vague in some areas about the "how"; can come across as a bit disjointed. Not prescriptive in lifecycle.	very limited detail about the "how", Not really a framework; may only fit certain cultures	Heavy process overhead.	New approach that is growing and adapting.	New approach that is growing and adapting. Some of the parts are "secret" unless you go to the class.	New approach that is growing and adapting.
Training / Resource availability	None known; roll your own	Training and coaching network available	Yes, multi-level training & Certifications	Yes, multi-level Certifications	None known; roll your own	Yes, multi-level Certifications	None known read and implement yourself for your need.	Scaled Professional Scrum training & certification is available	Courses are offered



Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Тема 6. Приоритеты и оценки задач и проектов; метрики процесса и продукта в гибком подходе



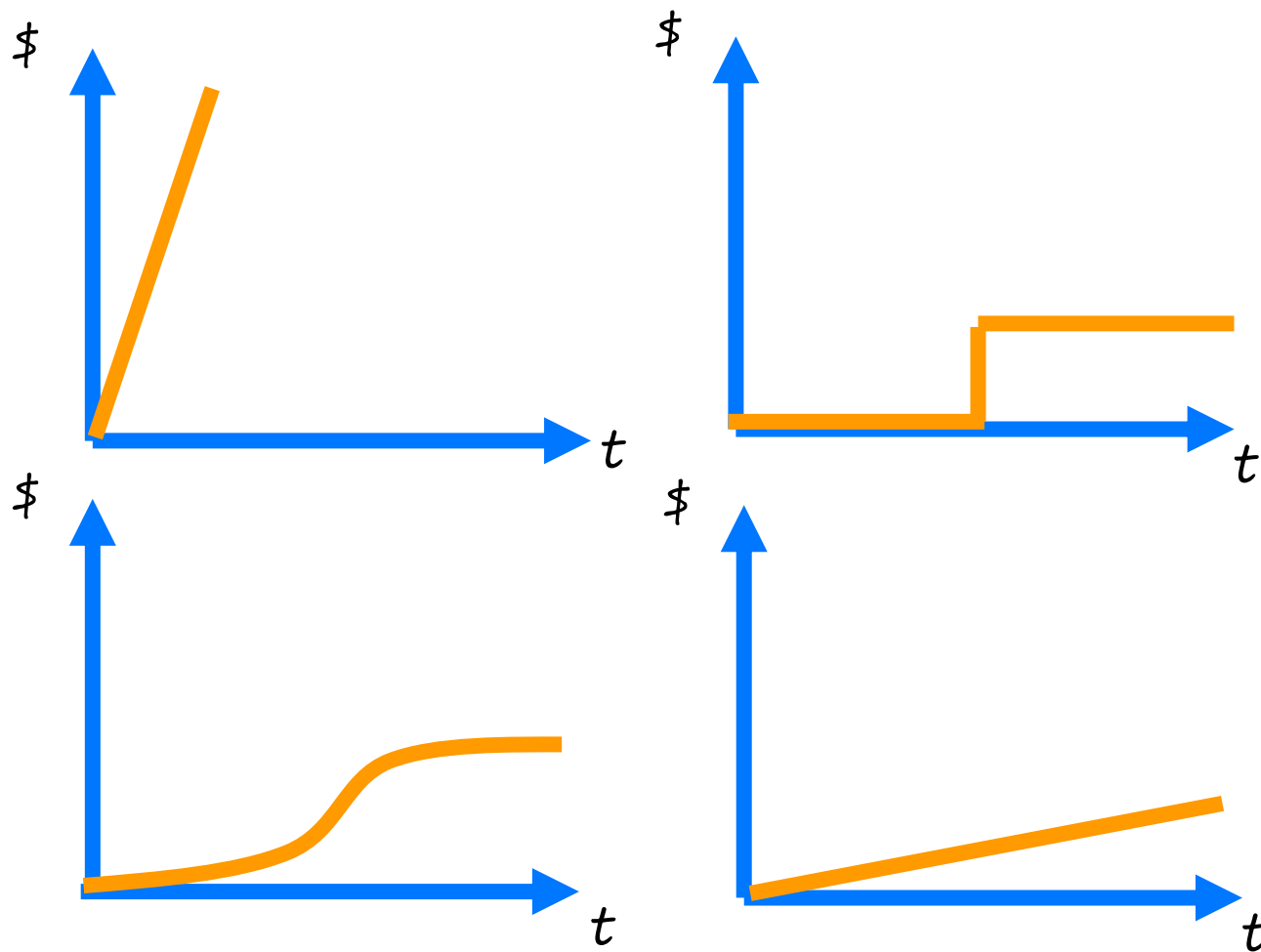
**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Приоритеты

Метод MoSCoW

- **M** – must – самый высокий приоритет, без которого продукт не будет сдан, проект не будет выполнен
- **S** – should – важные требования, не имеющие решающего значения, но обязательные к исполнению
- **C** – could – желательно сделать
- **W** – would – наименее критичные требования

По стоимости и времени



Метод футболки

приоритет = ценность - затраты

	Очень большие затраты (7)	Большие затраты (6)	Средние затраты (1)	Малые затраты (0)
Ценность очень большая (7)	0	1	6	7
Ценность большая (6)	-1	0	5	6
Ценность средняя (1)	-6	-5	0	1
Ценность малая (0)	-7	-6	-1	0

Weighted Shortest Job First. Сначала
наивесомейшая кратчайшая работа
 $WSJF = \text{ценность} / \text{затраты}$.

Название	Ценность, \$	Затраты, ч	WSJF	Номер в очереди
Задача 1	10000	160	63	10
Задача 2	20000	150	133	5
Задача 3	40000	300	133	4
Задача 4	50000	400	125	6
Задача 5	25000	70	357	1
Задача 6	1000	10	100	9
Задача				

GIST + ICE

<https://habr.com/ru/company/yandex/blog/434>

$$Score = \frac{Impact \times Confidence}{Effort} + K$$

Goals — цели организации/подразделения. Могут быть сформулированы в формате OKR.

Ideas — бесконечно пополняемый банк идей. Это предположения, как продвинуться к целям.

Step-projects — проекты для развития идей.

Tasks — детализация на уровне задач в трекере.

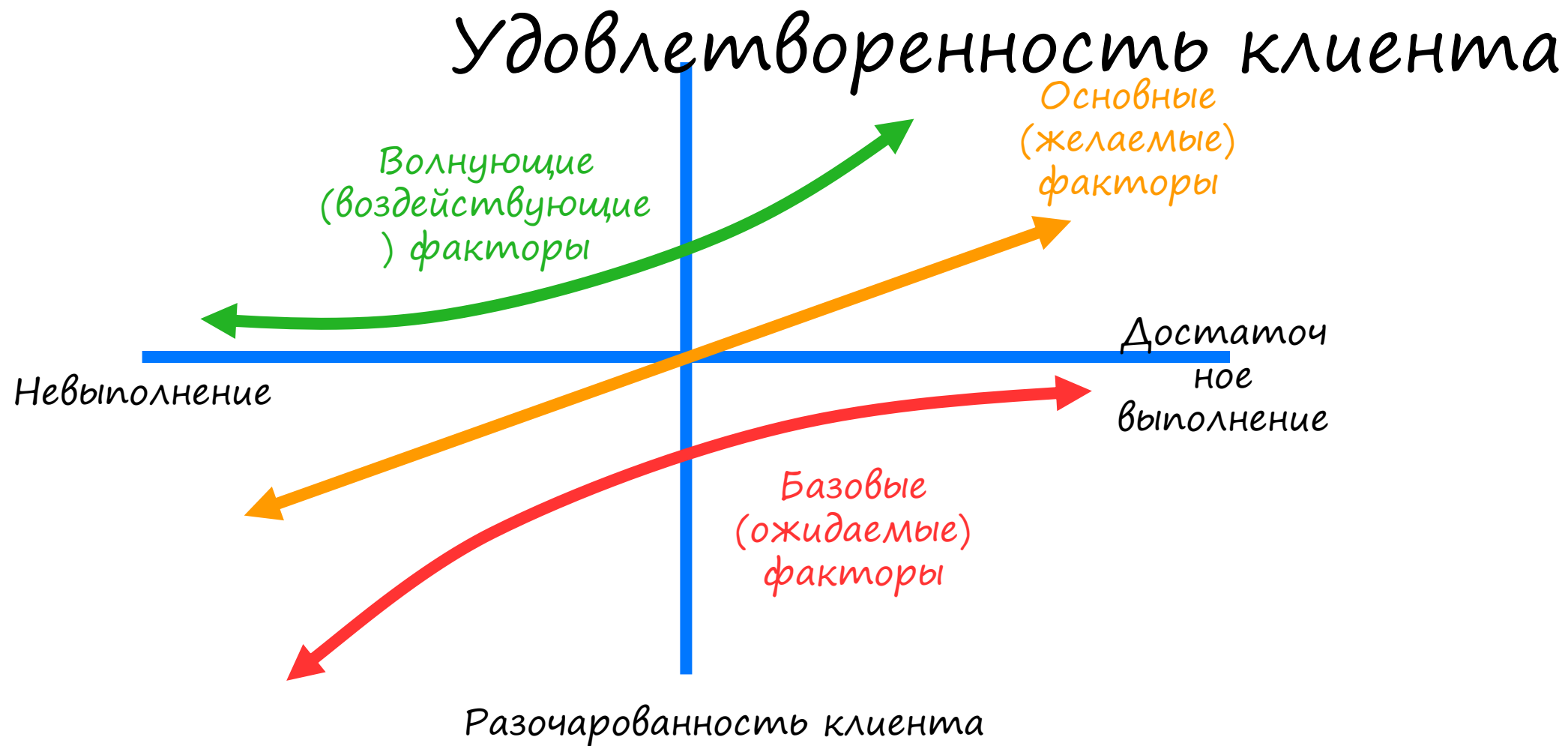
Impact — сколько бизнесу принесет эта фича.

Effort — затраты в человеконеделях.

Confidence — сколько подтверждений этой идее мы накопили.

K* — некоторые проекты мы делаем не из прямых бизнес-соображений, а, например, чтобы избежать репутационных издержек или если эти работы еще не согласованы с целями компании.

Метод КАНО



Метод КАНО

(пульт от телевизора, как пример)



То, что должно
быть:

- Переключатель каналов (цифры)
- Pre-Ch (предыдущий канал)
- Переключатель каналов +1
- Выключение ТВ
- Выбор источников трансляции
- Управление меню
- Выключение звука
- Громче/тише



То, что привлекает:

- Sleep
- P-mode
- P-size
- Стоп-кадр
- Эффекты изображения
- Вход в меню
- Выход из меню
- Список каналов



То, вызывает
Вау-эффект:

- Стиль ipod classic
- Сенсорный экран
- Возможность голосовать



Storymapping

Формирование карт
пользовательских историй на
примере почтового менеджера

Отправит
в письмо

Почтовый
менеджер

Обновить
встречу

Поиск по
нескольким
полям

Отправит
Отправит
в письмо с
вложением
письмо



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

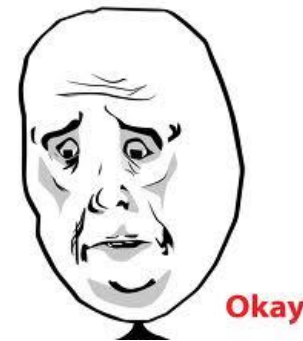
Оценки

Что такое хорошая оценка? Хорошая оценка – это такая оценка, точность которой $\pm 10\%$

Хорошая оценка – это такая оценка, которая находится внутри $\pm 25\%$ реального срока в 75% случаев.

Типичная ситуация

- Шеф:
 - Данный модуль должен быть сделан через три месяца, к первому сентября, вот список функций. Через сколько времени модуль будет готов?
- Работник:
 - Если у меня не будет проблем с разработкой, пауз и других непредвиденных обстоятельств, то я смогу его закончить через 6 месяцев
- Шеф:
 - Но нам нужен этот модуль к первому сентября
- Работник:
 - Ок...



Что на самом деле

- Цель – выпустить версию к 1 сентября
- Оценки смешиваются с целями: «сделать через 3 месяца»

Что в итоге?

- С большой долей вероятности модуль не будет сделан к сроку, как бы ни старались, просто физически не хватит времени, даже если работать круглосуточно
- Как следствие – недовольный шеф
- Если вдруг проявили героизм и сделали, то героем никто никого считать не будет

Что делать?

- Спросить шефа о том, почему так критично 3 месяца (может быть, будет выставка и нужно демо)
- По возможности адекватно оценить каждую функцию отдельно, выстроить их по приоритетам и выбрать те, которые нужны в первую очередь
- Не бояться шефа и не подгонять свои оценки под его цели, иначе будет еще хуже

На что обращать внимание

- Оценки – это не точные вычисления сроков, трудоемкости, объемов работ, а именно оценки. Это приблизительные значения, которые могут находиться в довольно широком диапазоне
- Вычисления могут дать более точные значения, но они так же могут не учитывать множество факторов и аналогично определяются в диапазоне

Оценки нулевого порядка (zero order или raw estimations)

- Иногда нет возможности уделить оценке хотя бы полчаса, например, в случае, если кто-то спрашивает о том, стоит ли вообще браться за некий проект и сколько это может теоретически занять
- В этом случае используются raw estimations, которые могут дать представление только о порядке величины трудоемкости (неделя, две, месяц, полгода...)

Что влияет на оценку любого проекта?

Команда не готова приступить к работе вовремя

Новые требования добавляются в проект, предыдущие требования устаревают.

Команда отвлекается на задачи в других проектах

У команды меньше опыта, чем планировалось

Поощряются «добровольные» овертаймы

Проблемы традиционных оценок

Политические требования внутри компании

«Торги» за выгодные оценки

Оценками занимаются некомпетентные специалисты

Конус неопределенности игнорируется

Никому не нравится оценивать заново!

Для того, чтобы сделать оценку, нужны метрики,
а не догадки. Но обычно метрик у нас нет.



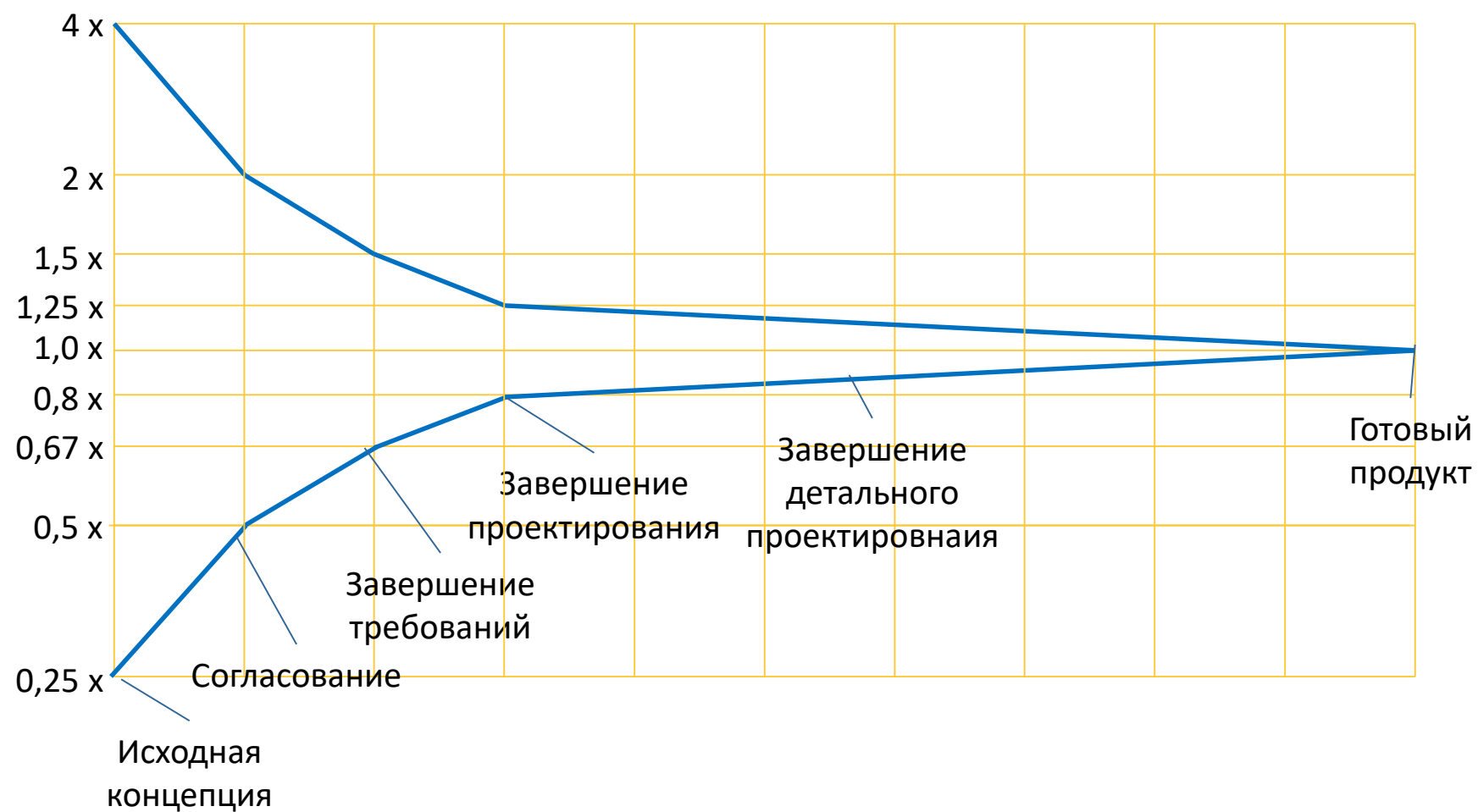
Почему мы совершаем ошибки?

Потому что нам важно сохранять оптимизм. Как минимум, чтобы убеждать окружающих в правильности своей оценки:)

Часто нам хочется сделать приятно нашему клиенту или руководителю оптимистичной оценкой.

Никому не нравится думать о гадких вещах, вроде локализации или инфраструктуры.

Помните о конусе неопределенности



Не секрет, что продуктивность лучших команд может быть от 10 до 25 раз лучше, чем продуктивность хороших команд.



Неопытная команда, лишенная менторства, отсутствие согласованных рабочих практик прозрачного процесса менеджмента изменений могут привести ваш проект к быстрой и болезненной смерти. Или, напротив, к длительному разложению.

Недостаток вовлеченности конечных пользователей, заказчиков и заинтересованных в проекте лиц, – и вы можете обнаружить, что никто, кроме вас не считает проект завершенным.

Порочный процесс оценки

Задачи, которые не попали в оценку, но все равно существуют в проекте

Важные фоновые задачи игнорируются: настройка инфраструктуры, как самый частый пример

Учитывается только время на непосредственное выполнение задач, но не на коммуникацию и обсуждения

Вы слышали эти слова?

«Мы будем более эффективны и продуктивны, чем предыдущая проектная команда!»

«Много вещей пошло наперекосяк на нашем последнем проекте. Впредь это не повторится.»

«Команда, работавшая на последнем проекте – просто идиоты. Мы гораздо умнее.»

Как не перепутать оценку и планирование?

Оценка должна быть непредвзятой и аналитической

Планирование – это агрессивный целеориентированный процесс

Планы часто выдаются за оценки

И даже хуже – часто планы конвертируются в расписание проекта

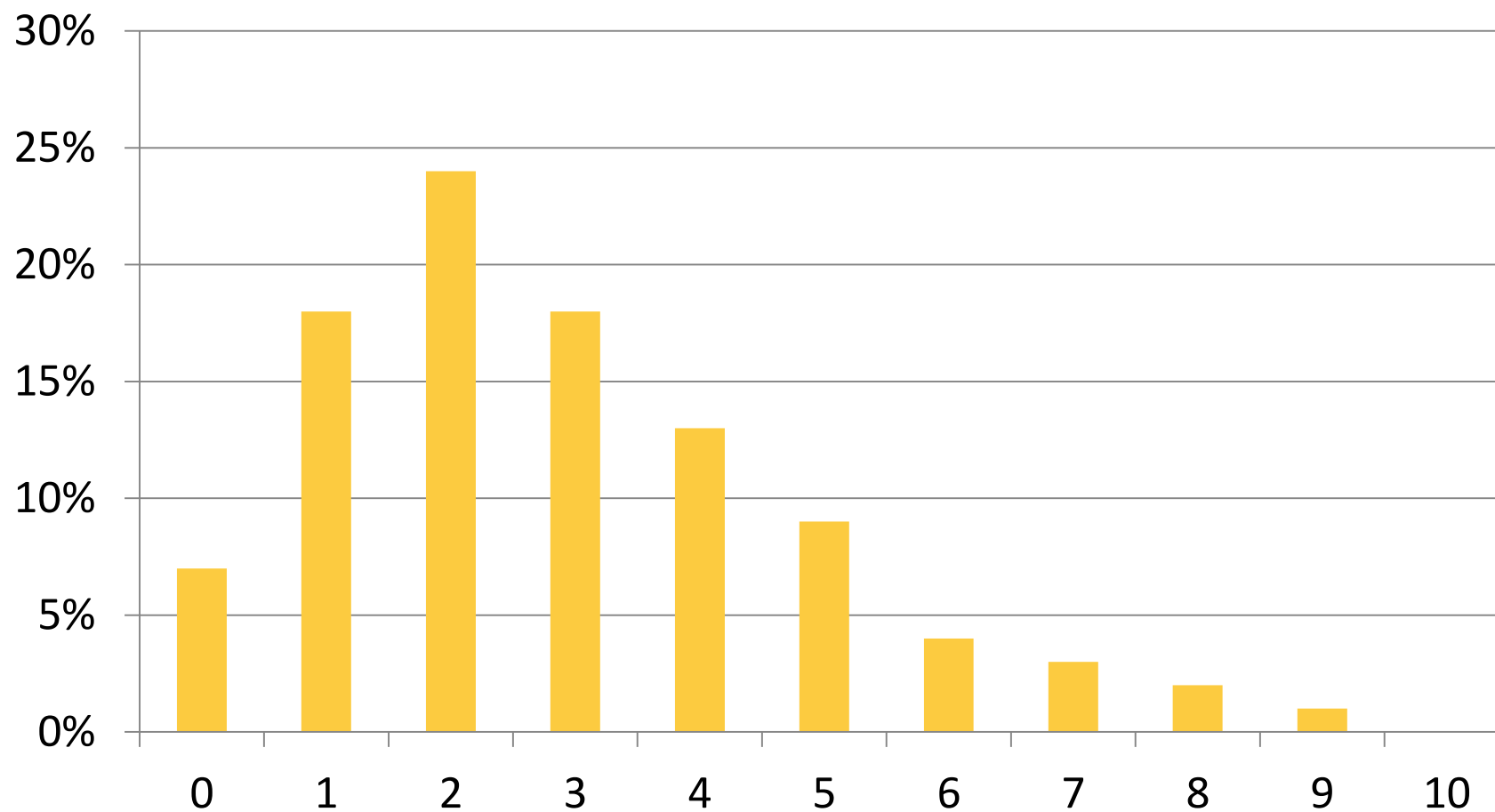
Тест по оценке

- Площадь Якутии
- Год рождения Александра Македонского
- Кассовые сборы “Крестного отца”
- Мощность всех АЭС России
- Объем озера “Байкал”
- Вес самого тяжелого кота
- Географическая широта Омска
- Наибольшая глубина Черного Моря
- Вес самого легкого автомобиля
- Минимальное расстояние от Земли до Марса

Тест по оценке

- | | |
|--|-----------------------------|
| • Площадь Якутии | • 3 083 523 км ² |
| • Год рождения Александра Македонского | • 356 год до н.э. |
| • Кассовые сборы “Крестного отца” | • \$243 798 921 |
| • Мощность всех АЭС России | • 23 643 МВт |
| • Объем озера “Байкал” | • 23 615,39 км ³ |
| • Вес самого тяжелого кота | • 21 кг |
| • Географическая широта Омска | • 54° 59' 00'' с.ш. |
| • Наибольшая глубина Черного Моря | • 2 212 м |
| • Вес самого легкого автомобиля | • 9,5 кг |
| • Минимальное расстояние от Земли до Марса | • 55,76 млн км |

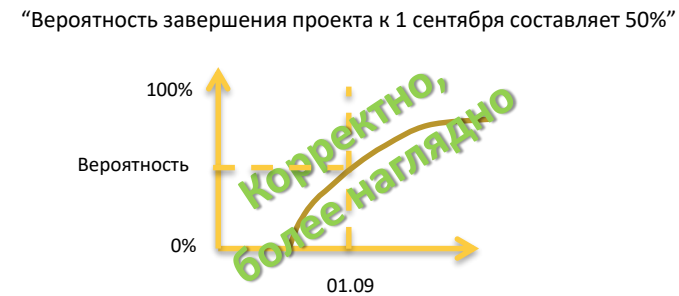
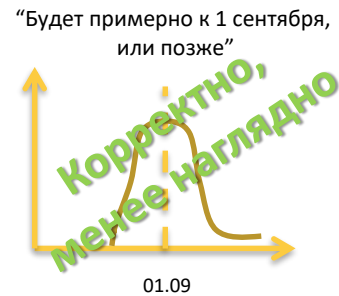
Результаты



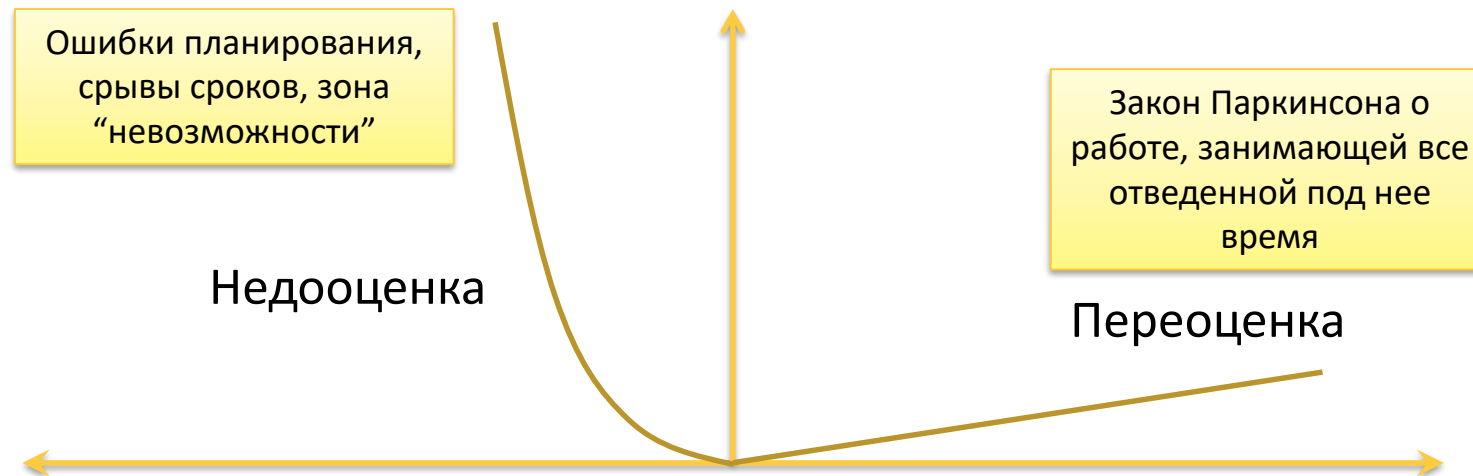
ВШУП НИУ ВШЭ

234

Если мы говорим о вероятности завершения проекта



Недооценка и переоценка



Принципы оценок в agile проектах

Относительные оценки

Коллективные оценки и обсуждение, planning poker

Частые переоценки (помним о конусе неопределенности)

Оценки отдельных небольших задач, а не проекта, спринта или большой пользовательской истории в целом

Подсчет, вычисление, экспертная оценка

- Те самые задачи про то, сколько шариков от пинг-понга нужно, чтобы заполнить автобус, или за какую сумму денег можно взяться перемыть все окна в небоскребах города...

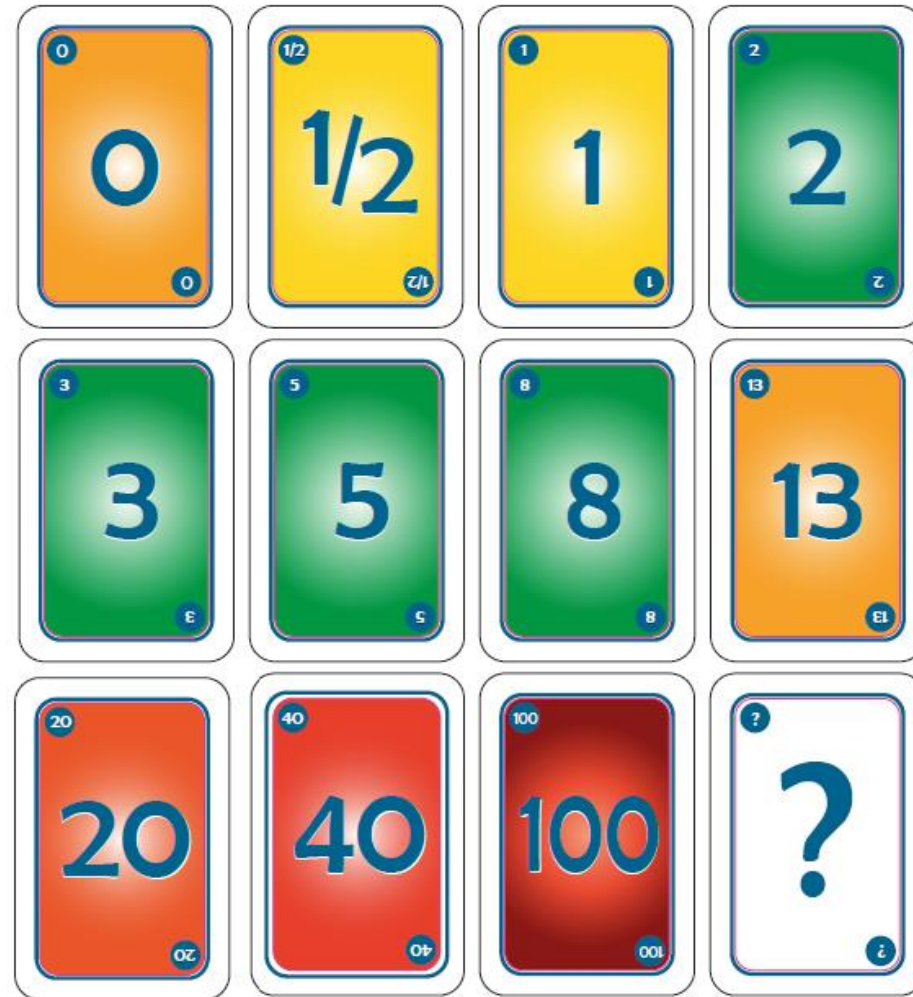
Калибровка

- Если есть возможность отслеживаем, сколько времени мы реально потратили на ту или иную задачу, и корректируем себя
- Сравниваем с историческими данными
- Сравниваем со средними данными по отрасли

Оценка по аналогии

- Если подобный проект уже когда-либо был реализован, то можно посмотреть на WBD и бэклоги для него, вычленив характерные части и использовать в новых оценках

Planning poker





**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Прямые оценки

Оценка через WBD

Единая оценка

- По моим оценкам проект займет 22 недели
- Реально заняло 29 недель
- Погрешность оценки всего 24%

Оценка через WBD

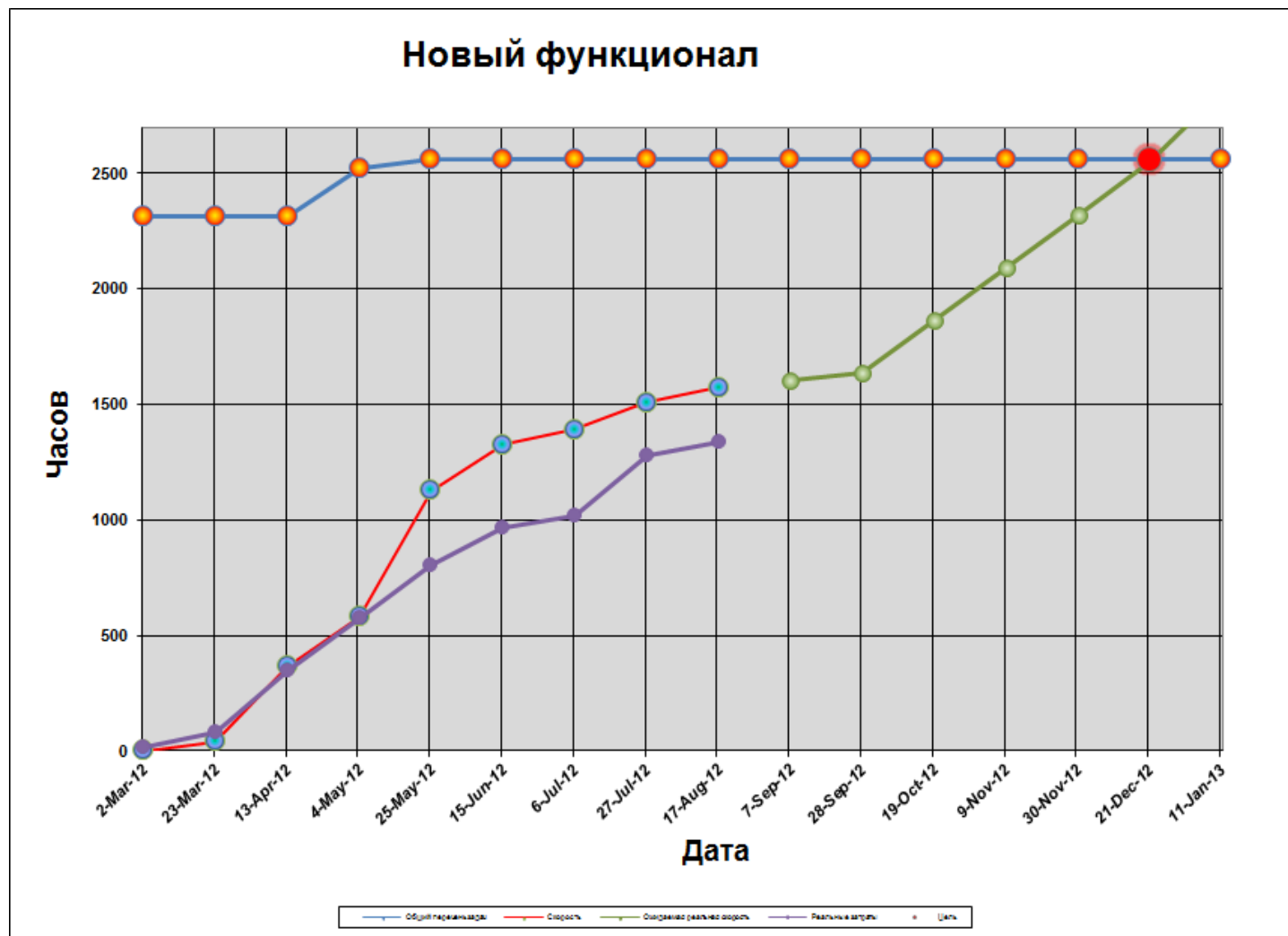
№	Оценка	Факт	Δ	δ
1	1.50	3.00	-1.50	50%
2	4.50	2.50	2.00	80%
3	3.00	1.50	1.50	100%
4	1.00	2.50	-1.50	60%
5	4.00	4.50	-0.50	11%
6	6.00	4.50	1.50	33%
7	2.00	3.00	-1.00	33%
8	1.00	1.50	-0.50	33%
9	3.00	2.50	0.50	20%
10	1.50	3.50	-2.00	57%
Итого	27.50	29.00	-1.50	
Ср.			-5%	48%

Триангуляция (PERT)

№	Лучший случай	Наиболее вероятный случай	Худший случай	Ожидаемый случай	Факт	MRE	Попали в диапазон
1	1.60	2.00	3.00	2.10	3.00	30%	Да
2	1.80	2.50	4.00	2.63	2.50	5%	Да
3	2.00	3.00	4.20	3.03	1.50	102%	Нет
4	0.80	1.20	1.60	1.20	2.50	52%	Нет
5	3.80	4.50	5.20	4.50	4.50	0%	Да
6	3.80	5.00	6.00	4.97	4.50	10%	Да
7	2.20	2.40	3.40	2.53	3.00	16%	Нет
8	0.80	1.20	2.20	1.30	1.50	13%	Да
9	1.60	2.50	3.00	2.43	2.50	3%	Да
10	1.60	4.00	6.00	3.93	3.50	12%	Да
Итого	20.00	28.30	38.60	28.63	29.00		70% Да
Среднее				$\delta = 2,5\%$		24%	

Оценка по PERT: $(\text{Лучший} + 4 * \text{Вероятный} + \text{Худший}) / 6 = \text{Ожидаемый}$

Метод освоенного объема в гибких методологиях (Agile EVM)





Метрики

Нельзя улучшить то, что нельзя измерить

Зачем нужны метрики?

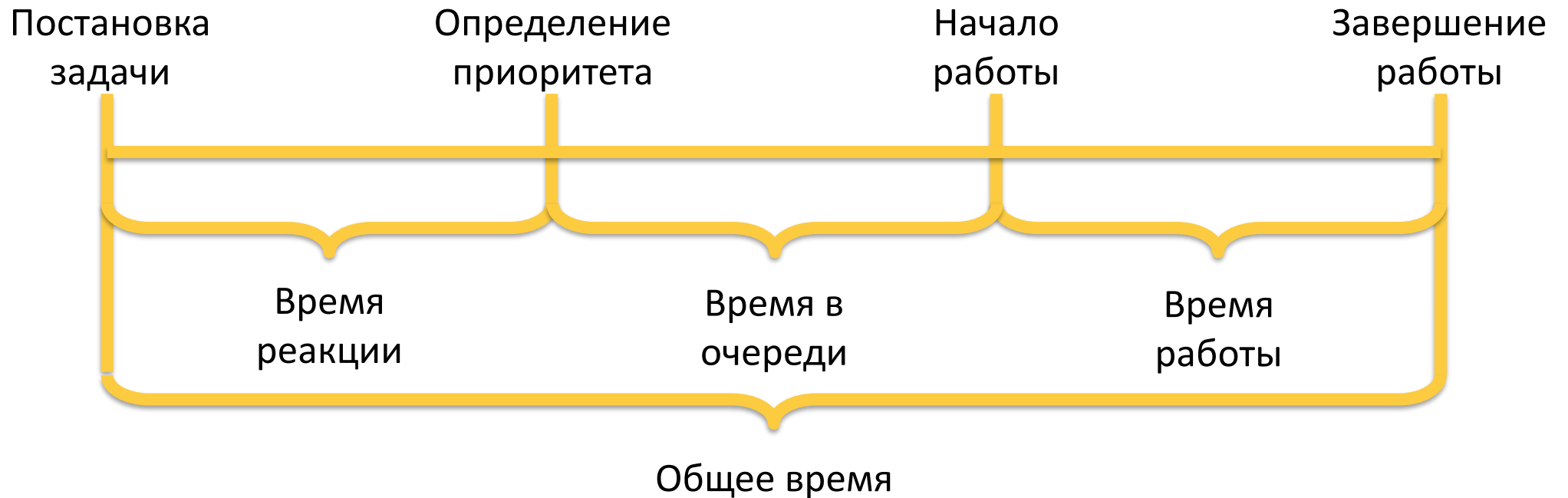
Быстро провалиться и не потерять деньги – это важно в инновационных проектах, где нужно проверить бизнес-гипотезу

Идентифицировать риски на ранней стадии проекта

Чтобы хорошо спланировать ваш проект и не оказаться на бирже труда:)

Пример измерений

- Lead time – время между тем, когда задача была поставлена и тем, когда клиент увидел ее в конечном продукте
- Cycle time – время между тем, когда была начата работа над задачей и тем, когда клиент увидел ее в конечном продукте.



При работе над улучшением lead time и cycle time, важно не подорвать уровень качества в проекте. Не забывайте отслеживать метрики качества, индивидуальные для вашего проекта.



Стремитесь быть предсказуемым и скучным:)

Для этого вам понадобится отслеживать скорость команды.

А еще полезно отслеживать:

- блокеры, преодоленные в течение итерации;
- блокеры, перенесенные на следующую итерацию;
- пользовательские истории, реализованные за итерацию;
- пользовательские истории, вынесенные за пределы итерации;
- дефекты продукта, вынесенные за пределы итерации

Как начать работать с метриками?

Обязательно вовлеките команду и утвердите все ключевые метрики вместе

Помните, что метрики собираются не ради коллекции, а ради улучшения вашего процесса

Никогда не используйте метрики в качестве орудия «линчевания» команды за плохую работу

Создайте ваш собственный дашборд метрик, который станет показателем здоровья проекта



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Тема 7. Гибкие команды. Управление людьми

Принципы TQM

1. Установите постоянство цели
2. Примите новую философию качества
3. Покончите с зависимостью от массового контроля
4. Покончите с практикой закупок по самой низкой цене
5. Улучшайте каждый процесс
6. Введите в практику современные подходы
7. Учредите «лидерство»
8. Изгоняйте страхи
9. Разрушайте барьеры между подразделениями
10. Откажитесь от использования пустых лозунгов
11. Устраните произвольно установленные задания и количественные нормы
12. Дайте работникам возможность гордиться своим трудом
13. Поощряйте стремление к образованию и совершенствованию
14. Вовлекайте всех сотрудников в преобразование организации

Гибкие команды

- Многофункциональные
- Самоорганизующиеся
- Практикуют коллективное владение продуктом
- Ориентированы на обучение

Подбор людей в команду

- Ищите универсалов (T-shaped person)
- Людей, не боящихся неопределенности
- Командных игроков, умеющих подавлять свое эго

Ложные цели управления людьми

- Автоматизация всех процессов
- Полная загрузка всех сотрудников в течение всего времени
- Получение максимального числа человеко-часов
- Стандартизация процедур

Некоторые принципы управления гибкой командой

- Сверхурочные не существуют
- Позвольте людям качественно выполнять свою работу
- Закон Паркинсона не работает

Еще про заблуждения

- Существует еще не изученный трюк, который поднимет производительность до небес
- Другие руководители умудряются получить прирост производительности в сто, двести и более процентов
- Технологии развиваются с такой скоростью, что невозможно за всем успевать
- Из-за отставания следует удвоить производительность
- Все уже автоматизировано; Не пора ли нам автоматизировать разработчиков?
- Люди будут лучше работать, если на них как следует надавить

Создайте комфортные условия

- Создайте условия для общения команды
- Позвольте команде организовать свои рабочие места
- Устраните шум (включая телефонные звонки)
- Пусть будет помещение с окнами



Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами

Тема 8. Как начать применять гибкие практики на уровне организации

Готовность меняться — это огромная сила. Даже если это значит, что часть в компании будет пребывать в полном смятении какое-то время.

Джек Уэлч

Почему переход к гибким методологиям сложен?

Успешные изменения не могут быть произойти только снизу вверх или только сверху вниз

Конечный результат сложно предсказать

Гибкие методологии первазивны и всепроникающи. Остаться в стороне сложно и не всем это по душе

Изменения происходят гораздо быстрее, чем когда-либо раньше

Лучшие практики опасны

Почему же переход стоит сил?

Лучшая продуктивность и меньшие затраты

Улучшенная вовлеченность сотрудников и удовлетворение работой

Продукт попадает на рынок быстрее

Качество улучшается

Улучшается удовлетворенность спонсоров и заинтересованных в проекте лиц

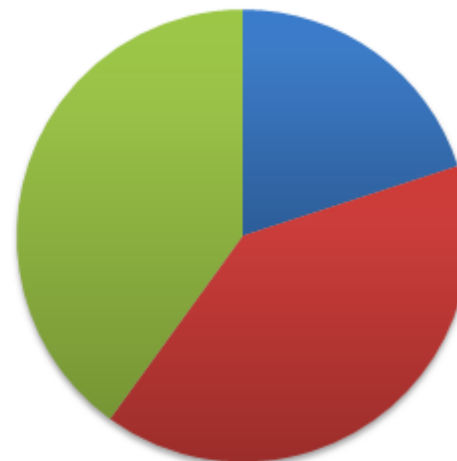
То, что мы делали до этого уже не работает:)

Доля принимаемых решений

Традиционный подход



Гибкий подход



■ Скрам мастер ■ Владелец продукта ■ Группа

Gustavsson, T., 2011. Agil projektledning. First edition.
Stockholm, Sweden: Sanoma Utbildning AB.

Что нужно для старта. Модель Adapt

Awareness – осознание и информированность о том, что текущий процесс не приносит хороших результатов

Desire – желание применять гибкие методологии как метод решить существующие проблемы

Ability – возможность преуспеть в реализации новых подходов

Promotion – продвижение идей гибкого подхода внутри компании, чтобы видеть успехи и перенимать опыт

Transfer – соучастие во внедрении agile повсеместно в компании

Что мешает информированности?

Отсутствие «большой картины» или невозможность ее увидеть

Отказ видеть то, что находится прямо перед собой

Смешение понятий «движение» и «прогресс»

Влияние собственной пропаганды «У нас все отлично!»

Как развить информированность?

Не стесняйтесь говорить о существовании проблемы

Используйте метрики

Привносите в процесс новый опыт и новых людей

Запустите пилотный проект

Сфокусируйтесь на самых важных причинах изменений

Как усилить желание изменений?

Информируйте окружающих о том, что есть лучший процесс и способ вести проекты:)

Создайте ощущение срочности

Работайте с теми, кто уже готов к изменениям

Уговорите команду провести agile тест драйв

Работайте со страхами

Не дискредитируйте прошлое

Как усилить вашу ВОЗМОЖНОСТЬ преуспеть?

Консультируйте и проводите тренинги

Назначайте ответственных

Делитесь информацией

Ставьте разумные цели

И сделайте это в конце концов!

Экспериментируйте, будьте терпеливы и готовы совершать ошибки.

Источники организационной гравитации

HR

Администрация

Маркетинг

Финансы

Почему стоит начать с малого?

Начать внедрение с маленькой команды гораздо не так дорого

При правильном подходе ранний успех практически гарантирован

Начиная с малого, у вас есть право на ошибки и много вторых шансов

Начинать с малого – это куда меньший стресс

Для такого старта вам не нужна полная реорганизация

Почему стоит начать сразу всем?

Всеобщее решение о переходе к гибким методологиям существенно уменьшает сопротивление

У вас не будут возникать проблемы на фоне того, что agile команды и команды, работающие по классическим подходам, плохо стыкуются

Всеобщий переход закончится куда быстрее, чем постепенное переползание

Как начинать всем вместе?

Бэклог улучшений на уровне компании

Создание сообщества, ответственного за успех перехода

Создание локальных групп по улучшению процессов

Выбор пилотного проекта

4 атрибута успешного пилотного проекта

Срок – медиана среднего проекта в вашей организации

Размер – проект может сделать одна команда, без массового привлечения других команд

Важность – выберите важный проект. Простой и низкоприоритетный проект не даст команде нужной мотивации и неубедителен для окружающих

Вовлеченность спонсора проекта

Подбор идеальной команды для «пилота»

В вашем проекте важно найти правильную комбинацию следующих людей:

Евангелисты agile – включайте в ваш проект как можно людей, которые готовы помогать словом и делом

Усердные оптимисты – признают необходимость изменений, но не так много знают об agile, чтобы пропагандировать его

Справедливые скептики – вам не нужен саботер, но включение в команду уважаемого и адекватного скептика может быть очень полезно. При успехе он станет вашим самым сильным евангелистом.



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Тема 9. Целеполагание

Почему это сложно?

- Потребитель не знает, что он хочет – **цели могут быть не определены**
- Требования могут меняться в любой момент - **цели могут меняться**
- У исполнителей нет понимания того, что они делают - **цели могут быть плохо донесены до команды**
- С расписанием может произойти все, что угодно
- С командой тоже
- Да и с продуктом
- Лучшее враг хорошего
- Каждой практике должно быть свое место и время
- И, вообще, все труднопредсказуемо

Какие проблемы актуальны?

- **Непонимание общих целей** ограничивается команды в возможностях реализации собственного потенциала. Это делает их просто исполнителями
- **Разобщенность целей** приводит рассинхронизации работы подразделений или даже к внутрикорпоративной «войне». Левая рука не ведает, что творит правая.
- **Разные трактовки результата** приводят к непониманию между отделами. Одни думают, что успех — это когда подписали договор с партнером. Другие, что это существенный рост заказов от нового партнерства.
- **Непрозрачность вызывает беспокойство** у команд, стейкхолдеров, акционеров.

Зачем нужно?

- Фокусировка и синхронизация внутри команд
- Движение команды в нужном для компании направлении
- Возможность объективной оценки результата
- Прозрачность для стейкхолдеров

Из чего состоит и как устроена цель в формате OKR?

OKR = Objectives & Key Results

O — objective, цель. Формулировка цели. Пример: «Запустить новую версию продукта»

KR — key results, ключевые результаты, критерии достижения. Мы договариваемся, что считается достижением цели. Обычно ключевых результатов

Вопрос

Запустить третью версию продукта

- Получить 1000 новых регистраций
- Получить 15 публикаций о продукте в медиа
- Достичь конверсии более 30%

Что здесь О, что KR?

Ответ

О: Запустить третью версию продукта

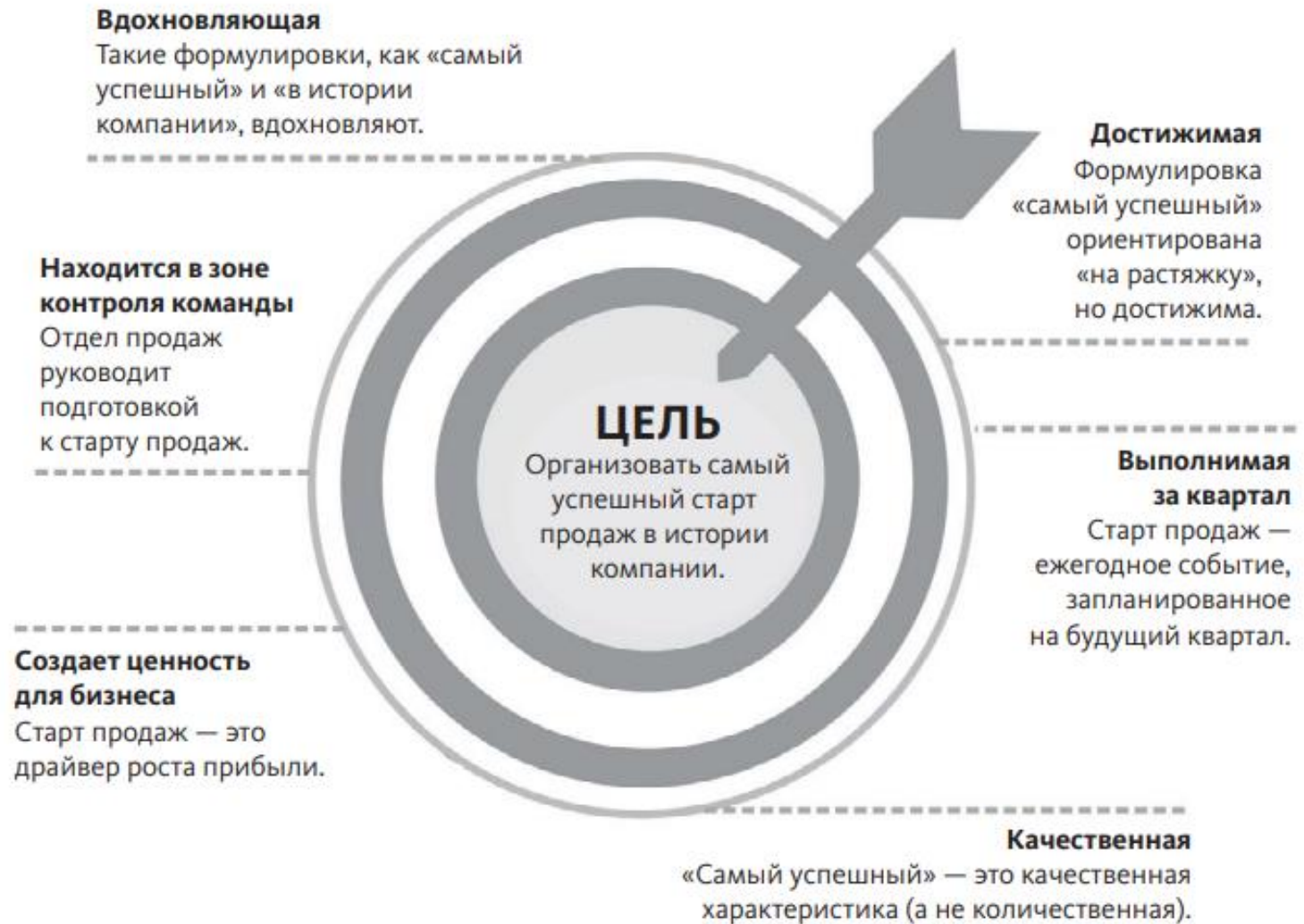
- KR1: Получить 1000 новых регистраций
- KR2: Получить 15 публикаций о продукте в медиа
- KR3: Достичь конверсии более 30%

Свойства OKR

1. Цели/Objective часто максимально амбициозные и труднодостижимые. Для таких выполнение KR на 80% - уже успех
2. Бывает цель «просто сделать». Ее KR надо выполнить на 100%. Ребрендинг или сделан в нужную дату или не сделан. Дом построен или нет.
3. 1-3 цели на команду, 2-5 ключевых результатов для каждой цели. Если больше, то это уже потеря фокуса.
4. Публичность – все сотрудники видят цели всех и каждого на каждом уровне. Это необходимо для поиска точек соприкосновения.
5. Вертикальная связность целей – фокусировка команды на действительно важных вещах. Формирование связанного дерева целей.
6. Измеримость результатов. В абсолютных (числа) или относительных (проценты) значениях.
7. Достижение OKR и премирование – не связанные вещи.

Пример цели

- Уточняющие вопросы
- Позитивная лексика
- Простота
- Начало с глагола



- Ключевые, а не все подряд
- Результаты, а не задачи
- Позитивная лексика
- Просто и ясно
- Поиск наилучшего
- Назначение ответственных



Что еще важно

- Цели должны вдохновлять и мотивировать.
- Большая часть целей должна формироваться снизу вверх, а не наоборот.
- Цели не должны диктовать решения, а скорее ставить проблемы.
- Цели не должны ограничиваться целями бизнеса, но затрагивать и личное развитие.

OKR vs KPI vs MBO?

OKR

- Цели и ключевые результаты на квартал.
- Вначале отвечаем на вопрос, «зачем мы это делаем».
- Направление движения на коротком промежутке времени.
- Это про исследование, поиск новых путей и точек роста.
- Поэтому ключевые результаты по умолчанию очень высоки, иначе зачем все это затевать.
- Не используется для наказания и поощрения сотрудников
- Пример: О стать топ-1 в своем сегменте.

KPI

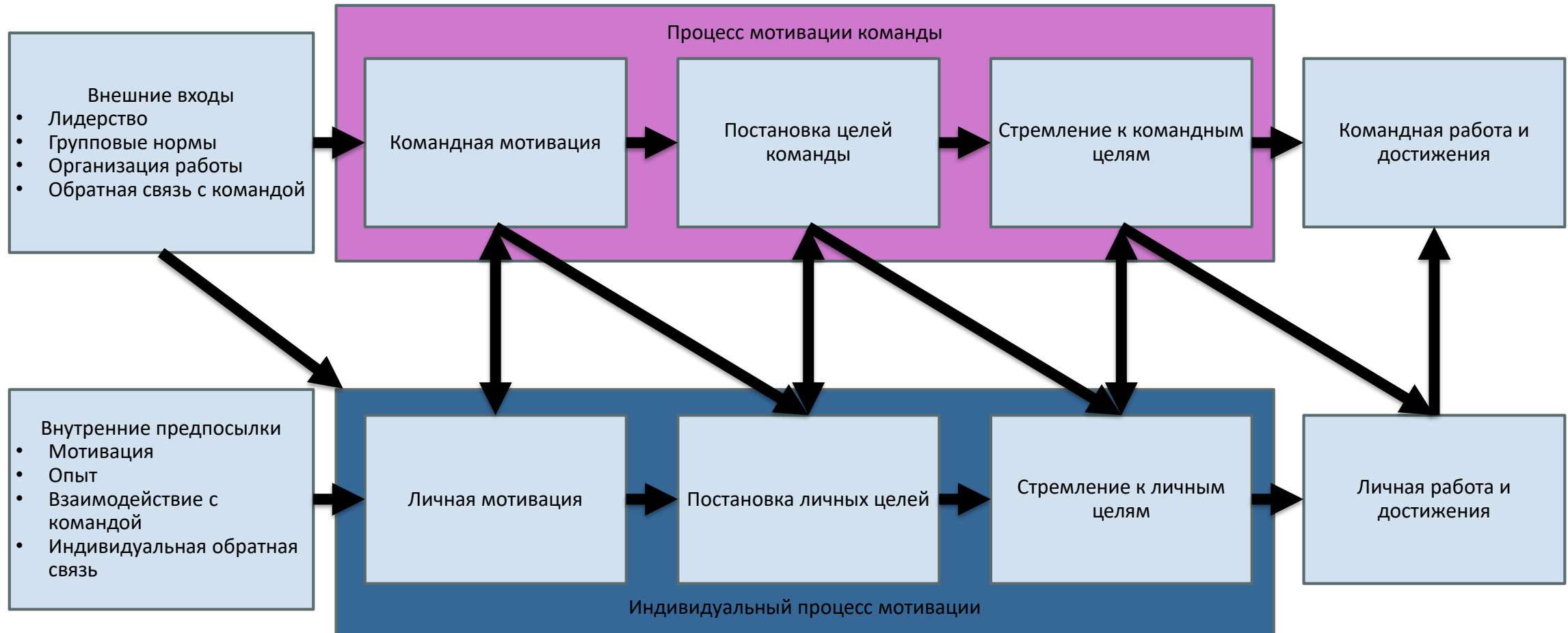
- Ключевые показатели эффективности, контролирующие ежедневную деятельность
- Явно достижимые показатели успешности деятельности, чаще всего понятной и налаженной.
- В них нет формулировки новой цели.
- Используется для наказания и поощрения сотрудников
- Примеры: EBITDA, Выручка, GMV YoY

MBO

- Цели на год
- Количество целей не ограничено
- Каскадирование сверху-вниз и снизу-вверх
- Формулировка по SMART
- Могут быть кросс-цели
- Пример: Привлечь 1,5 млн пользователей средний ценовой сегмент и обеспечить отношение расходов к доходам по сегменту не больше 24%.

Интегративная теоретическая модель командного и личного целеполагания

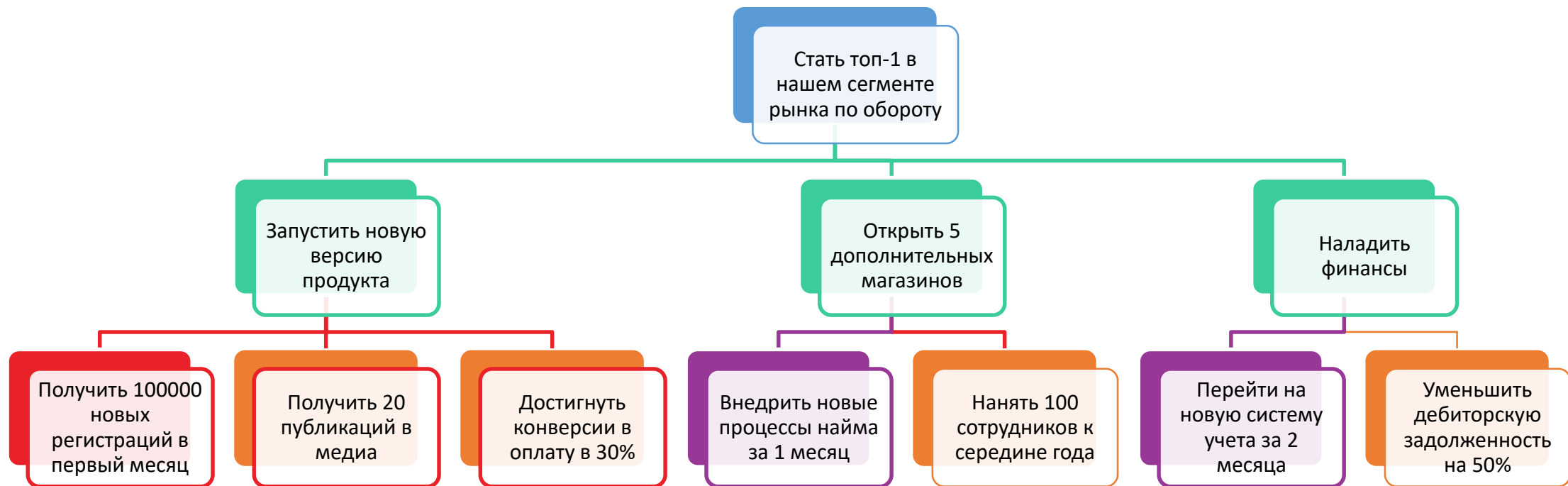
https://www.researchgate.net/publication/271995753_Toward_a_Systems_Theory_of_Motivated_Behavior_in_Work_Teams/download



Каким может быть процесс формирования OKR на практике?

- Стратегия Компании -> Годовые OKR -> Квартальные OKR
- 1-3 цели на команду, 2-4 KR на цель
- За каждой целью – дорожная карта (верхнеуровневое расписание работ)
- Формулирует продуктовая команда (с вовлечением в том числе команды разработки) – согласовывает руководитель продукта (процесс итеративный)

Цели и ключевые результаты образуют иерархию



Этапы планирования (пример).

Длительность: 30 дней

1. Команда формулирует черновик OKR и черновик дорожной карты (идеи, гипотезы для проверки, проекты)
2. Менеджер проекта или менеджер продукта согласовывает черновики с заинтересованными лицами
3. Подготовка оцифрованных ключевых результатов
4. Подготовка дорожной карты (расстановка приоритетов, скоринг и предварительная оценка проектов)
5. Согласование финальных OKR, включая дорожную карту

Типы ключевых результатов

Тип ключевого результата	Когда следует использовать	Примеры
Базовый уровень	При отсутствии метрики, отражающей важную цель	Получен базовый уровень онлайн активации купонов
Позитивная метрика	При постановке цели для метрики «Чем больше, тем лучше»	Выручка увеличена на 10% за счет email-рассылки
Негативная метрика	При постановке цели для метрики «Чем меньше, тем лучше»	Период обработки счетов сокращен с пяти до двух недель
Метрика пороговых значений	При постановке целевого диапазона значений для метрики	Коэффициент загрузки консультантов поддерживается между 70 и 80%
Веха	В случае невозможности выразить результат метрикой	Добавлена функция push-уведомлений

Трекинг успехов (пример)

- Еженедельный отчет в общее пространство (общий чат, доска, прогресс по KR, запуски, планы) – обеспечение прозрачности для всех заинтересованных лиц. (ответственный – менеджер продукта по умолчанию).
- Регулярная защита OKR со спонсором. Проверка-тестирование движения команды к цели (ответственный – менеджер продукта, могут участвовать и другие члены команды).
- Ежемесячный отчет для владельцев бизнеса и прочих заинтересованных лиц
- Приборная доска команды (если инструментов мониторинга нет – важно их сделать)

Критерий успеха

- По истечении заданного промежутка времени, например, квартала степень готовности по каждому из ключевых параметров оценивается по шкале от 0 до 1.
- Оценка производится самим сотрудником; цель считается достигнутой, если суммарно выполнено 70–75 % от запланированного.
- Если оказывается, что задача выполнена на 100 %, значит, выбранная цель была недостаточно амбициозной.
- Цели и ключевые результаты на год могут периодически пересматриваться, что позволяет компании оперативно реагировать на ситуацию на рынке.
- При этом цели на квартал менять не рекомендуется.

Рекомендуемая система оценки по шкале от 0 до 1

Баллов	Описание
1	Весьма амбициозный результат, кажется практически недостижимым.
0,7	Результат, которого мы надеемся достичь: сложный, но выполнимый.
0,3	Мы знаем, что этот результат достигим. Может потребоваться помощь других команд, либо мы справимся самостоятельно.
0	Прогресс отсутствует.

Еще раз про поощрения

- OKR не система мониторинга и контроля работы подчиненных.
- **Не используется для материальной мотивации — OKR управляет деньгами компании, а не сотрудников.**
- Не подходит для жестких, централизованных систем, подходит — для децентрализованных и запутанных.

5 вариантов внедрения OKR

Разработка OKR, уровень	Особенности
Только на уровне компании	• Четкая проработка целей (станут опорными точками для дальнейшего распространения OKR) • Демонстрация вовлеченности и поддержки со стороны топ-менеджмента
На уровне компании и бизнес-подразделения (или команды)	• Более амбициозный подход — Требуется единого понятийного аппарата и четких параметров внедрения
На всех уровнях компании	• Обеспечивает максимальную согласованность • Сложно начинать с этого варианта; много рисков
Пилотный запуск на уровне бизнес-подразделения или команды	• Подтверждает эффективность OKR • Демонстрирует быстрые победы и привлекает внимание других команд
На уровне проектов	• Оправданное решение для старта работы с OKR • Улучшает практику управления проектами

Что делать, что не делать



- Формировать OKR с привлечением всей команды
- Периодически смотреть шире своих OKR, смотреть на их согласованность со смежниками
- Создавать метрики
- Применять хитрые хаки и упрощения, чтобы запустить прототип быстрее
- Управлять накопившимися долгами по работе



- Неуправляемые KR
- O/KR не ведущие к целям уровнем выше
- Не обновлять прогресс по целям
- Не передоговариваться про цели, если усилия не приносят результат
- Ожидать, что OKR — это серебряная пуля
- «Ой, ну мы пока не можем это измерить»
- Мне такое сложно, скажите мне что делать и я сделаю. Хочу только проекты запускать и не думать про цели

Что еще не делать

- OKR используется для ведения обычных дел. Вместо OKR — план работ, список текущих задач или to-do лист. Цель OKR — масштабные изменения, а не текучка.
- Ошибки коммуникации. Для реализации амбициозных целей команды должны общаться между собой и учитывать OKR друг друга.
- Недооценка своих возможностей. Эту ошибку легко найти, если амбициозные цели выполняются на 100% — это значит, что роста нет.
- Неправильный выбор целей — их слишком много или мало, они недостаточно амбициозные или недостижимые.
- Смешивание методик — привязка к деньгам сотрудника или жестким KPI. Если вы не используете методики правильно, они не будут работать.
- Неготовность команд. OKR рассчитан на зрелых и ответственных исполнителей, методика раскрывает низкоэффективных сотрудников. Также сюда входят ошибки внедрения — не проведена подготовительная работа, не пройден этап проб и ошибок.

Потренируемся

- Что самое важное в этом объявлении для студентов, его читающих?
 - Елизавета Петрова, ректор Ухокрутского Государственного Университета, сегодня объявила, что в следующий четверг весь преподавательский состав едет в Заднепередонск на семинар по новым методам обучения. Среди спикеров будут антрополог Иван Сидоров, руководитель местного отделения Минпросвещения Степан Разин и мэр города Ирина Кравченко.



**Национальный исследовательский университет
Высшая школа экономики
Высшая школа управления проектами**

Бонус

Для больших команд и организаций (справочно)

<http://www.agilescaling.org/ask-matrix.html>

Aspects / Criteria	Scrum-of-Scrums (SoS) PO meta-scrum	Large Scale Scrum (LeSS) Larman/Vodde	Scaled Agile Framework (SAFe) Leffingwell	Disciplined Agile Delivery (DAD) + Agility at Scale Ambler/Lines	Spotify "model" (Tribes, Squads, Chapters & Guilds) Kniberg	DSDM Drive Strategy Deliver More	Recipes for Agile Governance (RAGE)	Nexus / Scaled Professional Scrum Scrum.org	Scrum at Scale Sutherland/Brown
Description	An important mechanism that may be enough for smaller organizations but is not a full scaling approach	Larman / Vodde model as documented in "Scaling Lean & Agile Development"	The method documented by Dean Leffingwell and Scaled Agile, Inc.	Scott Ambler model documented in the book "Disciplined Agile Delivery"	The method used at Spotify, featuring Squads, Tribes, Chapters & Guilds.	DSDM is a robust Agile project management and delivery framework that delivers the right solution at the right time	Traditional-agile hybrid of portfolio-project planning	Scrum-based scaling with an "exoskeleton" called Nexus plus over 40 practices	Scrum Inc.'s modular framework for scaling Scrum
Web Link	guide.agilealliance.org/guide/scrumofscrums.html	less.works/	scaledagileframework.com/	disciplinedagiledelivery.com/	scaling-agile-spotify-11.pdf	www.dsdm.org/	www.cprime.com/tag/agile-governance	www.scrum.org/Resources/The-Nexus-Guide	www.scruminc.com/
What Team level frameworks are supported? (Scrum, Kanban, XP, etc.)	Scrum	Scrum	Scrum / Kanban / specific XP practices "mandated"	Scrum /Lean	Own method though partly Scrum-like	Own Hybrid Agile Scrum method	Scrum /Lean	Scrum	Scrum
Software centric - how often used outside of SW or IT?	Could use anywhere you use Scrum	Focused on Software or SW/HW	Focused on Software or SW/HW	Has been used outside of IT	Spotify only	Has been used outside of IT	Focused on Software	Could use anywhere you use Scrum	Could use anywhere you use Scrum
Big Positives / Key Differentiators	simple, standard Scrum focus on dependencies & resolutions	Good PO scaling; strong principle alignment, Non-prescriptive - gives "suggestions"	The "big picture" and completeness; getting Agile "in the door" at large corporations; actively evolving.	Lots of content; strong in areas such as architecture, design and dev ops; incorporates many good models.	very agile, entrepreneurial, distributed teams, low overhead	Very established following in the UK	Fluid and adaptive	Authored by Ken Schwaber	Lightweight authored by Jeff Sutherland
Key Risks / Concerns	limited scaling, limited documentation, not clearly defined Not likely "sufficient" for large scale; some differences in implementation	A more "radically agile" approach that may be a hard sell in larger traditional orgs with many layers and specializations.	Little info on "how", most need certified SPCs to implement properly; Seen as prescriptive; not "agile enough" in its structures; "quick start and leave" issues some places	vague in some areas about the "how"; can come across as a bit disjointed. Not prescriptive in lifecycle.	very limited detail about the "how", Not really a framework; may only fit certain cultures	Heavy process overhead.	New approach that is growing and adapting.	New approach that is growing and adapting. Some of the parts are "secret" unless you go to the class.	New approach that is growing and adapting.
Training / Resource availability	None known; roll your own	Training and coaching network available	Yes, multi-level training & Certifications	Yes, multi-level Certifications	None known; roll your own	Yes, multi-level Certifications	None known read and implement yourself for your need.	Scaled Professional Scrum training & certification is available	Courses are offered

Спасибо за внимание!

