

Процессы разработки от Waterfall до гибких подходов и как выбрать тот, что требуется твоей команде

Евгений
Россинский

Learn.
Create.
Collaborate.

1956

Водопад (Waterfall)



Этапы:

- определение системных требований
- определение программных требований
- анализ
- проектирование
- разработка
- тестирование
- эксплуатация

Развивается в итеративную модель разработки

1956

Итеративный водопад

- Почему-то все считают, что итеративный подход это заслуга именно гибких методологий
- На самом деле это всего лишь оптическая иллюзия при изменении масштаба
- Ну и грамотный PR



Спросим человечество: “Как ты жило до
появления водопада в 1956?”



Человечество: “Да точно так же как и после 1956”

Человечество: “Просто в 1956 мы осознали, что процессы стали настолько сложными, что из этого пора сделать науку”

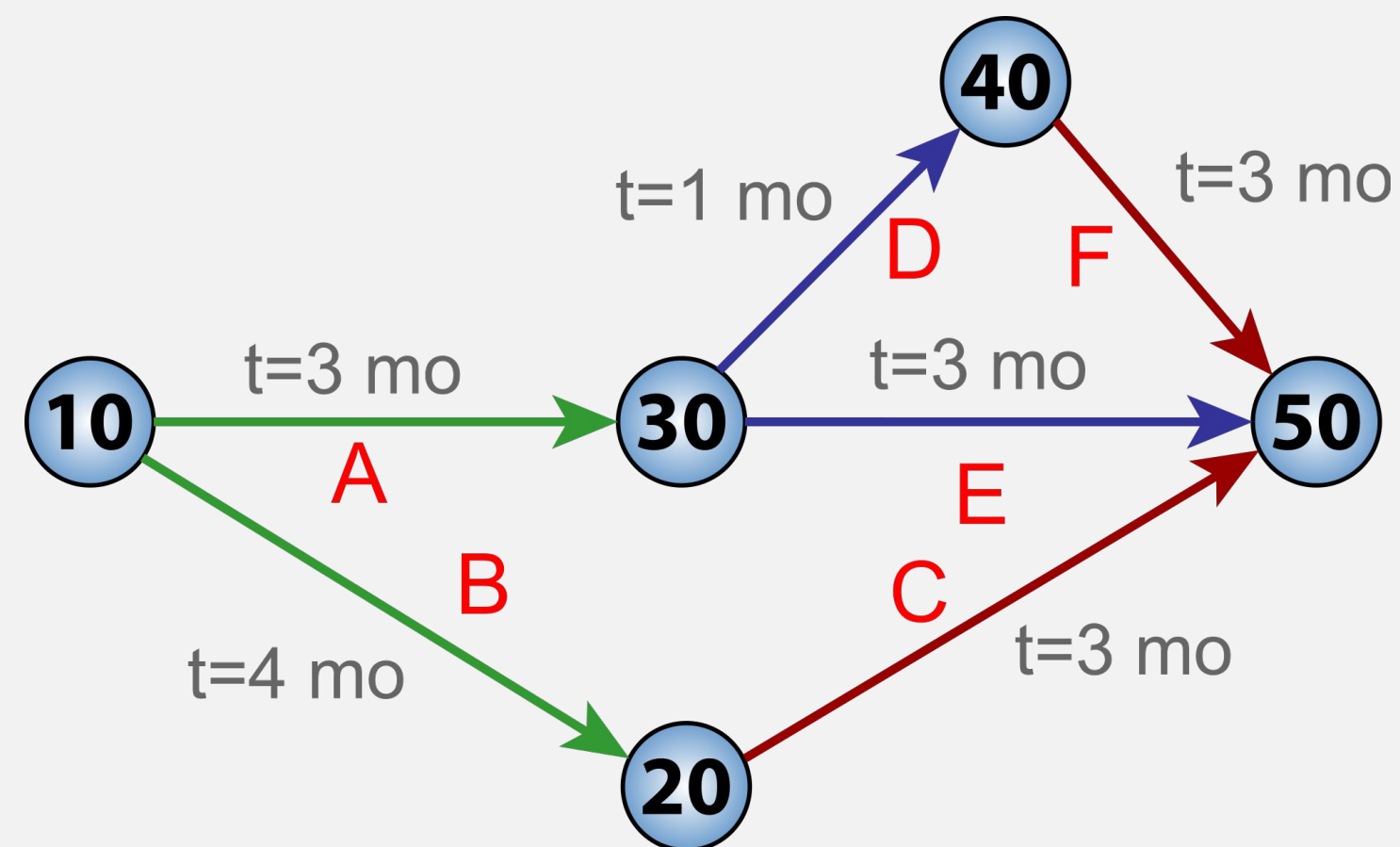
Человечество: “И плевать нам, как вы это назвали, главное, что оно работает”



1958

PERT

**PROGRAM (PROJECT) EVALUATION AND
REVIEW TECHNIQUE**



- Метод критического пути
- Самая известная часть PERT — это диаграммы взаимосвязей работ и событий
- Вершины – события
- Дуги – работы
- Подготовка к зимним олимпийским играм 1968 года
- Грубо говоря, не совсем методология разработки, скорее инструмент оценки рисков и планирования

1959

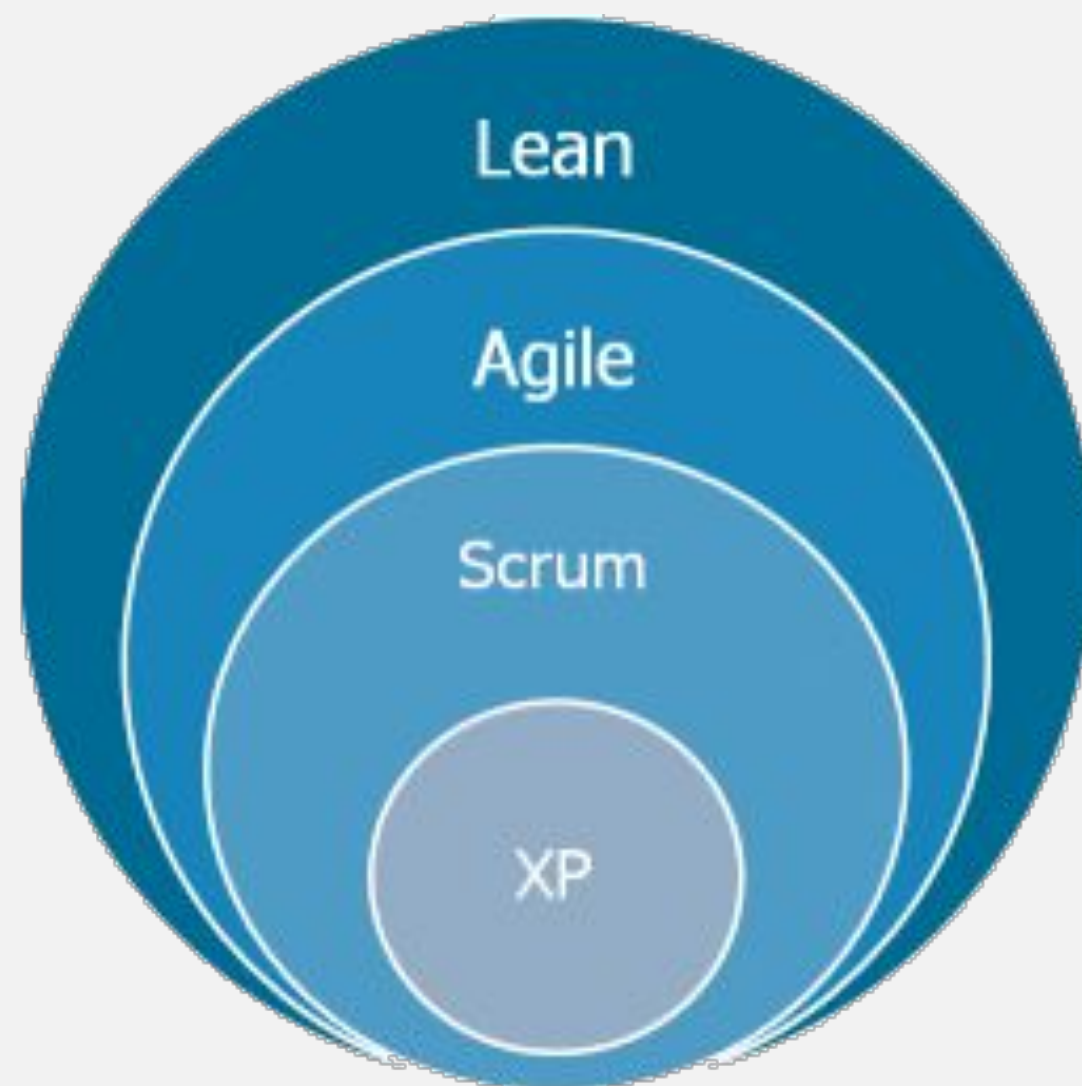
Kanban



195x

Lean

Toyota Production System

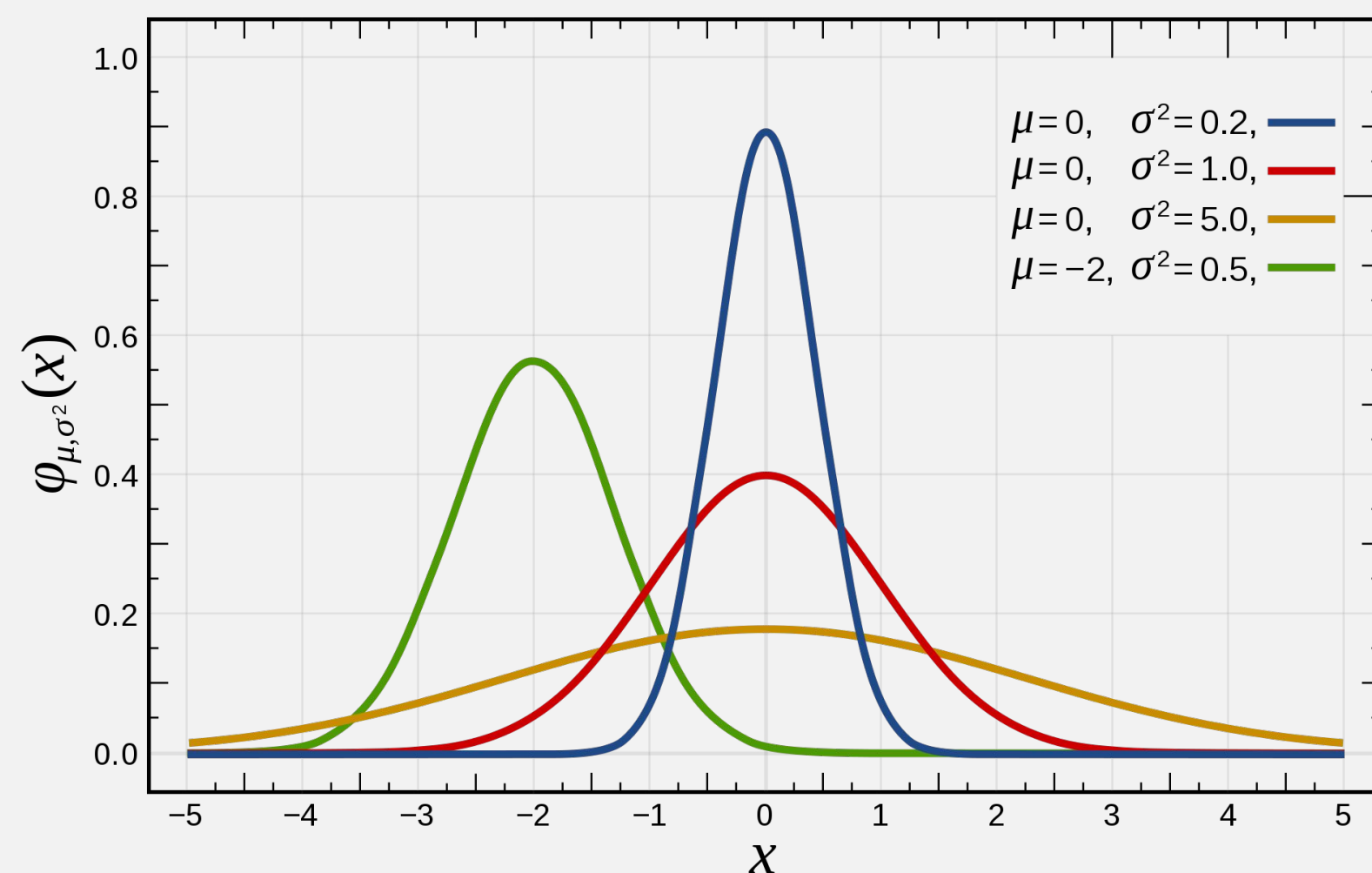


- Прекрасен для систем массового обслуживания
- Основное понятие в Lean — это ценность продукта (value)
- Классификация потерь
- Инструмент для поиска проблем в процессах
- Философия постоянного совершенствования всех процессов производства продукта

1986

Six Sigma

Motorola



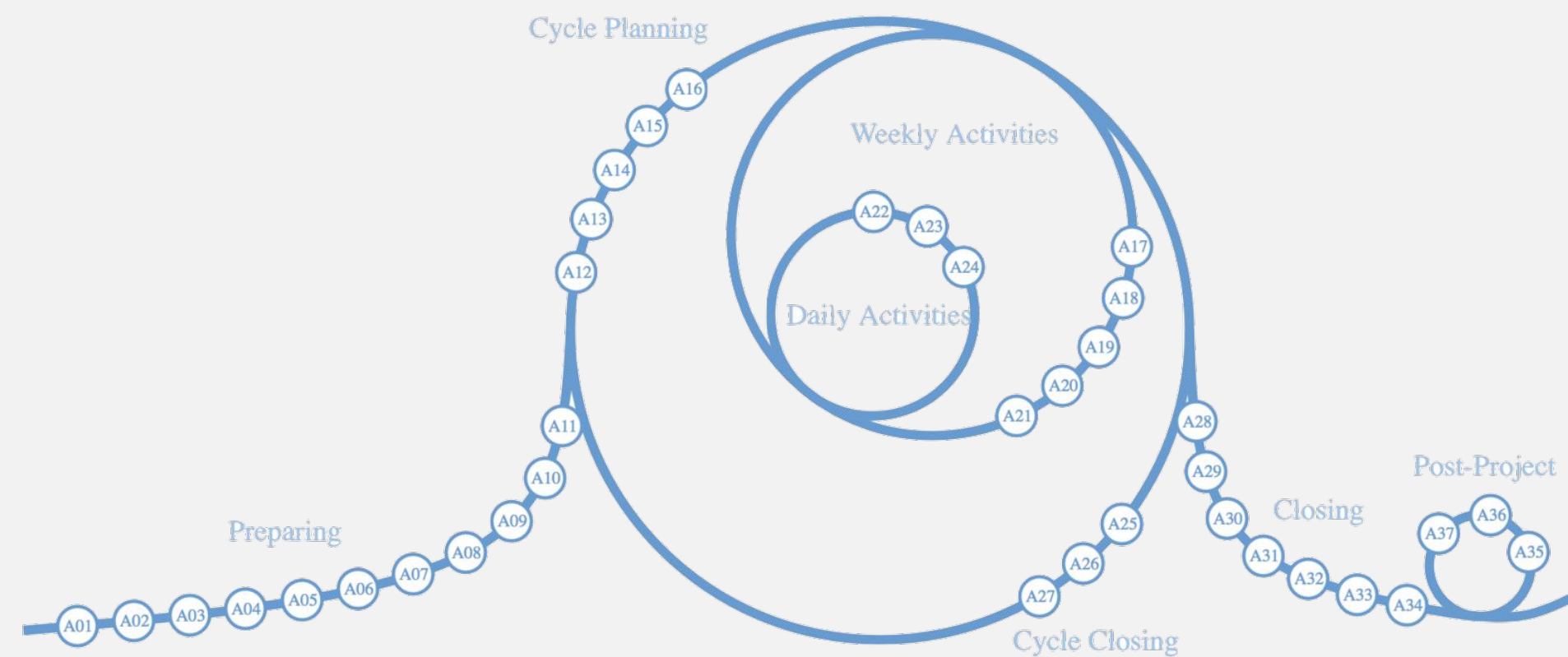
- Искренний интерес к клиенту
- Управление на основе данных и фактов
- Ориентированность на процесс, управление процессом и совершенствование процесса
- Стремление к совершенству плюс снисходительность к неудачам
- Интересные роли
 - Чемпион
 - Зеленый пояс
 - Черный пояс
 - Сенсей (мастер черных поясов)
- изделие, отвечающее требованиям, занимает на оси абсцисс положение в нуле, если оно лучше, чем требуется — правее нуля, хуже — левее.

1986

PMBoK

Десять областей знаний, которыми должен обладать руководитель

- Управление интеграцией проекта
- Управление содержанием проекта
- Управление сроками проекта
- Управление стоимостью проекта
- Управление качеством проекта
- Управление человеческими ресурсами проекта
- Управление коммуникациями проекта
- Управление рисками проекта
- Управление поставками проекта
- Управление заинтересованными сторонами проекта

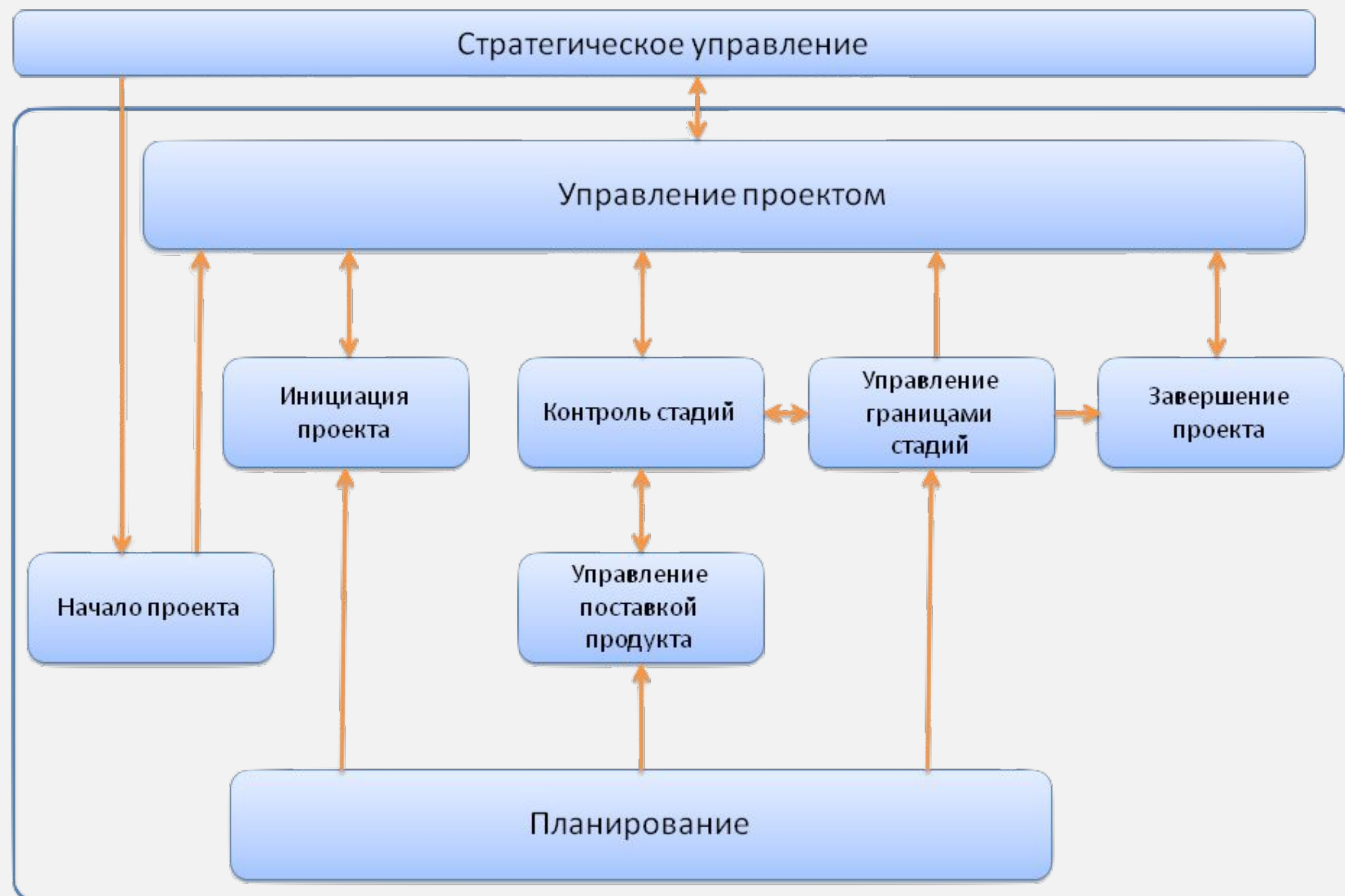


1989

PRINCE2

«P**R**ojects IN Controlled Environments version 2»

Британские ученые :)



- Интересные роли
 - Менеджер проекта
 - Комитет проекта (project board),
 - Заказчик
 - Главный пользователь
 - Главный специалист
 - служба project assurance (аналог проектного офиса) - независимое мнение о проекте
 - business report
 - user report
 - technical report
- Прекрасно интегрируется с Agile

1991

RAD

RAPID APPLICATION DEVELOPMENT

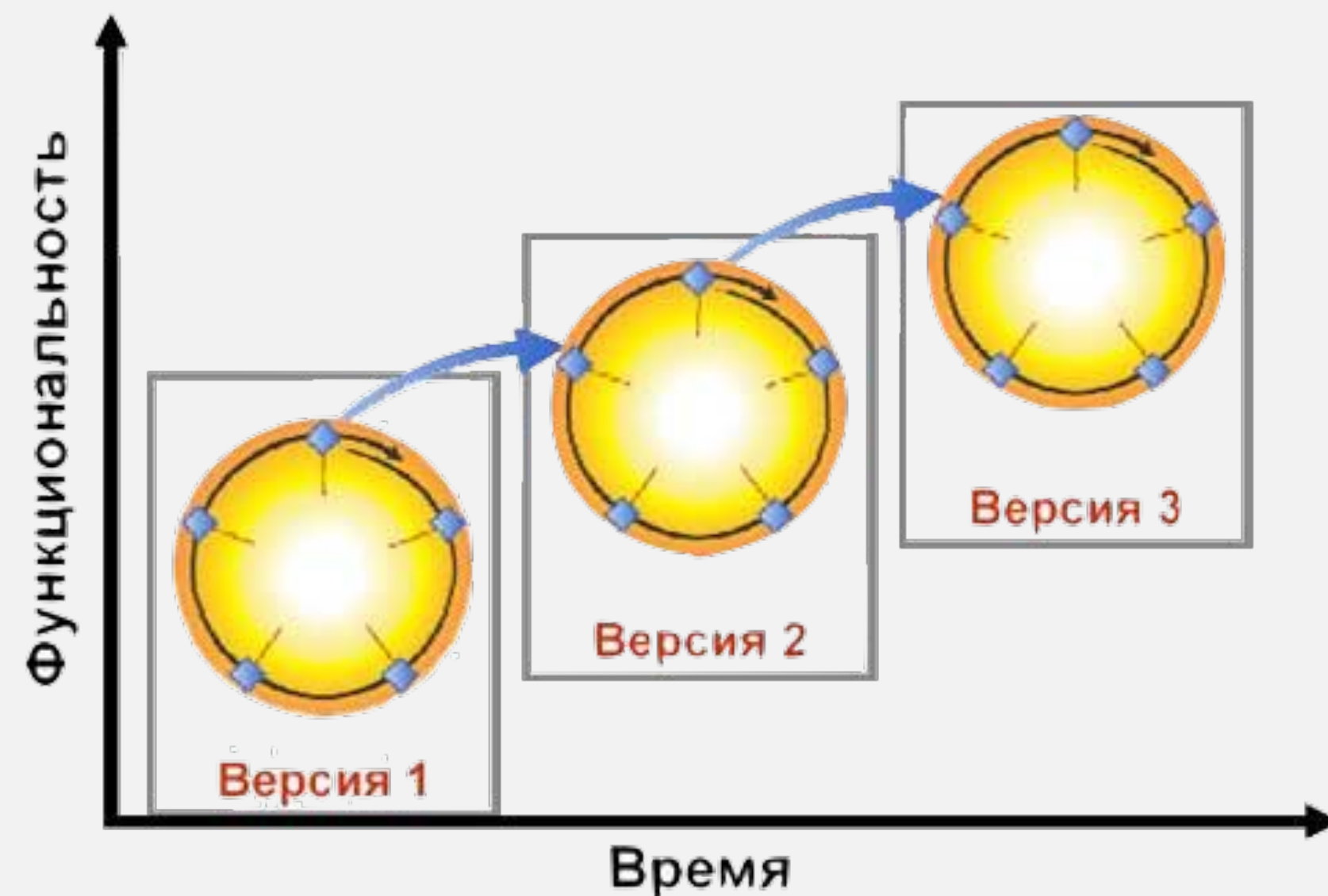


- Концепция быстрой разработки программного обеспечения
- Построение прототипов при минимальном планировании
- Расстановка приоритетов, упор делается на время и затрачиваемые ресурсы
- Если разработка не критичной части функционала не будет укладываться в сроки, RAD подразумевает отказ от данного функционала.

1993

MSF

MICROSOFT SOLUTIONS FRAMEWORK

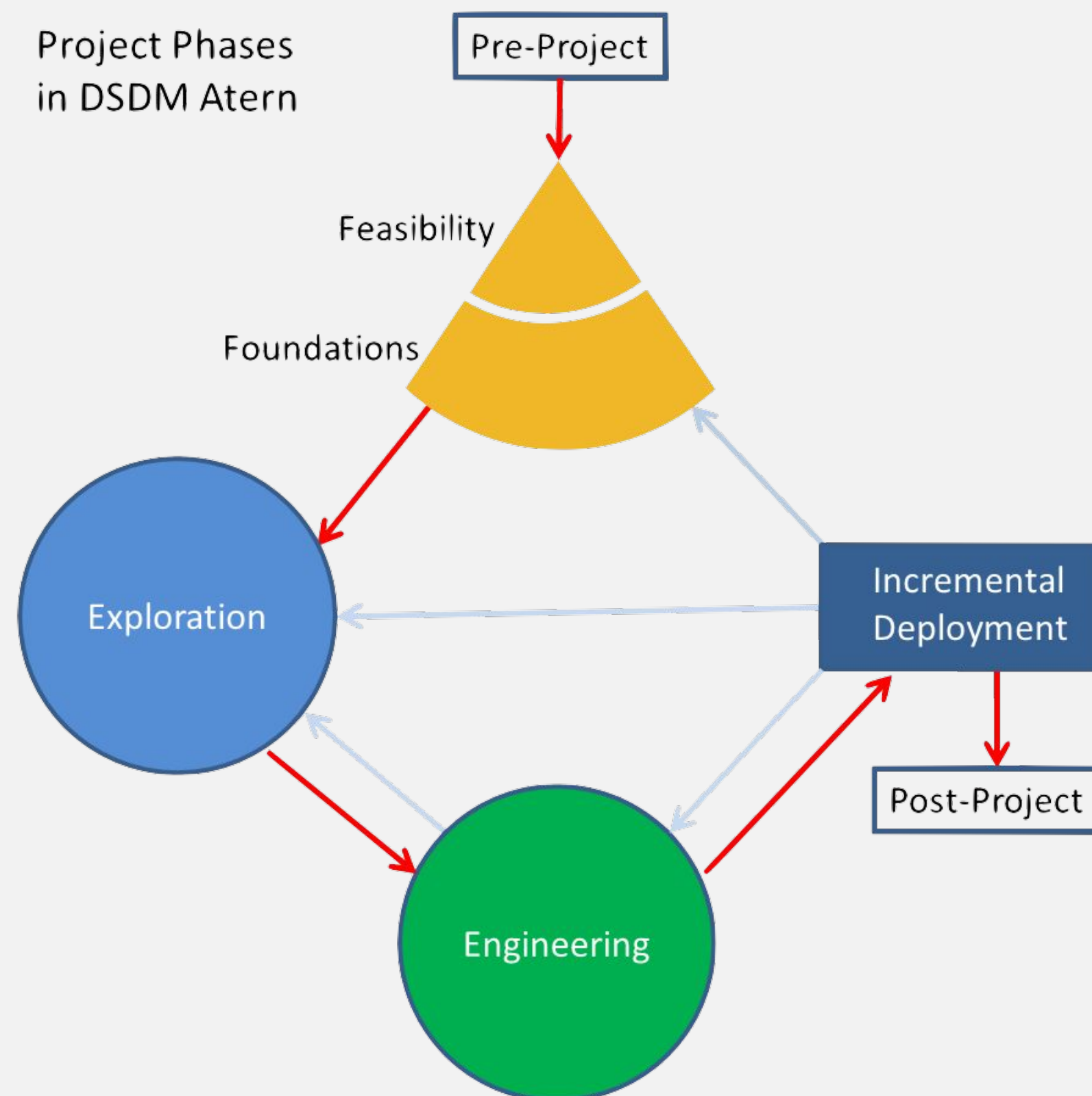


- Этапы похожи на Waterfall, с упором на эффективное распределение ролей в команде
- Поощрение коммуникаций
- Максимально открытая информация о проекте внутри команды
- Наделение членов команды полномочиями, ответственностью и отчетностью
- Проявляйте заботу о команде соратников
- Непрерывное обучение
- Контрольные точки, как события в PERT

1994

DSDM

Разработан на основе RAD 17ю британскими компаниями



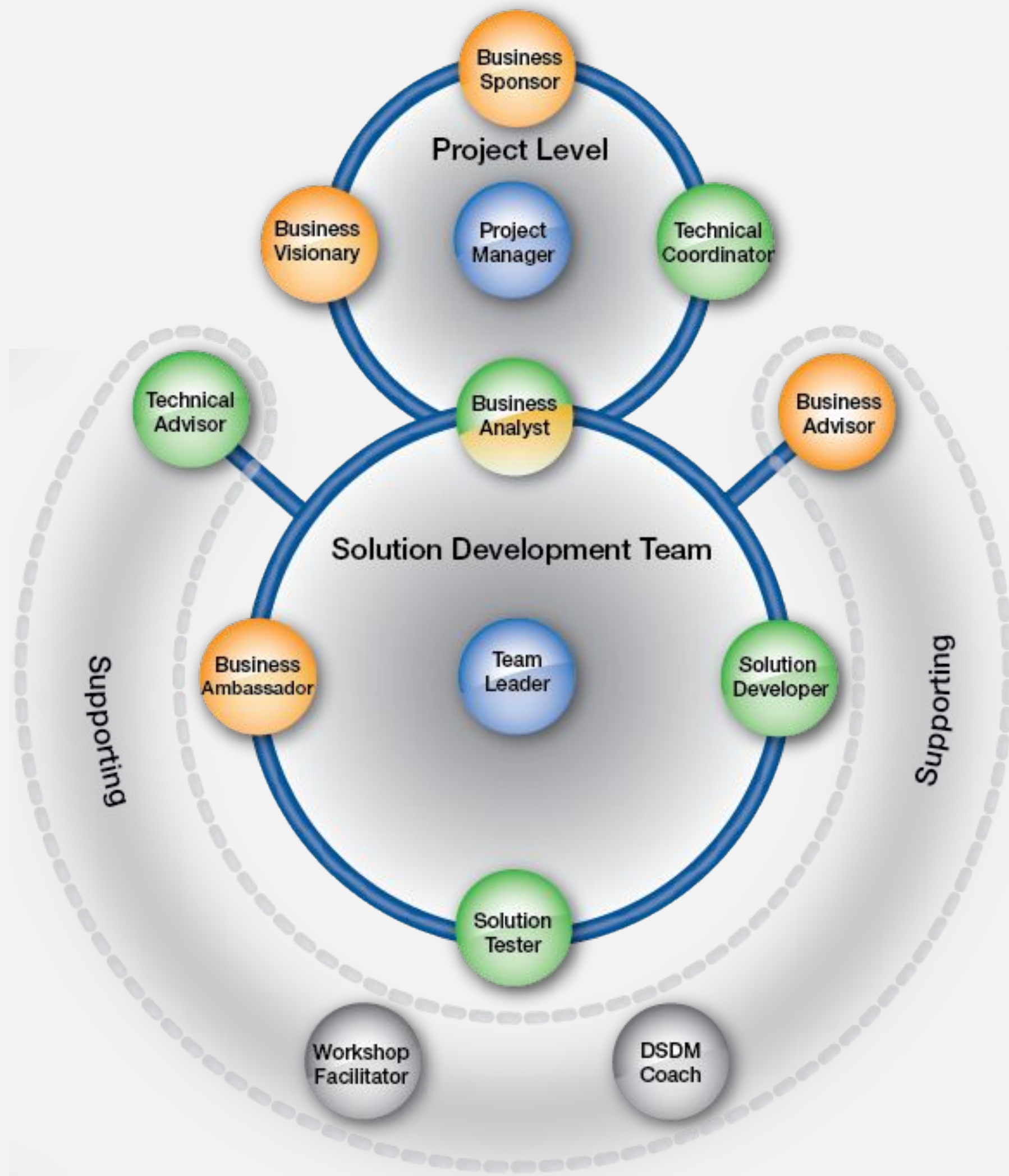
- Таймбокс – делим весь проект на части, каждая со своим бюджетом и сроками выполнения.
- Принцип назначения приоритетов «MoSCoW»
 - должно быть (must have)
 - следует, чтобы было (should have)
 - нужно (could have)
 - можно (want have)
- семинары
- построение прототипов
- **обязательный анализ экономической целесообразности**

Роли DSDM

Разработан на основе RAD 17ю британскими компаниями

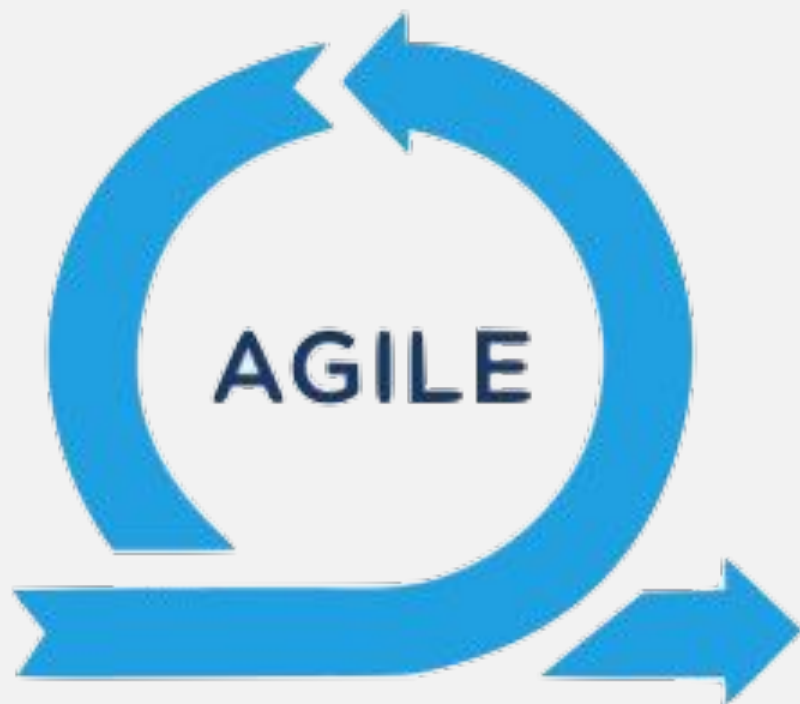
1994

- Менеджер проекта
- Провидец (коммерческие цели)
- Чемпион проекта (распоряжение ресурсами)
- Лидер команды (руководит разработчиками)
- Технический координатор (архитектура)
- Разработчик
- Тестировщик
- Представительный пользователь
- Пользователь консультант
- Секретарь (протоколирование)
- Фасилитатор (взаимодействие между членами команды)



2001

Agile Manifesto



Люди и общение

а не

Процессы и инструменты

Работающее приложение

а не

Сложная документация

Сотрудничество с клиентом

а не

Составление контрактов

Реакция на изменения

а не

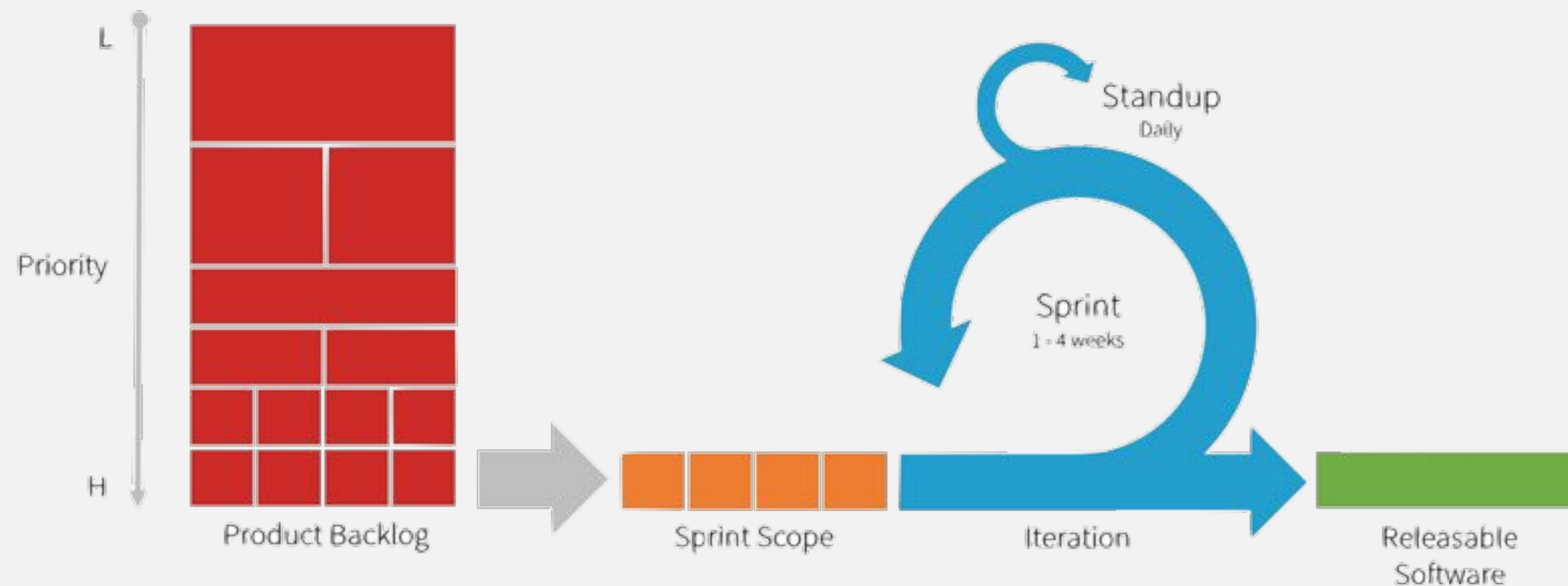
Следование плану

2002

Scrum Alliance

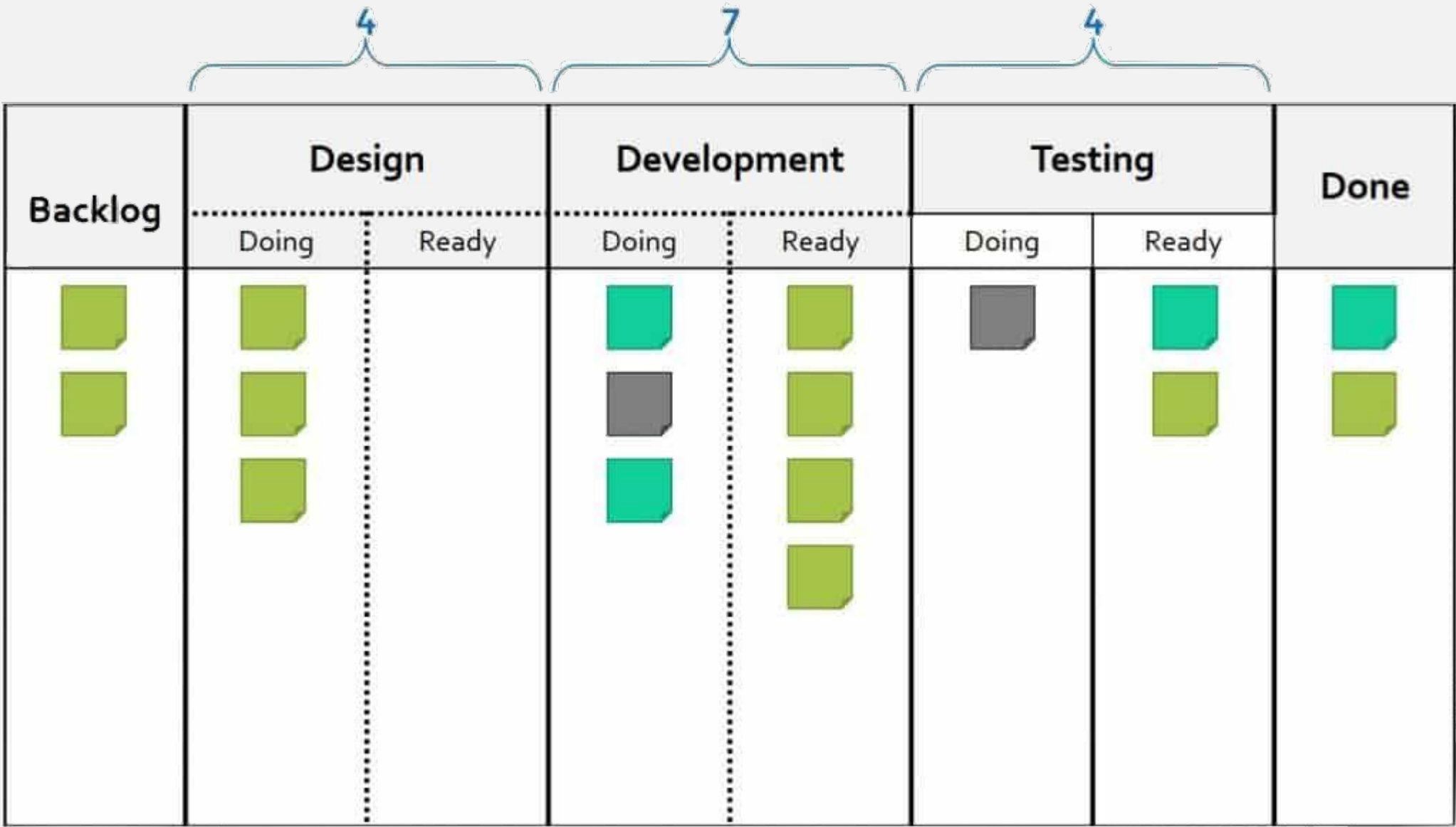
Первые упоминания Scrum в 1986, но популярность пришла после Agile Manifesto

- Есть ли в зале люди, которые не знают, что такое Scrum?
- Основная идея - наладить коммуникацию внутри команды с помощью простых ритуалов
- Правила есть, они не очень жесткие



Kanban

имеется ввиду применение для разработки ПО



- Есть ли в зале люди, которые не знают, что такое Kanban?
- Правил меньше, чем в Scrum'e
- Подходит для более зрелых команд
- Может работать сверху над любым существующим процессом
- WIP лимиты

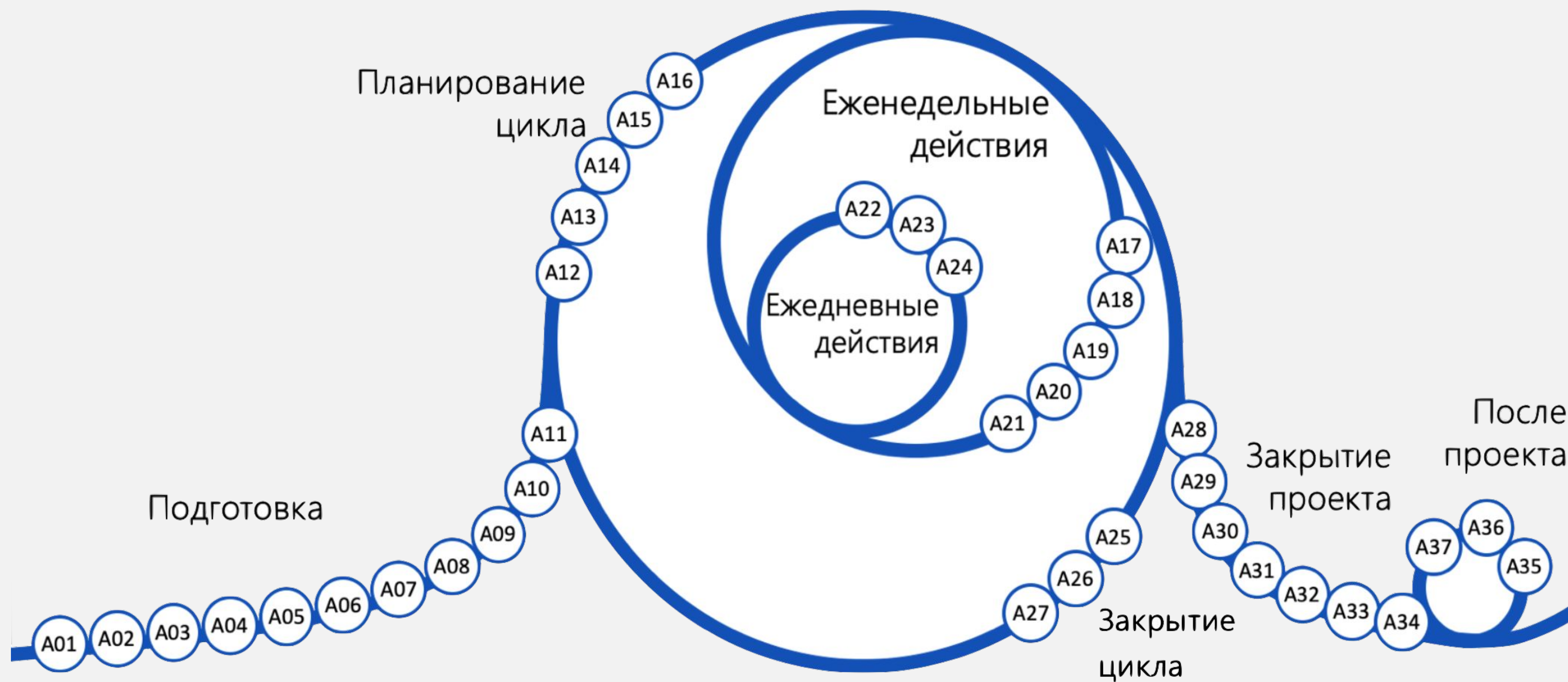
2018

p3express

- Проще чем PMBOK
- people
- projects
- products

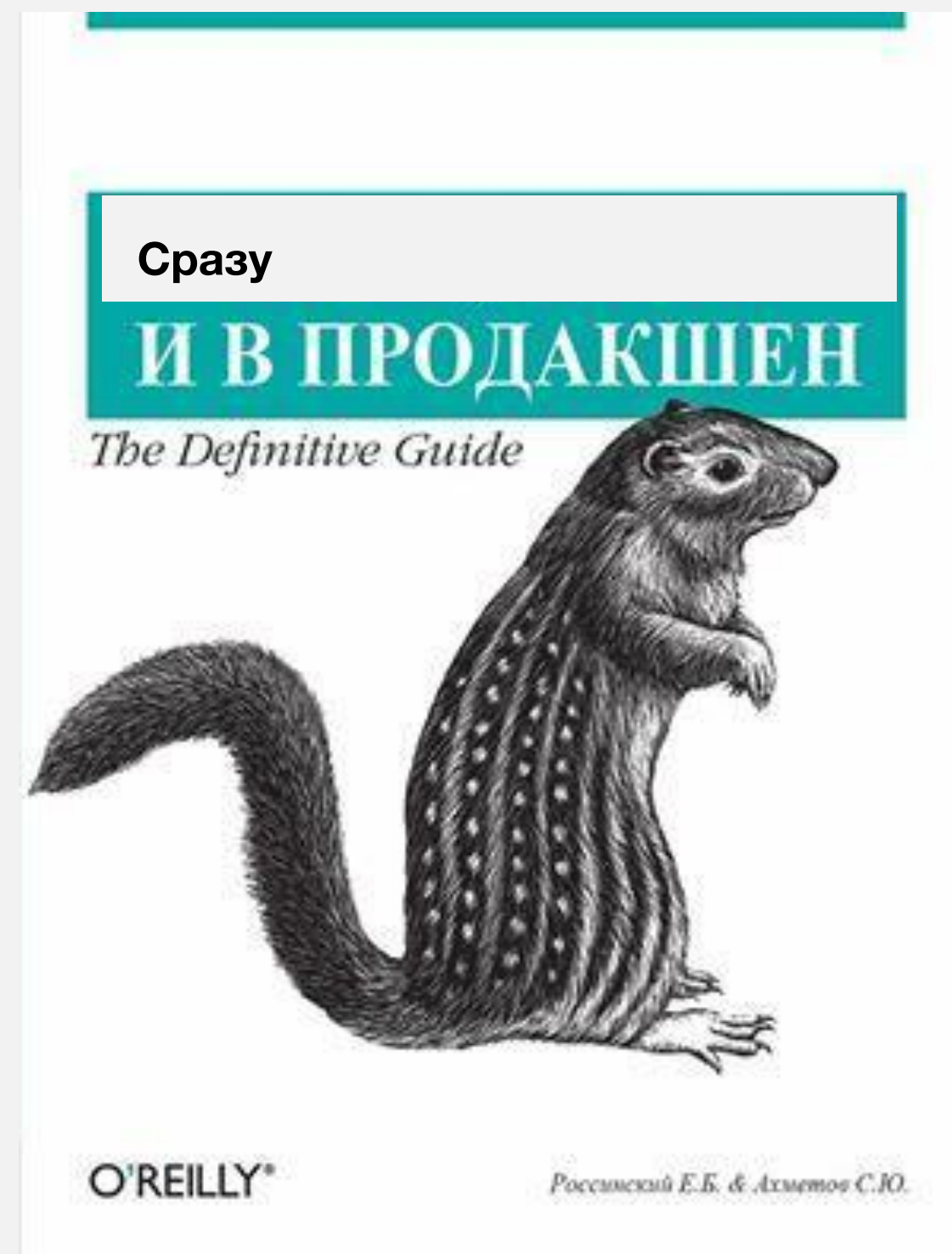
■ отправить материалы:

- pdf
- Trello шаблон доски



МЛН. лет. до н.э.

Вечная методология



- Неважно, какие у тебя есть ресурсы
- Неважно, какая у тебя команда
- Неважно, какие у тебя процессы
- Важно, что функционал нужен вчера

Как **выб**рать то, что
подойдет **твоей**
команде?



Предположим, вы
работаете в команде с
какими-то **процессами**



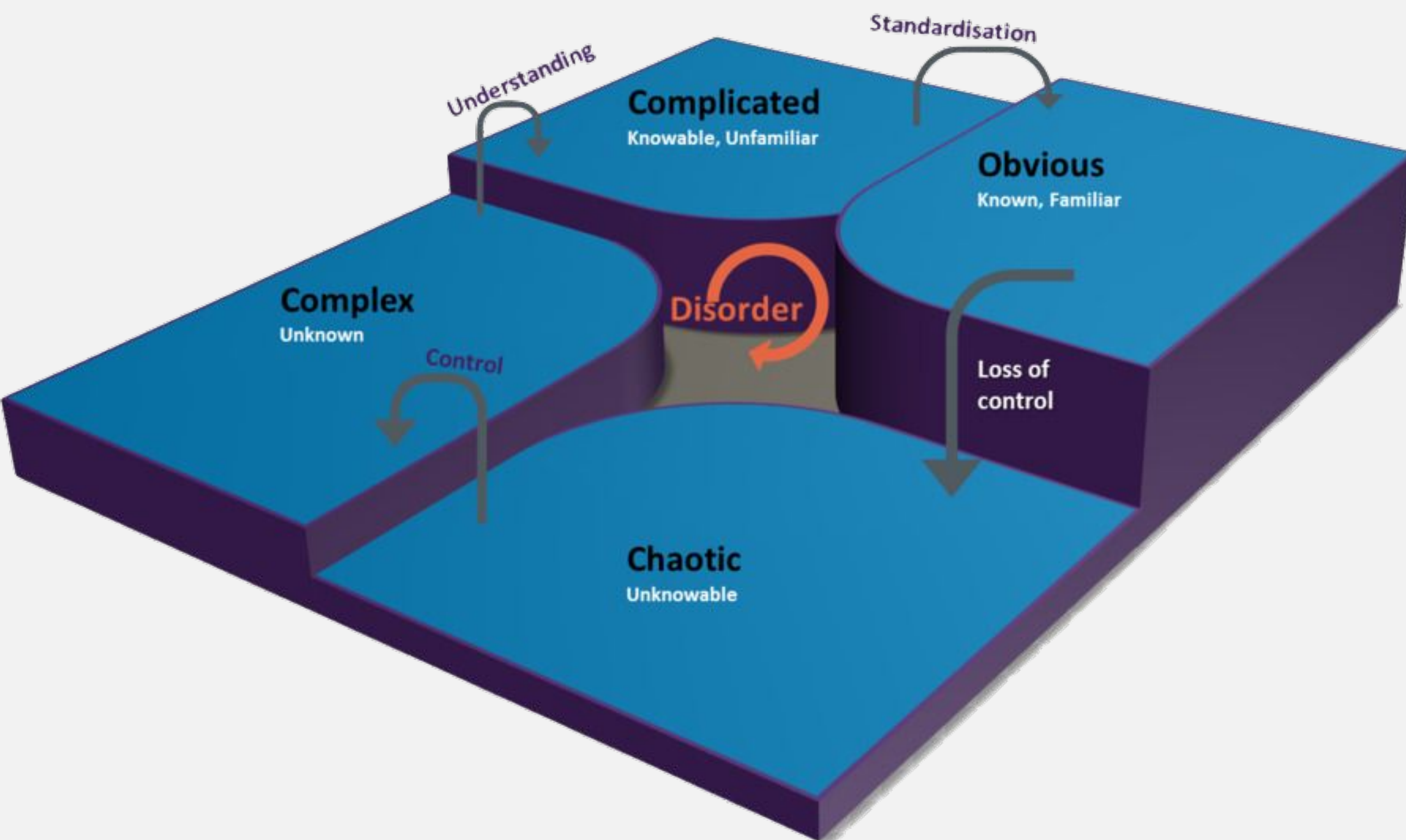
Попробуем заняться самолечением.

Проведем **классификацию** своей
компании с помощью **Cynefin Framework**

Learn.
Create.
Collaborate.

Cynefin Framework

Психотерапевт методологий, теория запутанности



Домены:

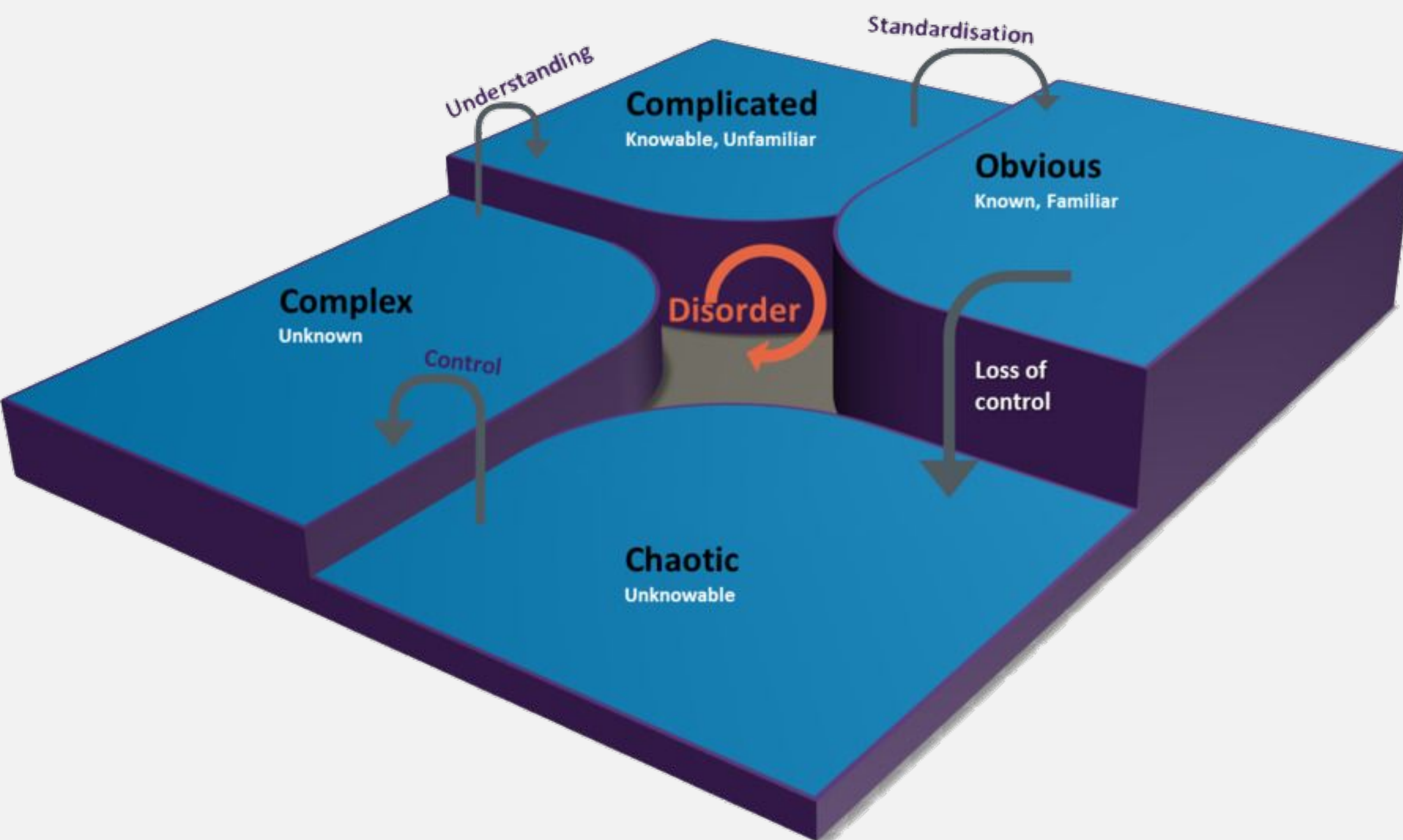
- Очевидный (Obvious)
 - Ощутить – Откатегоризировать – Отреагировать
 - Best practices
 - Управляется «сверху-вниз»
- Сложный (Complicated)
 - Ощутить – Проанализировать – Отреагировать
 - Лишь хорошие практики
 - Управляется «сверху-вниз»

1999

Домены:

Cynefin Framework

Психотерапевт методологий, теория запутанности



- Запутанный (Complex)
 - Попробуй – Ощути – Отреагируй
 - Ни хороших, ни тем более лучших практик в этом домене быть не может
 - Эффективный управленец – лидер слуга
- Хаотический (Chaos)
 - Действуй – Ощути – Реагируй
 - инновационные практики, появившиеся случайно
 - авторитаризм, лидерство
- Неопределенный (Disorder)
 - Все что не относится к 4м доменам

Теперь понятно, почему гибкие
методологии стали популярны?

Learn.
Create.
Collaborate.

Определили, в каком домене вы
находитесь, и в какой **стремитесь?**

Самое время сделать внешний процессов аудит

Learn.
Create.
Collaborate.

Не все понимают цели

- Синхронизация бизнеса и разработки
- Определение цели и миссии
- Нужно ли что-то менять?

План

- Тактический план трансформации
- Стратегический план трансформации
- Экономический план трансформации
- Анализ рисков

Наставники

- В ближайшие полгода идеологи преобразований должны быть в команде

Метрики

- ТТМ
- Эффективность релизов
- Критерии успеха
- Критерии провала

Какого размера ваша команда?

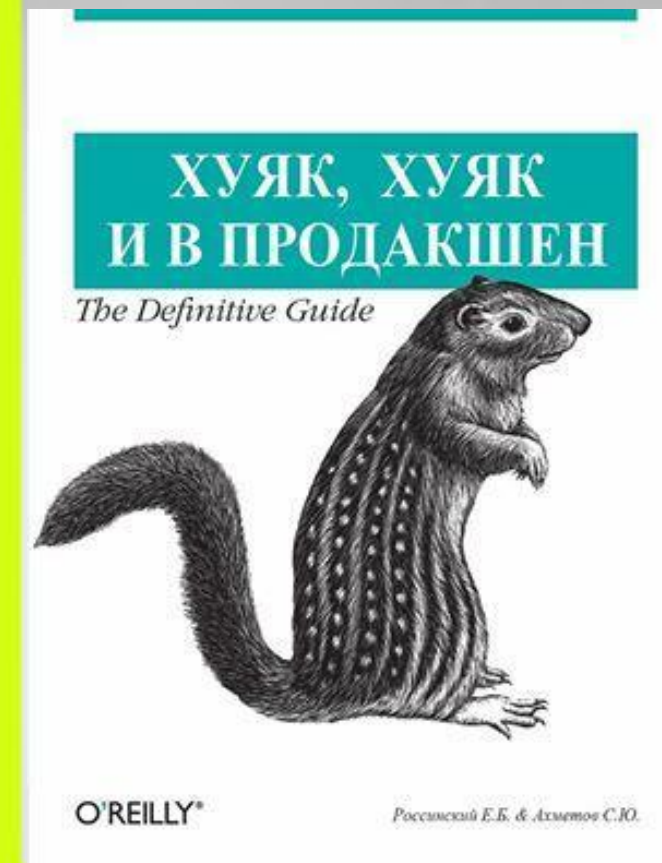
Есть ли возможность потренироваться
с новыми правилами на небольшой
группе добровольцев?

Learn.
Create.
Collaborate.

Решение принято.

Сжали булки и бежим, пока не
достигнете критерия **успеха** или
критерия **провала**

Learn.
Create.
Collaborate.



Время для маленького практического занятия.

Возьмите трубочку и два стаканчика...

Learn.
Create.
Collaborate.



Не важно, как называется твоя
методология,

главное, чтобы **команда** была
счастлива и **показывала**, нужный
бизнесу **результат**



evgeny@rossinskiy.ru