

Проектное управление при разработке вычислительной техники

к.т.н. Никаноров А.В.

Проект

Временное предприятие, направленное на создание уникального продукта, услуги или результата (см. PMBOK)

Предприятие с определёнными датами начала и завершения, предпринятое для создания продукта или услуги (сервиса) в соответствии с заданными ресурсами и требованиями (ГОСТ Р 57193-2016 Системная программная инженерия)

Проект = Риски & Ограничения

Что значит управлять проектом?

Обеспечивать выполнение работы в рамках согласованных сроков, стоимости и содержания работ, обеспечивая удовлетворенность Потребителя.

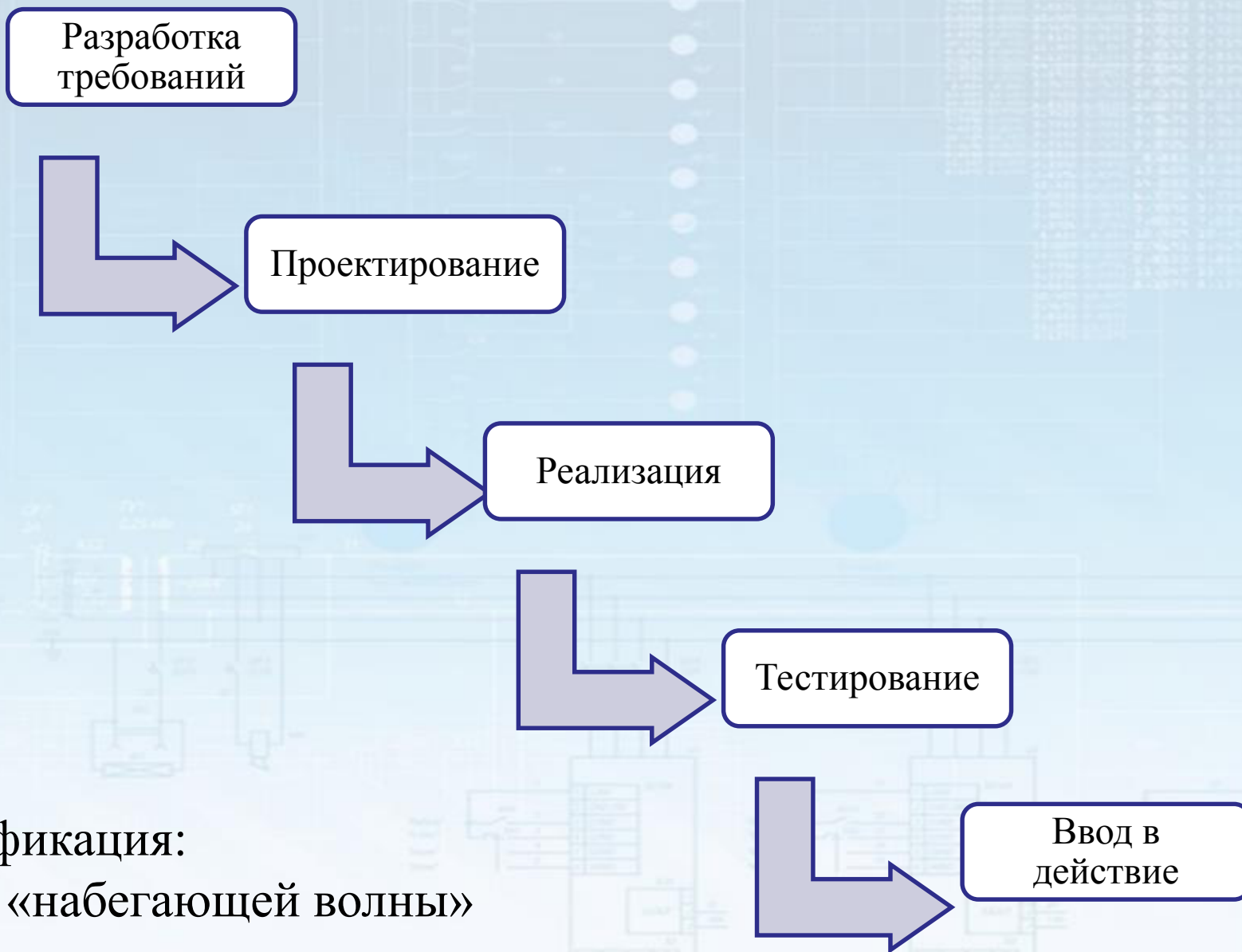


Общие модели жизненного цикла проекта

- Водопадная, каскадная
- V-Model
- Спиральная
- Итеративная
- Инкрементная («мультиводопад»)
- ...



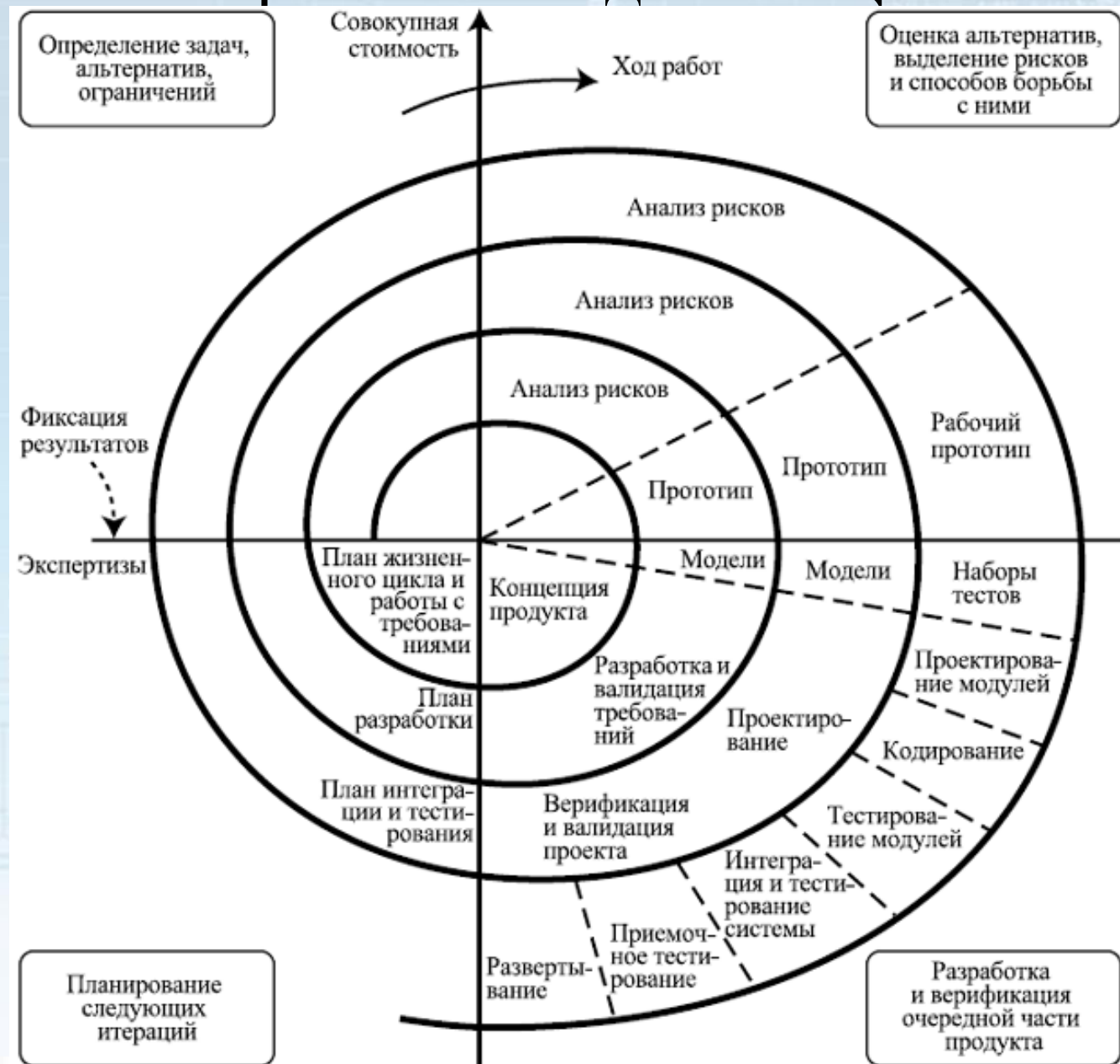
Каскадная (водопадная, waterfall) модель ЖЦ



Рекомендации для построения ИСР (иерархической структуры работ)

- Ориентированность на поставки
- Первый уровень – главная поставка (продукт)
- Второй уровень должен давать стороннему наблюдателю общее понимание, «что будет сделано на проекте»
- Все узлы (по крайней мере, одного уровня) поименованы в одном стиле (лучше - существительное).
- Можно построить ИСР исходя из жизненного цикла проекта (проектирование, тестирование).
- Узлы не перекрываются по содержанию.
- 100% - входят все поставки, и внешние, и внутренние
- Не нужно строить ИСР исходя из оргструктуры.

Спиральная модель ЖЦ



V - модель

Требования
заказчика

Проверка общих требований

Приемочные
испытания

Системные
требования

Проверка функционала

Системные
тесты

Требования к
- Электронике
- Механике
- ПО

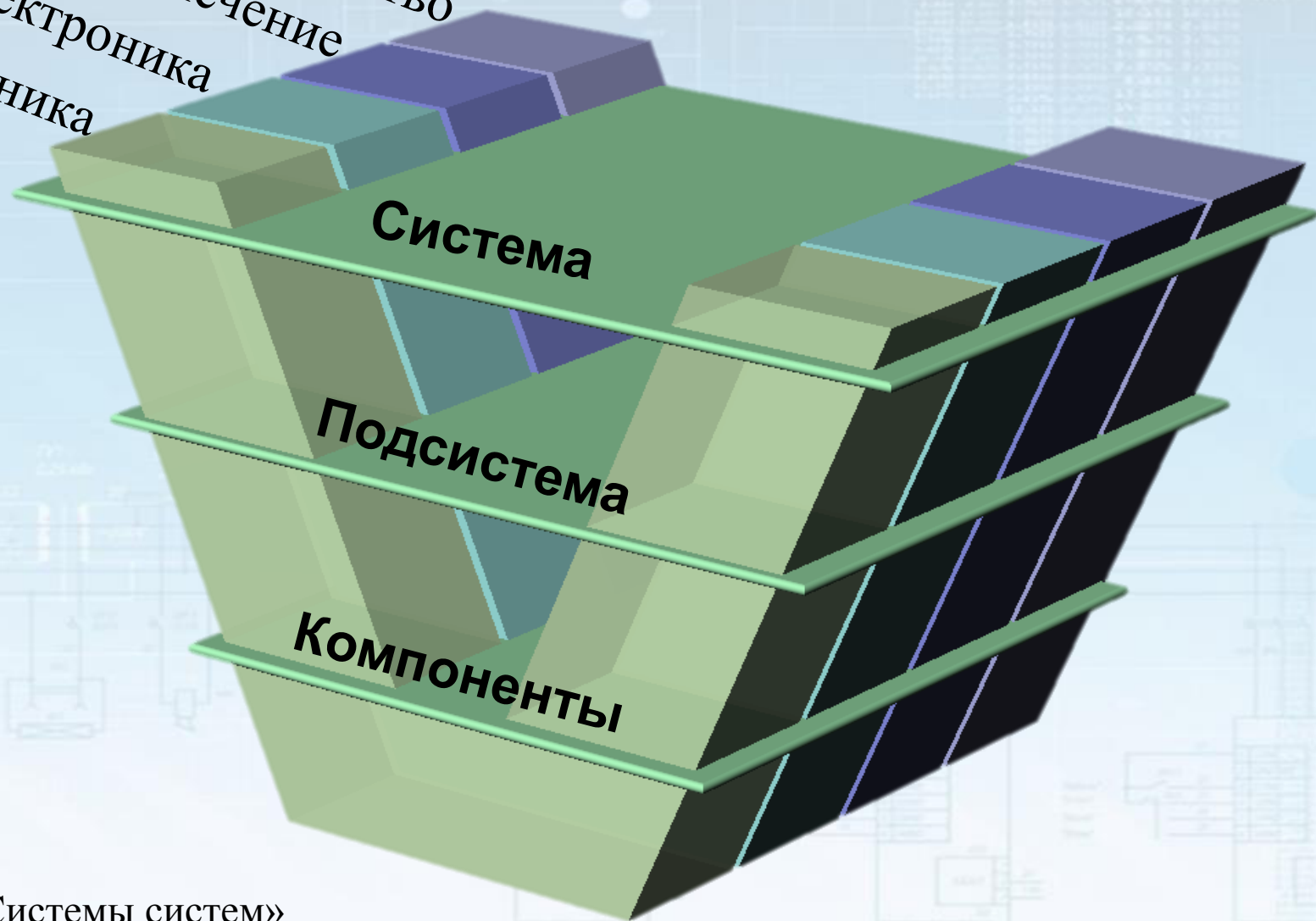
Проверка реализации

Тестирование
- Электроники
- Механики
- ПО

Создание отдельных компонентов
Интеграция компонентов и синтез системы

W – модель

Производство
Электроника
Механика
Программное обеспечение



Пример для «Системы систем»

V - модель

- **Разработка** продукта в полном **соответствии с требованиями**
- **Учет изменений** на всех уровнях разработки
- **Тесты, проверки, сертификация проверяют требования**
- Обеспечивается **сквозной мониторинг** разработки продукта
- **Результат:**
Конечный продукт соответствует требованиям на все 100%

Ключевые подходы к управлению проектами

Зачем нужны стандарты и фреймворки:

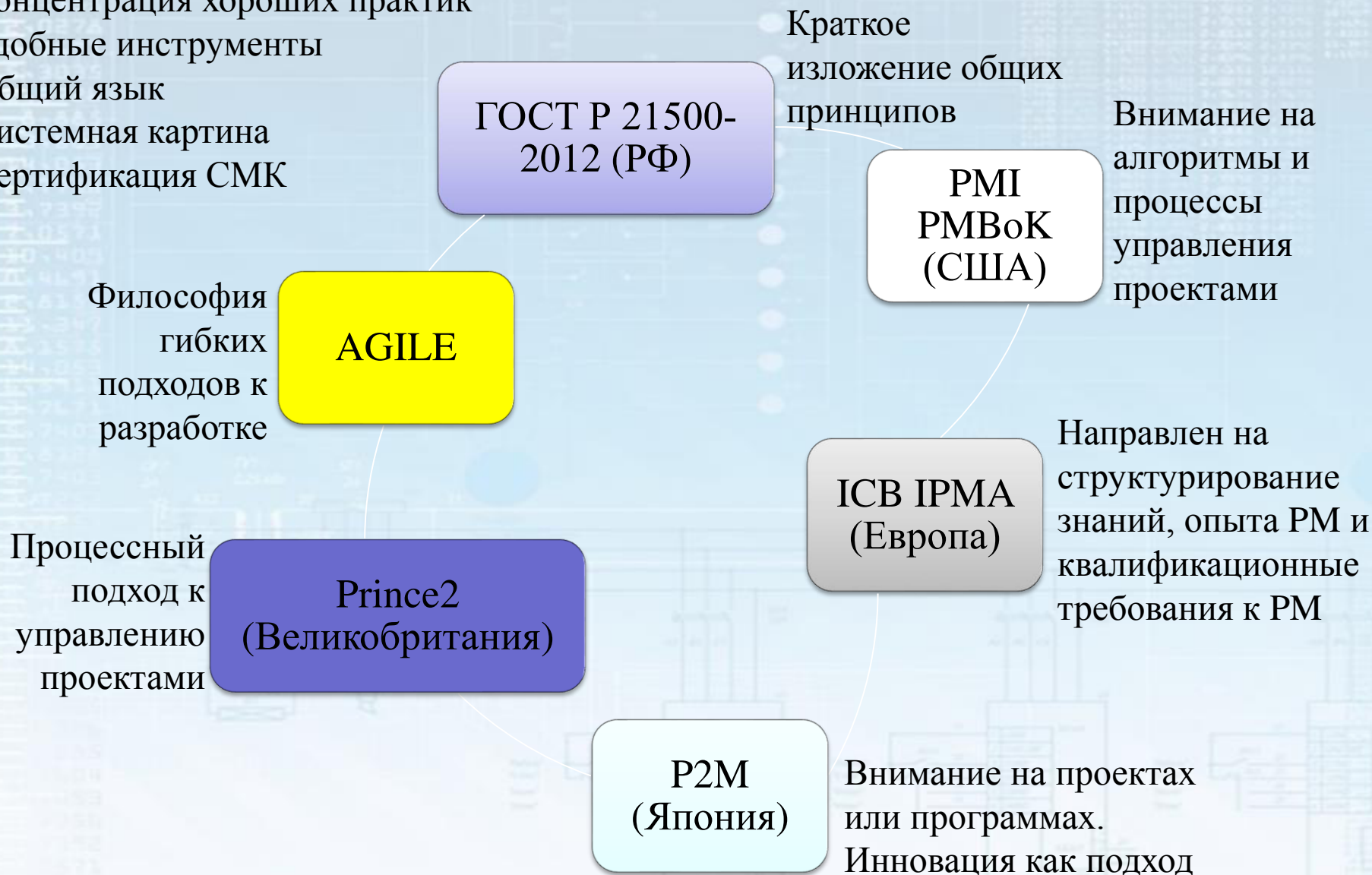
Концентрация хороших практик

Удобные инструменты

Общий язык

Системная картина

Сертификация СМК



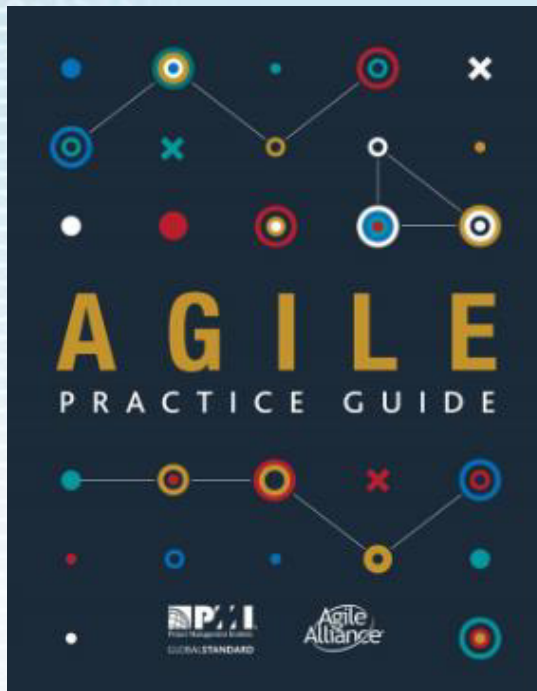
ГОСТы РФ по управлению проектами

ГОСТ Р ИСО/МЭК ТО 16326-2002;
ГОСТ Р ИСО 10006-2005;
ГОСТ Р 52806-2007;
ГОСТ Р 52807-2007;
ГОСТ Р 53892-2010;
ГОСТ Р 54 869-2011;
ГОСТ Р 54 870-2011;
ГОСТ Р 54 871-2011;
ГОСТ Р ИСО 21500-2014.

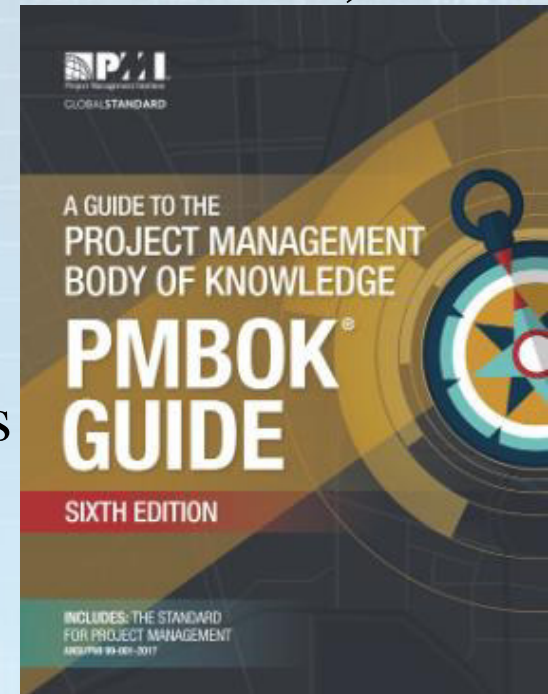


PMI (Project management institute)

- **PMBoK 6 (2017)**
- Agile Practice Guide (2017)
- Practice Standard for Project Risk Management
- Practice Standard for Work Breakdown Structures



- ISO 21500: 2012, Guidance on project management
- ISO 21503: 2017, Guidance on programme management
- ISO 21504: 2015, Guidance on portfolio management
- ISO 21505: 2017, Guidance on governance



Инициирование



Планирование



Исполнение



Мониторинг и
управление



Завершение

PMBoK - 10 областей знаний



Области знаний состоят из процессов

запомним



- ■ ■ ■ ■ ■ управление интеграцией
- ■ ■ ■ ■ ■ управление содержанием
- ■ ■ ■ ■ ■ управление временем
- ■ ■ управление стоимостью
- ■ ■ управление качеством
- ■ ■ управление HR
- ■ ■ ■ ■ ■ управление коммуникациями
- ■ ■ ■ ■ ■ управление рисками
- ■ ■ ■ ■ ■ управление закупками
- ■ ■ ■ ■ ■ управление заинтересованными лицами

Пример процесса: «создать расписание», «определить себестоимость работ», «инспектировать качество» и т.п.

PMBoK 6 (2017г) - 49 процессов

Инициирование Планирование		Исполнение		Мониторинг и управление Завершение	
◆	◆	◆ ◆	◆ ◆	◆	Управление интеграцией проекта
	◆ ◆ ◆ ◆		◆ ◆		Управление содержанием проекта
	◆ ◆ ◆ ◆ ◆		◆		Управление расписанием проекта
	◆ ◆ ◆		◆		Управление стоимостью проекта
	◆	◆	◆		Управление качеством проекта
	◆ ◆	◆ ◆ ◆	◆		Управление ресурсами проекта
	◆	◆	◆		Управление коммуникациями проекта
	◆ ◆ ◆ ◆ ◆	◆	◆		Управление рисками проекта
	◆	◆	◆		Управление поставками проекта
◆	◆	◆	◆		Управление стэйкхолдерами

Содержание устава проекта

Краткое описание целей,
базовых ограничений по срокам и стоимости



Все роли



Спонсор



Остальные заинтересованные лица



Заказчик и пользователи



Менеджер проекта



Команда



Остальные заинтересованные

Жизненный цикл проекта



Управление содержанием. Требования

- **Требование** - единичная задокументированная необходимость
- **Функциональные требования** описывают точное поведение (функционирование) системы, т.е. «*ЧТО система должна делать*»
- **Нефункциональные требования** описывают насколько хорошо это поведение должно исполняться

Инжиниринг требований =

Формирование требований +

Управление требованиями

Этапы работы с требованиями

В процессе Планирования:

1. сбор и формирование требований
2. декомпозиция и детализация требований
3. учет ограничивающих факторов
4. структуризация требований

В процессе Исполнения:

5. изменение требований



Мониторинг:

6. проверка реализации требований

Проблемы анализа требований

Проблемы заинтересованных сторон (stakeholders)

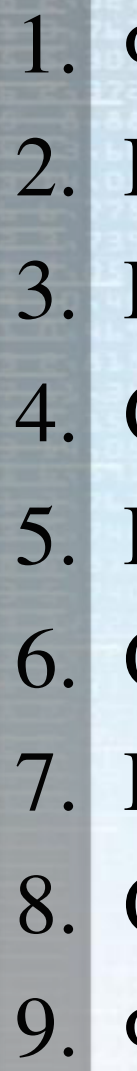
- пользователи не понимают то, что они хотят, или у пользователей нет ясного представления об их требованиях;
 - пользователи не соглашаются с ранее записанными требованиями;
 - пользователи настаивают на новых требованиях после того, как стоимость и график работ были установлены;
 - коммуникация с пользователями является медленной;
 - пользователи часто не участвуют в обзорах требований или неспособны в них участвовать;
 - пользователи технически не подготовлены;
 - пользователи не понимают процесса разработки.
-
- Это может привести к ситуации, где пользовательские требования продолжают изменяться, даже когда система или разработка новой продукции были начаты.

Проблемы анализа требований

Проблемы инженеров / разработчиков

- У технического персонала и конечных пользователей могут быть **различные мнения**. Они могут неправильно полагать, что они находятся во взаимопонимании, пока готовое изделие не будет отправлено.
- Инженеры и разработчики могут попытаться **подкорректировать требования** чтобы они соответствовали существующей системе или модели, вместо того, чтобы разработать систему, соответствующую потребностям клиента.
- Анализ требований зачастую может выполняться инженерами, а не персоналом с навыками работы с людьми и знаниями проблемной области.

Общая последовательность планирования содержания, сроков и стоимости

- 
1. Формирование требований
 2. Подготовка концепции
 3. Разработка иерархической структуры работы
 4. Связывание задач в последовательности
 5. Назначение ресурсов на задачи
 6. Определение сроков задач и этапов
 7. Привязка к реальному календарю
 8. Определение трудозатрат, материальных затрат
 9. Формирование расходной части бюджета

Agile-манифест

разработки программного обеспечения

- 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:

Больше ценим

Люди и взаимодействие

Работающий продукт

Сотрудничество с заказчиком

Готовность к изменениям

процессы и инструменты

исчерпывающая документация

согласование условий контракта

следование первоначальному плану

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

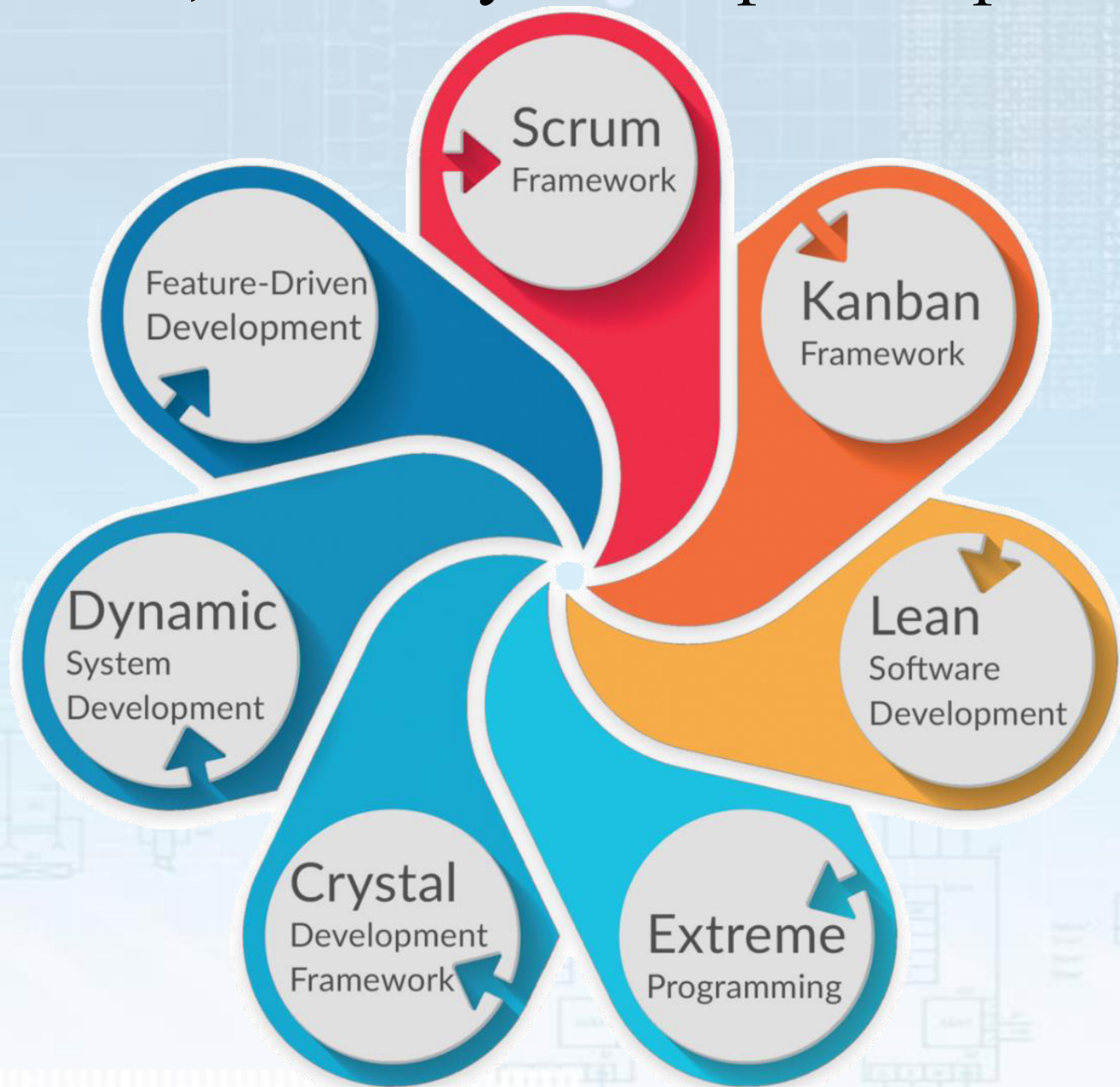
Основные принципы Agile



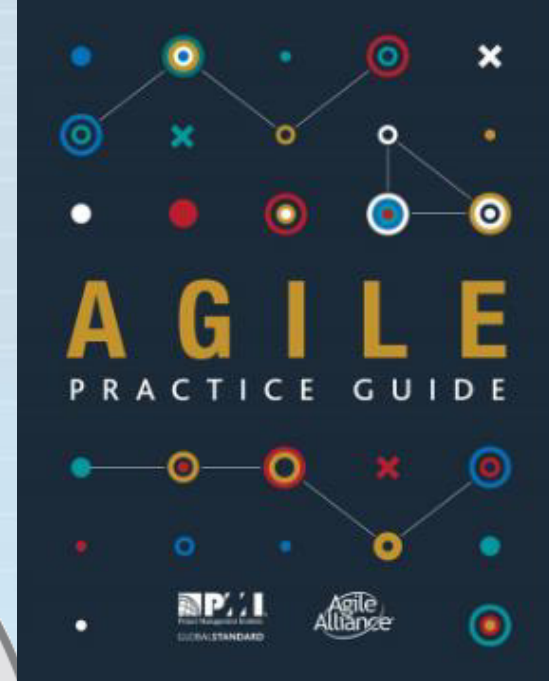
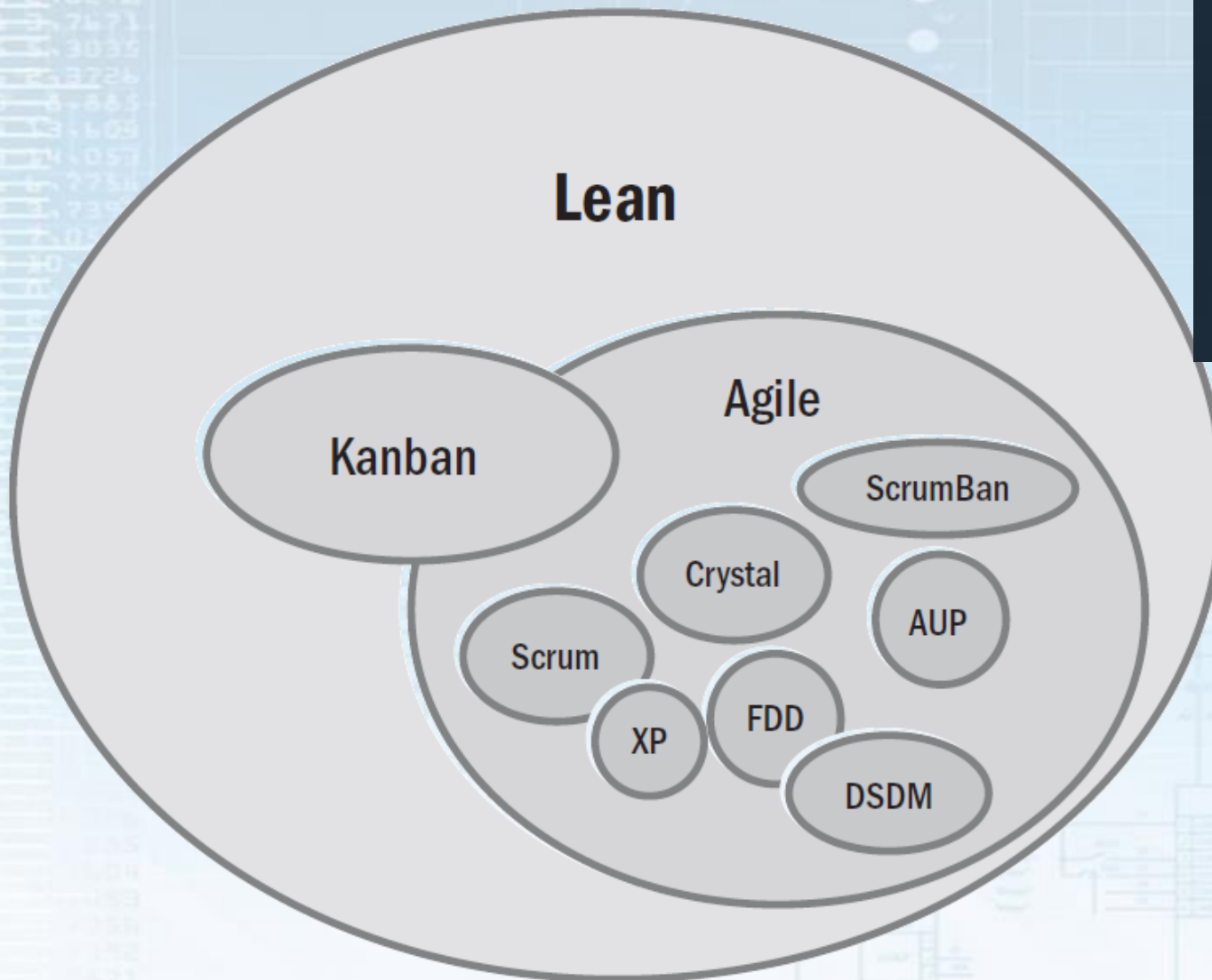
Работающий продукт — основной показатель прогресса

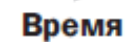
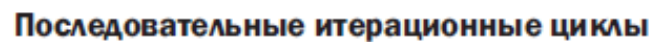
1. Наивысший приоритет - удовлетворение потребностей заказчика, благодаря регулярной и ранней поставке продукта.
2. Работающий продукт следует выпускать как можно чаще (2 недели .. 1 мес).
3. Разработчики и бизнес должны ежедневно работать вместе.

Методологии, использующие философию Agile



Agile Practice Guide (2017)





Перспектива PMBOK 7

**Стандарты,
основанные на
процедурах**

- PMBOK6
(49 процедур)



**Стандарты,
основанные на
принципах**

- PMBOK7

ПРИНЦИП фиксирует и обобщает общепринятую цель для практики дисциплин и функций управления портфелем, программой и проектом

Возможная перспектива PMBOK 7

Стандарты,
основанные на
процедурах

- PMBOK6
(49 процедур)



Стандарты,
основанные на
принципах

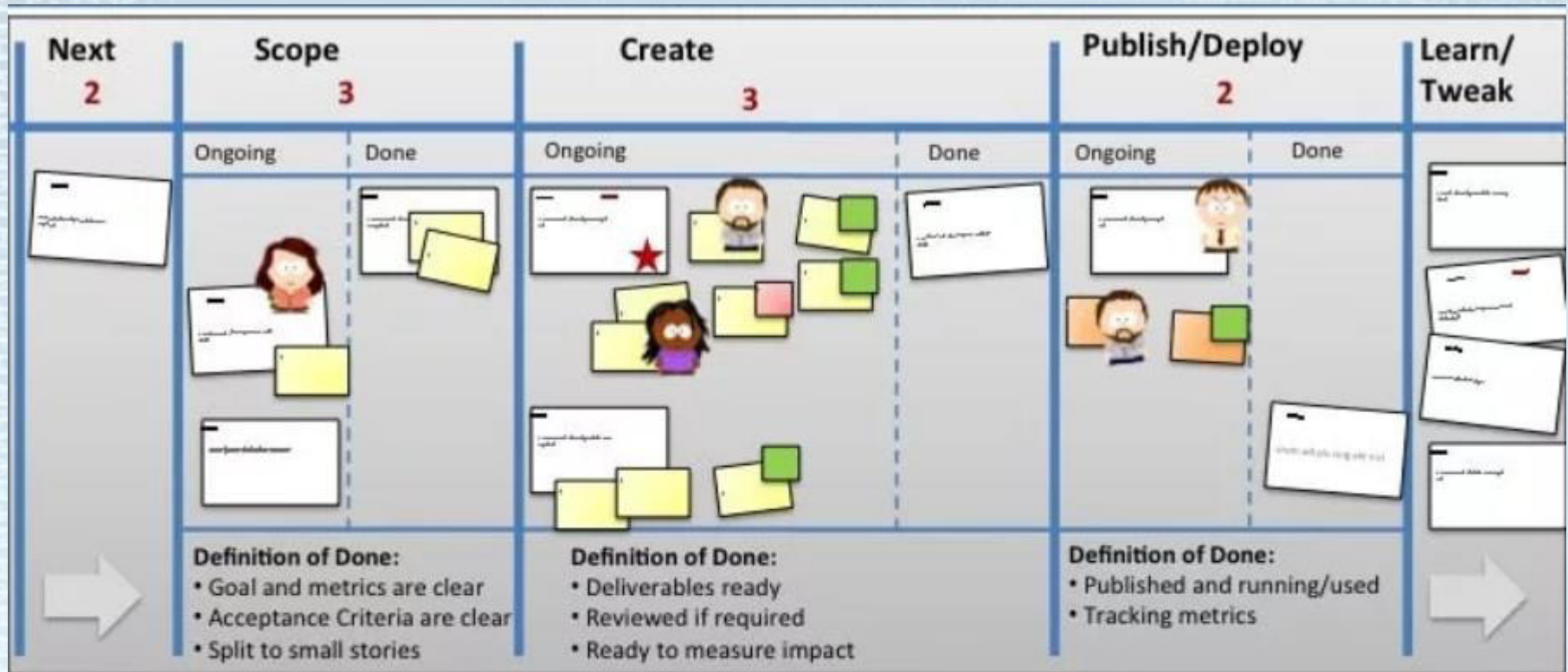
- PMBOK7

ПРИНЦИП фиксирует и обобщает общепринятую цель для практики дисциплин и функций управления портфелем, программой и проектом

NUPP (Nearly Universal Principles of Projects)

- NUPP - это набор почти универсальных проектных принципов, которым лучше следовать во всех проектах, независимо от используемых методологий и подходов, для достижения максимального если мы хотим добиться успеха
- NUP1: выбирай результаты и истину, а не привязанности
- NUP2: береги и оптимизируй энергию и ресурсы
- NUP3: всегда будь проактивен
- NUP4: помни, что прочность цепи определяется по самому слабому звену
- NUP5: не делай ничего без четкой цели
- NUP6: используй воспроизводимые элементы

Канбан



Feature / story

Date when added to board

2017-08-20

2017-09-30

(description)



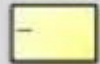
Hard deadline
(if applicable)

★ = priority

★★★ = panic

Who is analyzing / testing right now

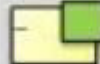
Task / defect



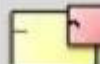
= task



= defect



= completed



= blocked



= who is doing this right now

What to pull first

- **Panic** ★★★
(should be swarmed and kept moving. Interrupt other work and break WIP limits as necessary)
- **Priority** ★
- **Hard deadline** (only if deadline is at risk)
- **Oldest**

Adapted from Henrik Kniberg Crisp.se

SCRUM



Входные данные от руководства, команды, заинтересованных сторон, клиентов и пользователей



Владелец компании



Скрам-команда

Нумерованный список задач: функции, контент, ...

Бэклог проекта

Команда выбирает список приоритетных задач на спринт

Планирование Спринта

Список задач на спринт

Бэклог спринта



Диаграмма сгорания и скрам доска

Скрам мастер



Ежедневные скрам собрания



Обзор спринта



Демонстрация готового продукта



Ретроспектива спринта

Каждые 24 часа

1-4
Недельный спринт

Дата окончания спринта и команда не меняются

Модель Кеневин (Дэйв Сноуден)

cognitive-edge.com

**ЗАПУТАННЫЕ
системы**

**УПОРЯДОЧЕННЫЕ
сложные**

Беспорядок

**ХАОТИЧНЫЕ
системы**

**УПОРЯДОЧЕННЫЕ
простые**

Модель Кеневин (Дэйв Сноуден)

ЗАПУТАННЫЕ

Новые
практики

СИСТЕМЫ

Agile
Распределенное
руководство

Измеряем
Ощущаем
Реагируем

УПОРЯДОЧЕННЫЕ сложные

Хорошие
практики

Ощущаем
Анализируем
Реагируем

PMBoK
Набегающая
волна

ХАОТИЧНЫЕ

Новейшие
практики

СИСТЕМЫ

Do & Fix
«Делай что можешь
и будь что будет»

Действуем
Ощущаем
Реагируем

УПОРЯДОЧЕННЫЕ простые

Лучшая
практика

Ощущаем
Классифицируем
Реагируем

Водопад
Директивное
управление