КПО

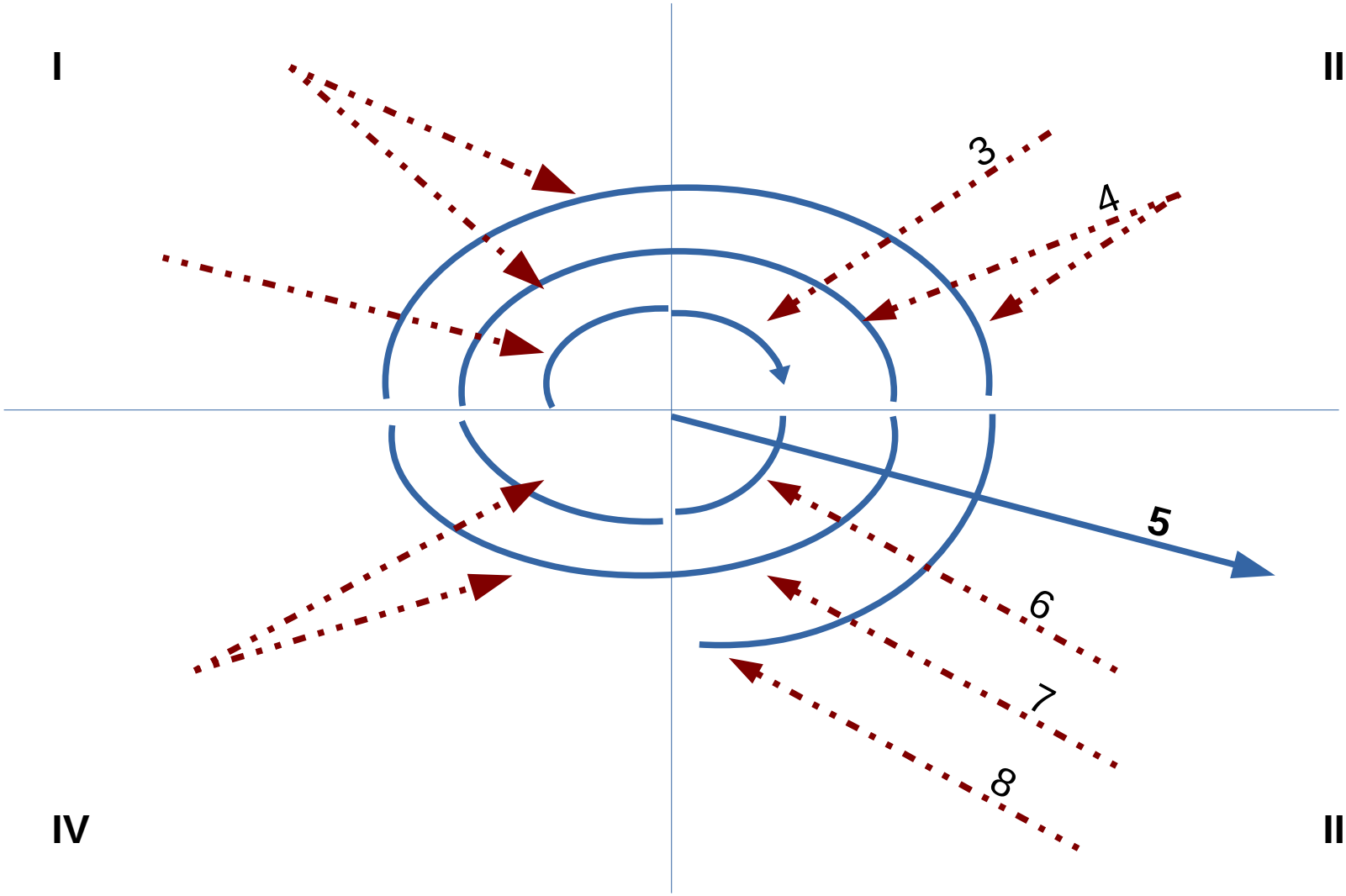
Современные стратегии конструирования программного обеспечения

Лекция №2

(версия 1.0)

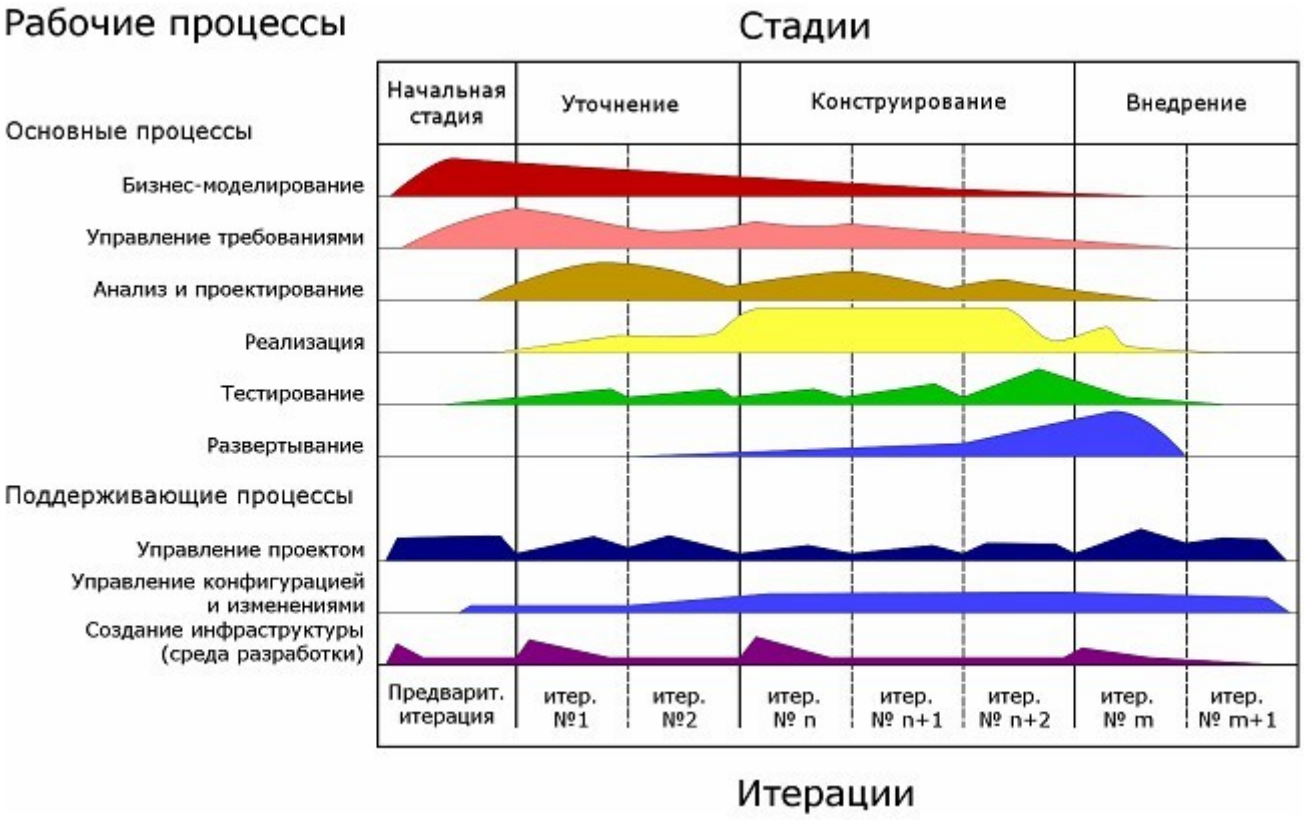
спиральная модель

Барри Боэм, 1988



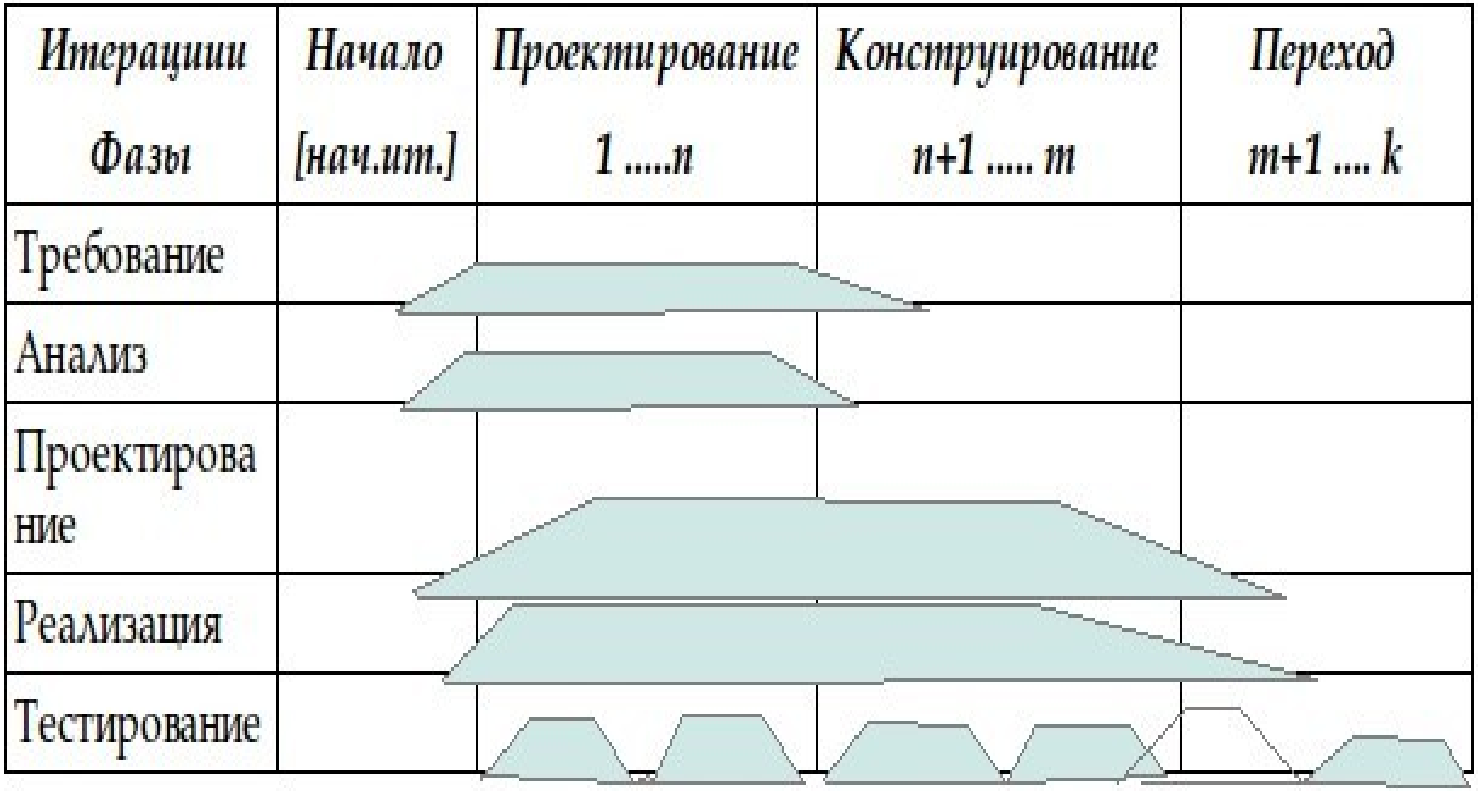
USDP Якобсон, Буч и Рамбо, 1999

UNIFIED SOFTWARE DEVELOPMENT PROCESS



USDP Якобсон, Буч и Рамбо, 1999

UNIFIED SOFTWARE DEVELOPMENT PROCESS



USDP

USDP описывает шесть моделей:

- Модель *вариантов использования*
- *Аналитическая* модель
- Модель *проектирования*
- Модель *развертывания*
- Модель *реализации*
- Модель *тестирования*

Гибкие технологии

Гибкая методология разработки программного обеспечения ориентирована на использование итеративного подхода, при котором программный продукт создается постепенно, небольшими шагами, включающими реализация определенного набора требований.

Ключевые постулаты гибкой разработки

- Люди и их взаимодействие
- Создание работающего программного обеспечения
- Сотрудничество с заказчиком
- Реакция на изменение

Ключевые правила поведения

- Уважение мнения каждого участника команды
- Быть правдивым при любом общении
- Прозрачность всех данных, действий и решений
- Уверенность, что каждый участник поддержит команду
- Приверженность команде и её целям

Принципы гибкой методологии

1) Высшим приоритетом следует считать удовлетворение пожеланий заказчика
2) Не игнорировать изменения требований
3) Частое создание новых работающих версии ПО
4) Заказчики и разработчики должны работать совместно
5) Проекты должны воплощать в жизнь целеустремленные люди
6) Эффективный метод передачи информации – разговор лицом к лицу

Принципы гибкой методологии (продолжение)

7) Работающая программа – основной показатель прогресса в проекте
8) *Гибкие процессы способствуют долгосрочной разработке*
9) Непрестанное внимание к качественному проектированию
10) Простота
11) *Самые лучшие решения выдают самоорганизующиеся команды*
12) Команда должна регулярно задумываться над тем, как стать ещё более эффективной

Гибкие методологии

Agile Modeling	Adaptive software development
Agile Unified Process	Feature driven development
OpenUP	Getting Real
Agile Data Method	MSF fog Agile Software Development
DSDM	**Scrum**
Extreme programming	

XP

Кент Бек, 1999

eXtreme Programming

	XP-экстремум	*XP-реализация*
Проверка кода	Код проверяется всё время	Парное программирование
Тестирование	Тестирование выполняется всё время, даже с помощью заказчика	Тестирование модуля, функциональное тестирование

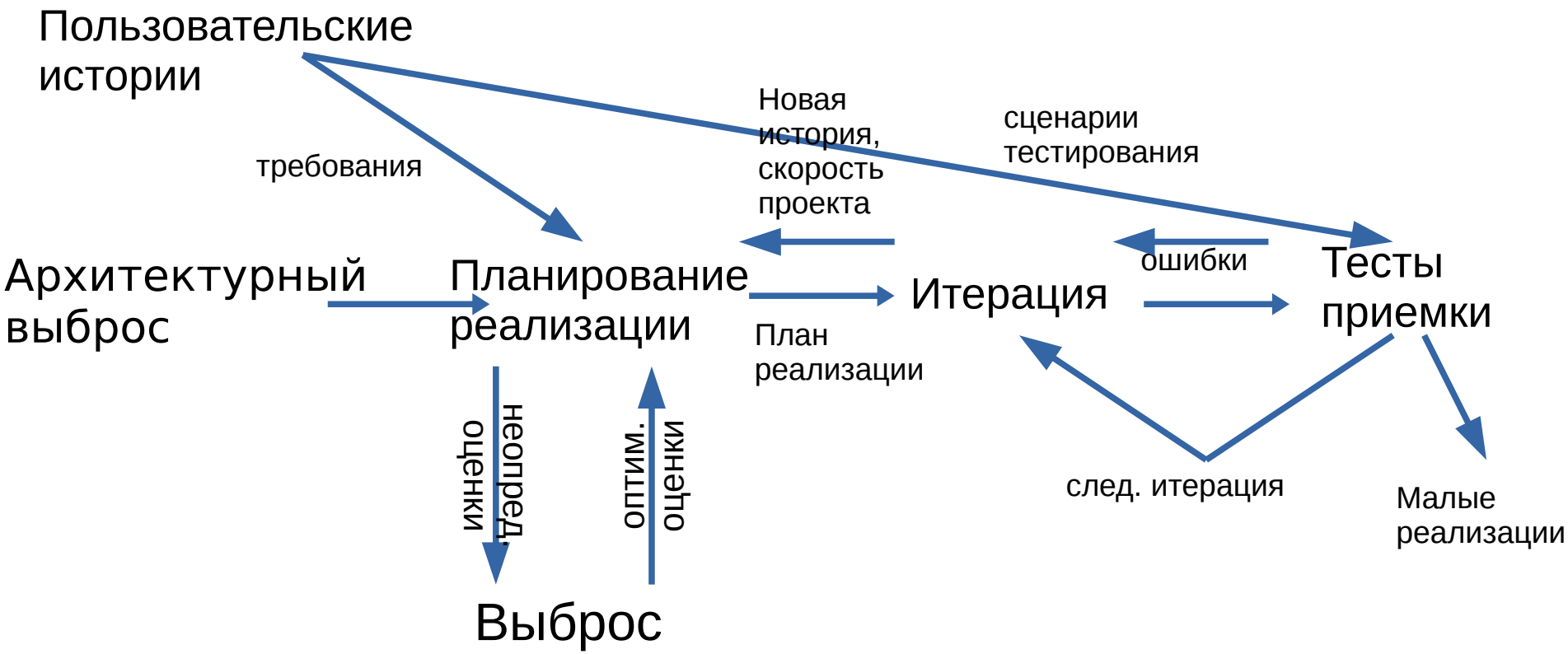
XP

Проектирование	Проектирование – часть ежедневной работы разработчика	Реорганизация (refactoring)
Простота	Для системы выбирается простейшее проектное решение, поддерживающее ее текущую функциональность	Самая простая вещь, которая могла бы работать. Принцип KISS. «Это вам не понадобится»
Архитектура	Каждый постоянно работает над уточнением архитектуры	Метафора

XP

Тестирование интеграции	Интегрируется и тестируется несколько раз в день	Непрерывная интеграция
Короткие итерации	Итерации предельно коротки – секунды, минуты, часы, а не неделя, месяц, год	Игра планирования

XP-реализация

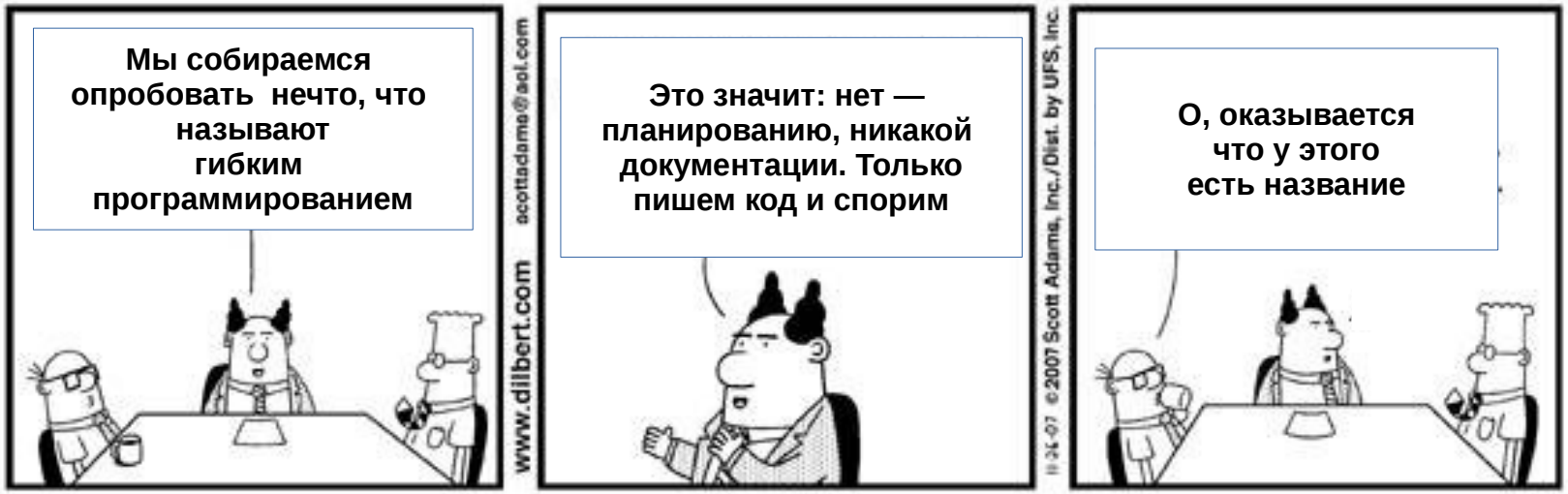


базис XP

1)Игра планирования (*Planning game*)
2)Частая смена версий (*Small releases*)
3)Метафора (*Metaphor*)
4)Простое проектирование
5)Тестирование (*TDD - Test Driven Development*)
6)Реорганизация (*Refactoring*)
7)Парное программирование
8)Коллективное владение кодом
9)Непрерывная интеграция.
10)40-часовая неделя.
11)Локальный заказчик.
12)Стандарты кодирования

Agile

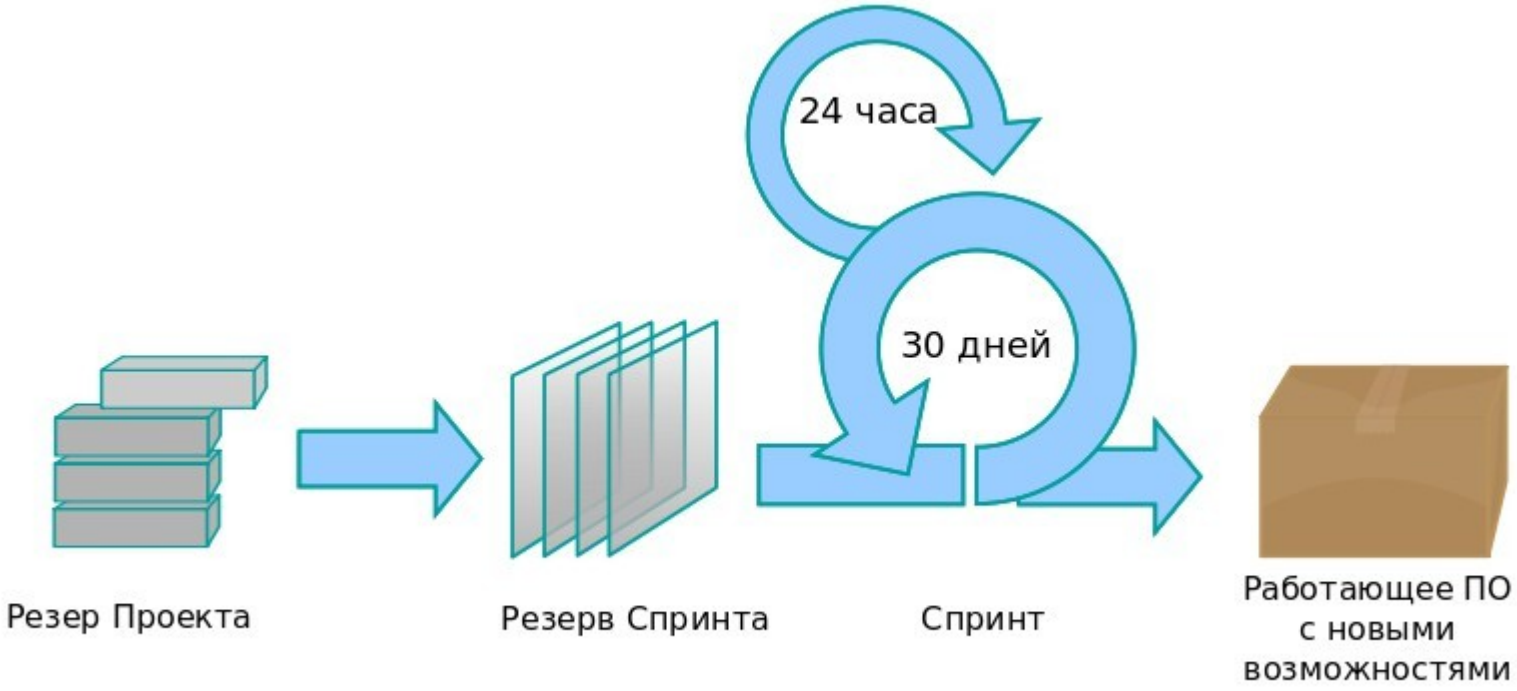
В феврале 2001 в штате Юта США был выпущен «Манифест гибкой методологии разработки программного обеспечения». Данный манифест был одобрен и подписан представителями методологий экстремального программирования, Crystal Clear, DSDM, Feature driven development, Scrum, Adaptive software development, Pragmatic Programming.



Scrum

Scrum — методология управления разработкой информационных систем для гибкой разработки программного обеспечения. Scrum чётко делает акцент на качественном контроле процесса разработки. Кроме управления проектами по разработке ПО Scrum может также использоваться в работе команд поддержки программного обеспечения (software support teams), или как подход управления разработкой и сопровождением программ: Scrum of Scrums.

Scrum



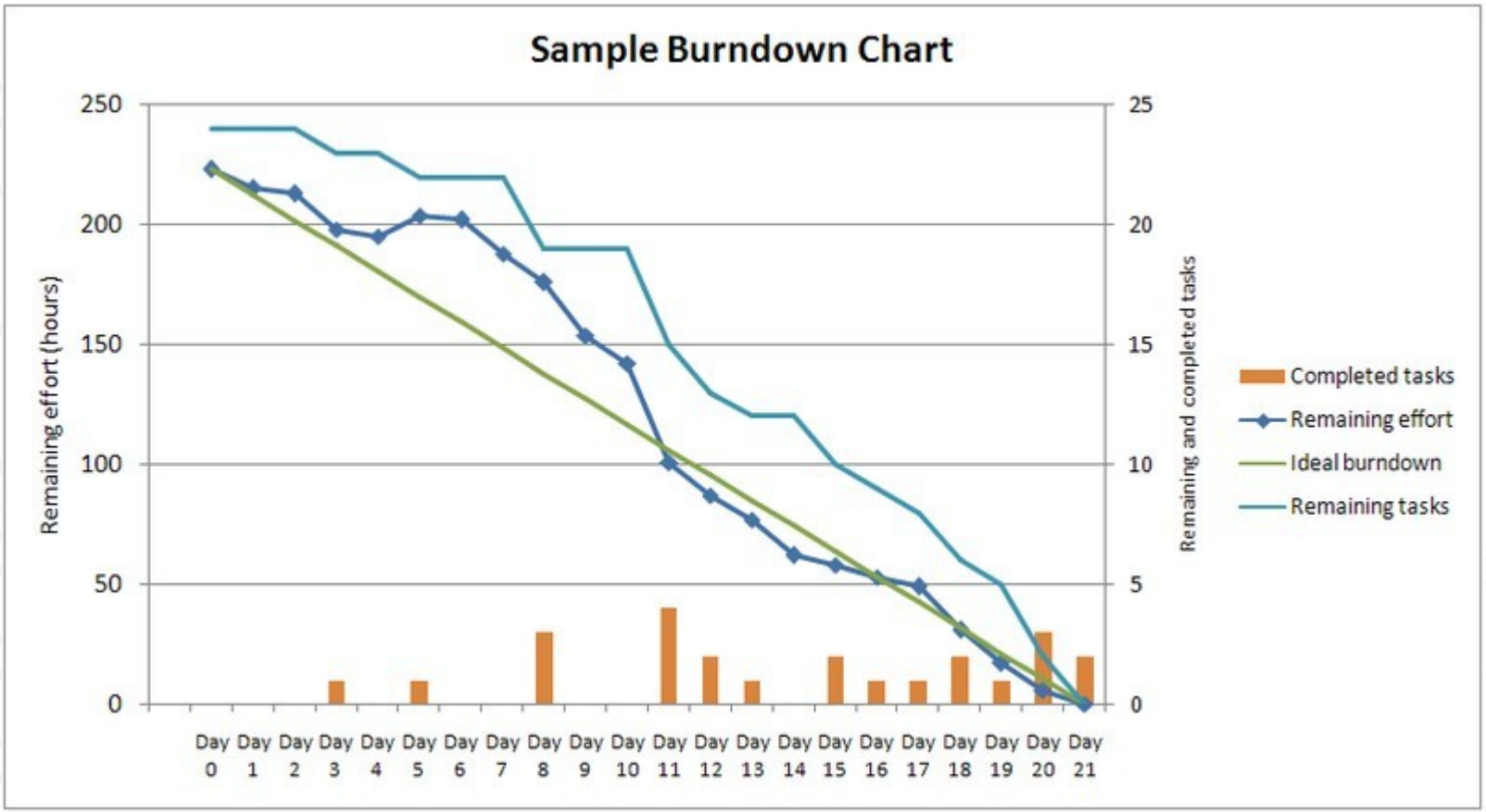
Scrum

Спринт — итерация в скрам, в ходе которой создаётся функциональный рост программного обеспечения. Жёстко фиксирован по времени. Длительность одного спринта от 2 до 4 недель.

Scrum

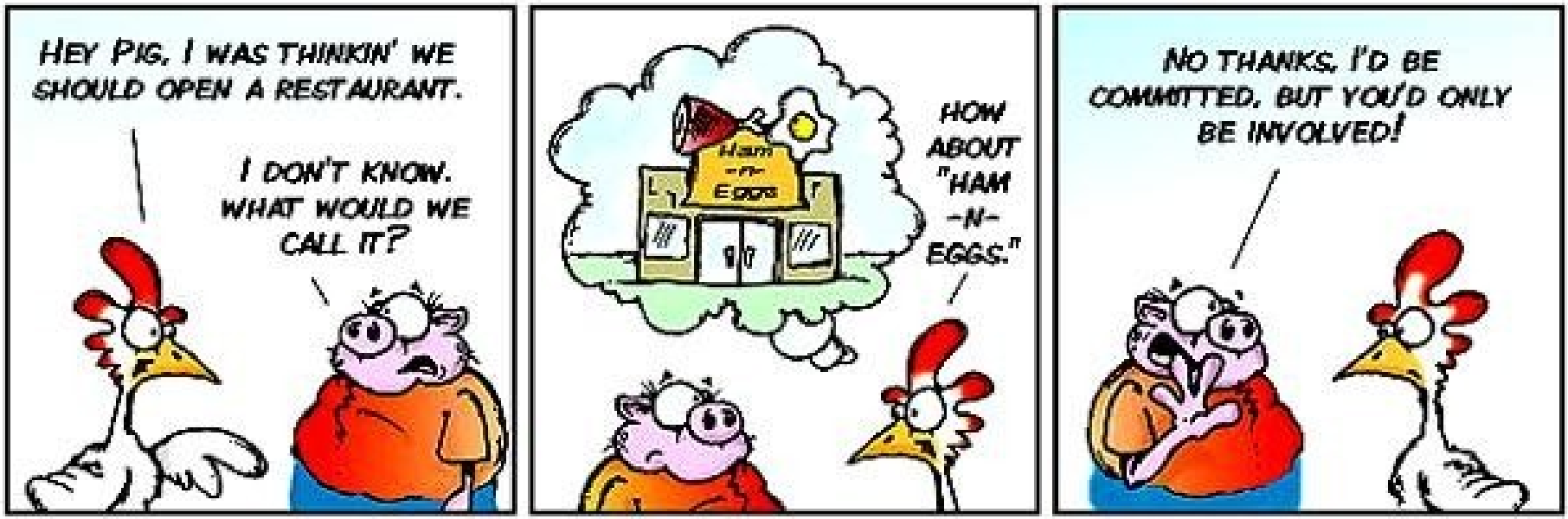
Резерв проекта — это список требований к функциональности, упорядоченный по их степени важности, подлежащих реализации. Элементы этого списка называются «пожеланиями пользователя» (user story) или элементами резерва (backlog items). Резерв проекта открыт для редактирования для всех участников скрам процесса.

Scrum



Scrum (роли)

По методике Scrum в производственном процессе есть определенные роли, разбитые на 2 группы «свиней» и «кур».



Scrum (роли)

Свиньи:

- Скрам-мастер (ScrumMaster)
- Владелец проекта (Product Owner)
- Скрам-команда (Scrum Team)

Scrum (роли)

Куры:

- Пользователи (Users)
- Клиенты, Продавцы (Stakeholders)
- Управляющие (Managers)
- Эксперты-консультанты (Consulting Experts)

Scrum (встречи)