Занятие 2. Жизненный цикл ПО и его этапы

Гадияк Никита Андреевич
Старший инженер по тестированию ПО

План занятия

1. **Этапы жизненного цикла ПО**
2. **Основные модели разработки ПО и роль тестирования в процессе разработки**
3. **Краткое описание цикла тестирования ПО**

Компания хороших идей
Bellintegrator

Жизненный цикл программного обеспечения

Жизненный цикл ПО
Стандарты

- ГОСТ 34.601-90
- ISO/IEC 12207
- Custom Development Method
- Rational Unified Process (RUP)
- Microsoft Solution Framework (MSF)
- Extreme Programming (XP).

Жизненный цикл ПО
ISO/IEC 12207

ISO/IEC 12207:2008

Systems and software engineering — Software life cycle processes — стандарт ISO, описывающий процессы жизненного цикла программного обеспечения.

Жизненный цикл ПО
ISO/IEC 12207. Основные процессы

- Приобретение;
- Поставка;
- Разработка;
- Эксплуатация;
- Сопровождение.

Жизненный цикл ПО
ISO/IEC 12207. Вспомогательные процессы

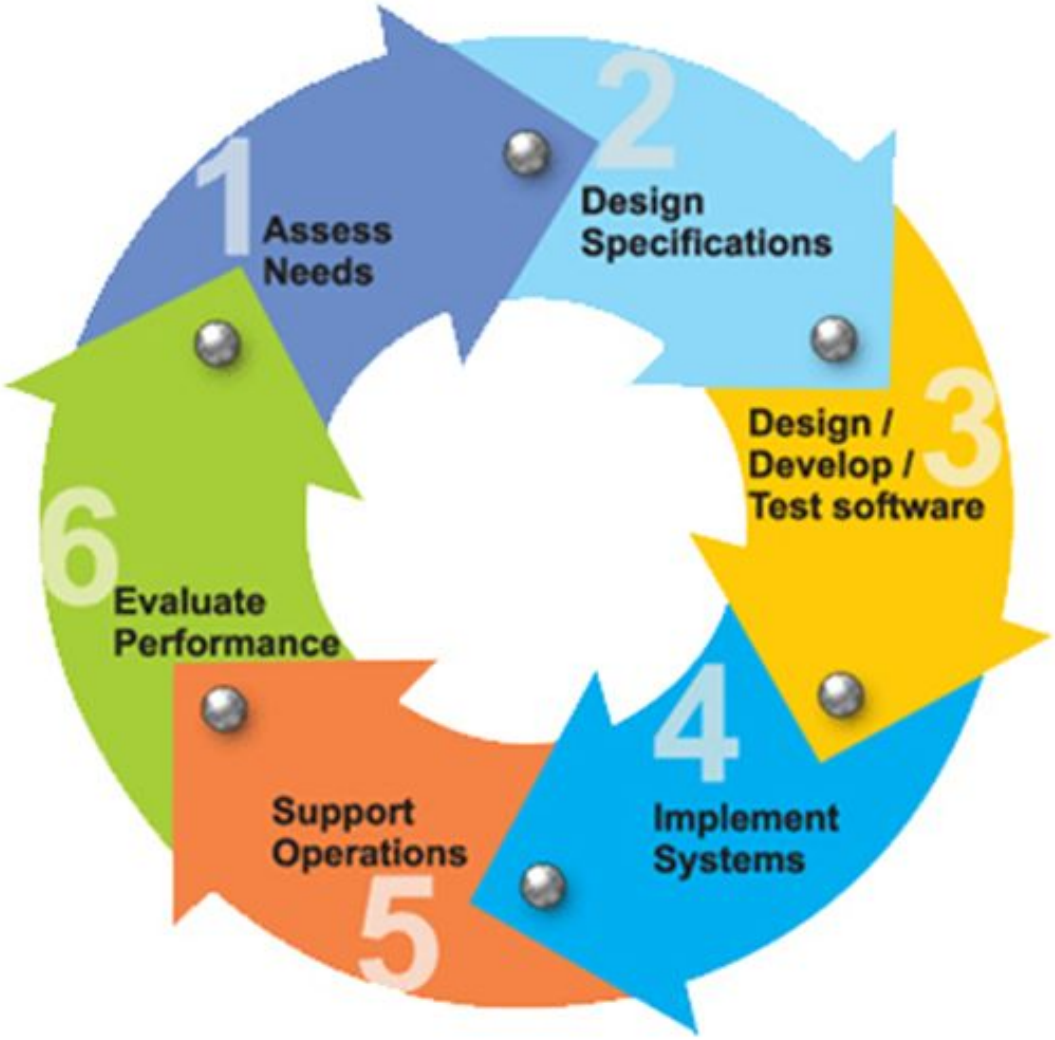
- Документирование;
- Управление конфигурацией;
- Обеспечение качества;
- Разрешение проблем;
- Аудит;
- Аттестация;
- Совместная оценка;
- Верификация.

Жизненный цикл ПО
ISO/IEC 12207. Организационные процессы

- Создание инфраструктуры;
- Управление;
- Обучение;
- Усовершенствование.

Компания хороших идей
Bellintegrator

Модели разработки ПО



Модели разработки ПО
Что такое модель разработки

Под моделью обычно понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении жизненного цикла.

- **Этапы**:
  - **Анализ осуществимости; стратегическое планирование; анализ требований;**
  - **проектирование (предварительное и детальное);**
  - **кодирование (программирование);**
  - **отладка и тестирование; интеграция;**
  - **Внедрение; эксплуатация и сопровождение.**

- **Результаты работ на каждом этапе**
- **Ключевые события (точки принятия решений**)

Модели разработки ПО
характеристики

Эффективность

- затраты/бюджет
- сроки

Прозрачность

- статус работ известен в любой момент проекта

Предсказуемость

- реальные трудозатраты и сроки находятся в запланированных (сметных) пределах

Управляемость

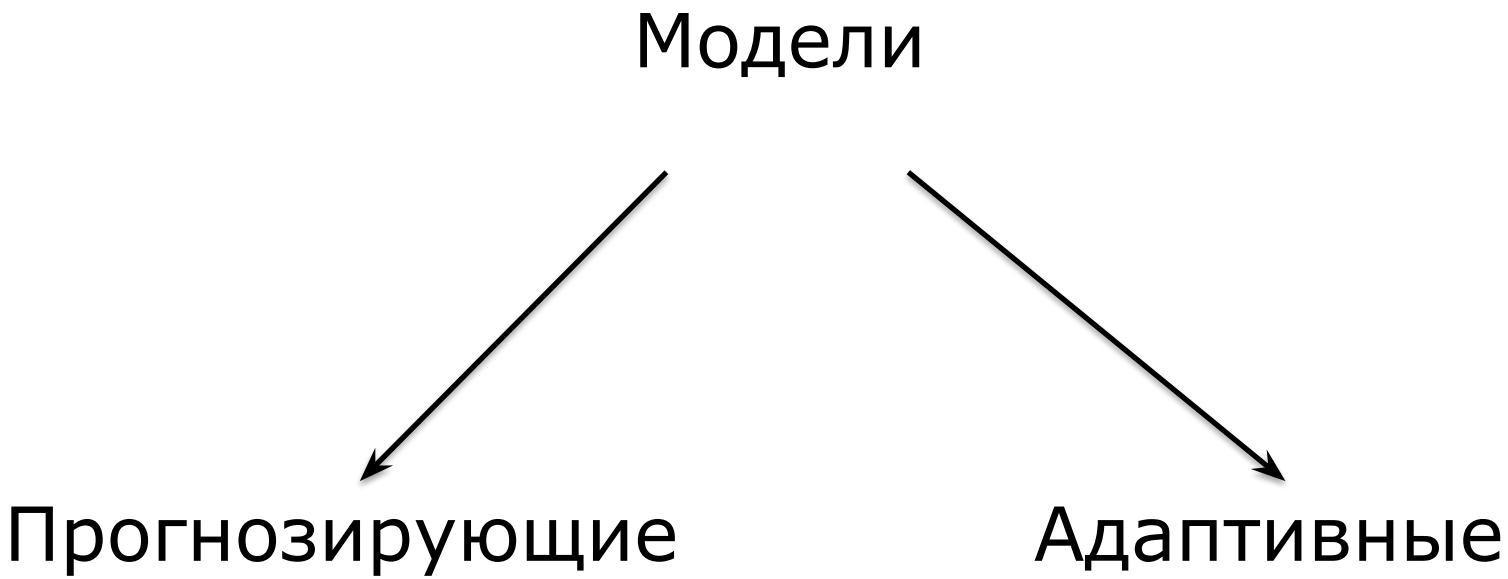
- возможность внесения корректив по ходу проекта (изменяющиеся требования и др.)

Сдерживание рисков

- устойчивость к влиянию внешних факторов

Компания хороших идей
Bellintegrator

Модели разработки ПО
характеристики



Модели разработки ПО
Популярные модели

Прогнозирующие модели:

- Прямая разработка
- Водопадная (каскадная) модель
- V-модель
- Итерационные модели

Компания хороших идей Bellintegrator

Модели разработки ПО
Популярные модели

Адаптивные модели (agile):

- Адаптивная разработка (ASD)
- Dynamic System Development Method (DSDM)
- Feature Driven Development (FDD)
- Crystal
- Экстремальное программирование
- SCRUM
- RUP

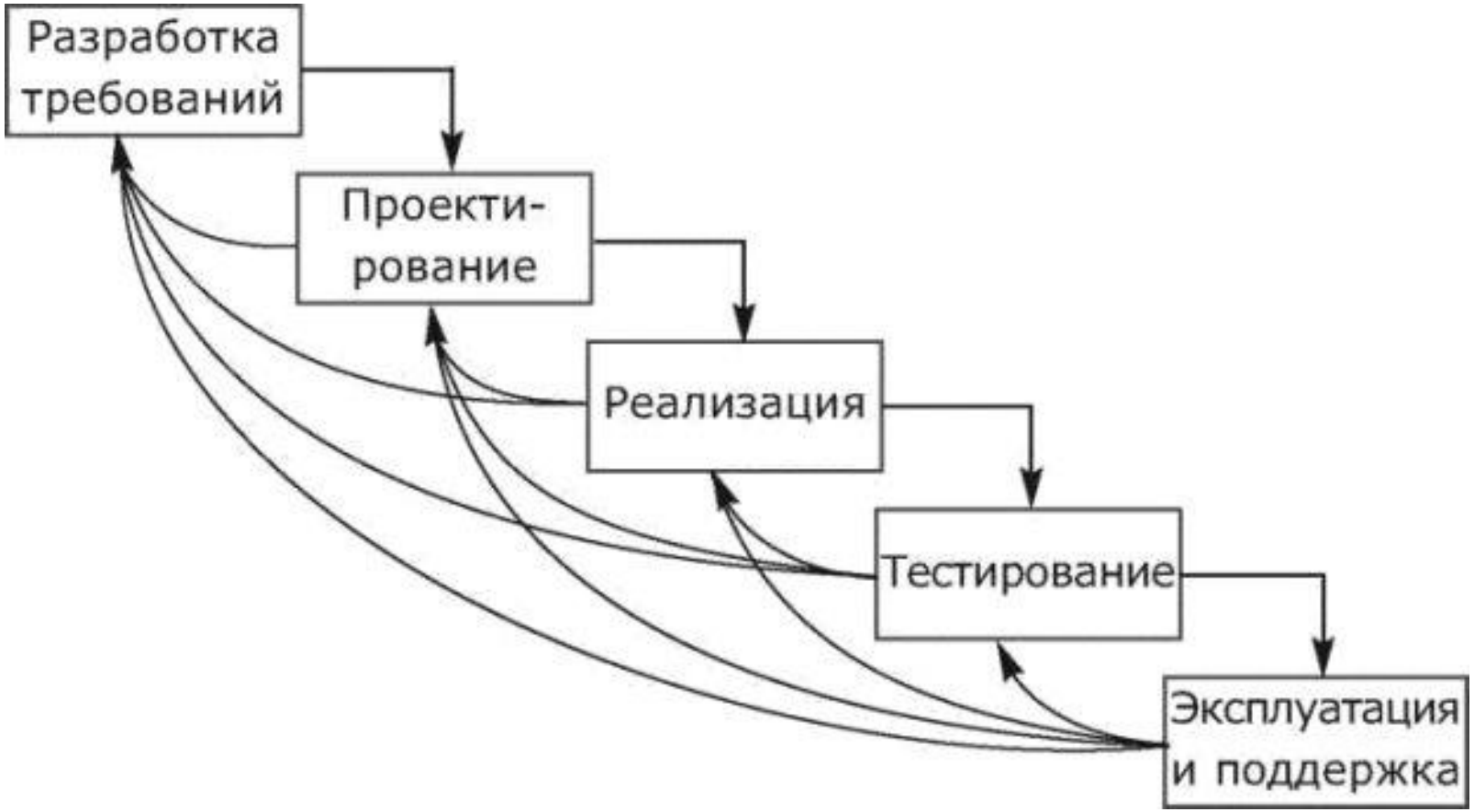
…

Компания хороших идей
Bellintegrator

Модели разработки ПО
Каскадная модель

Компания хороших идей
Bellintegrator

Модели разработки ПО
Каскадная модель



Модели разработки ПО
Каскадная модель

Компания хороших идей
Bellintegrator

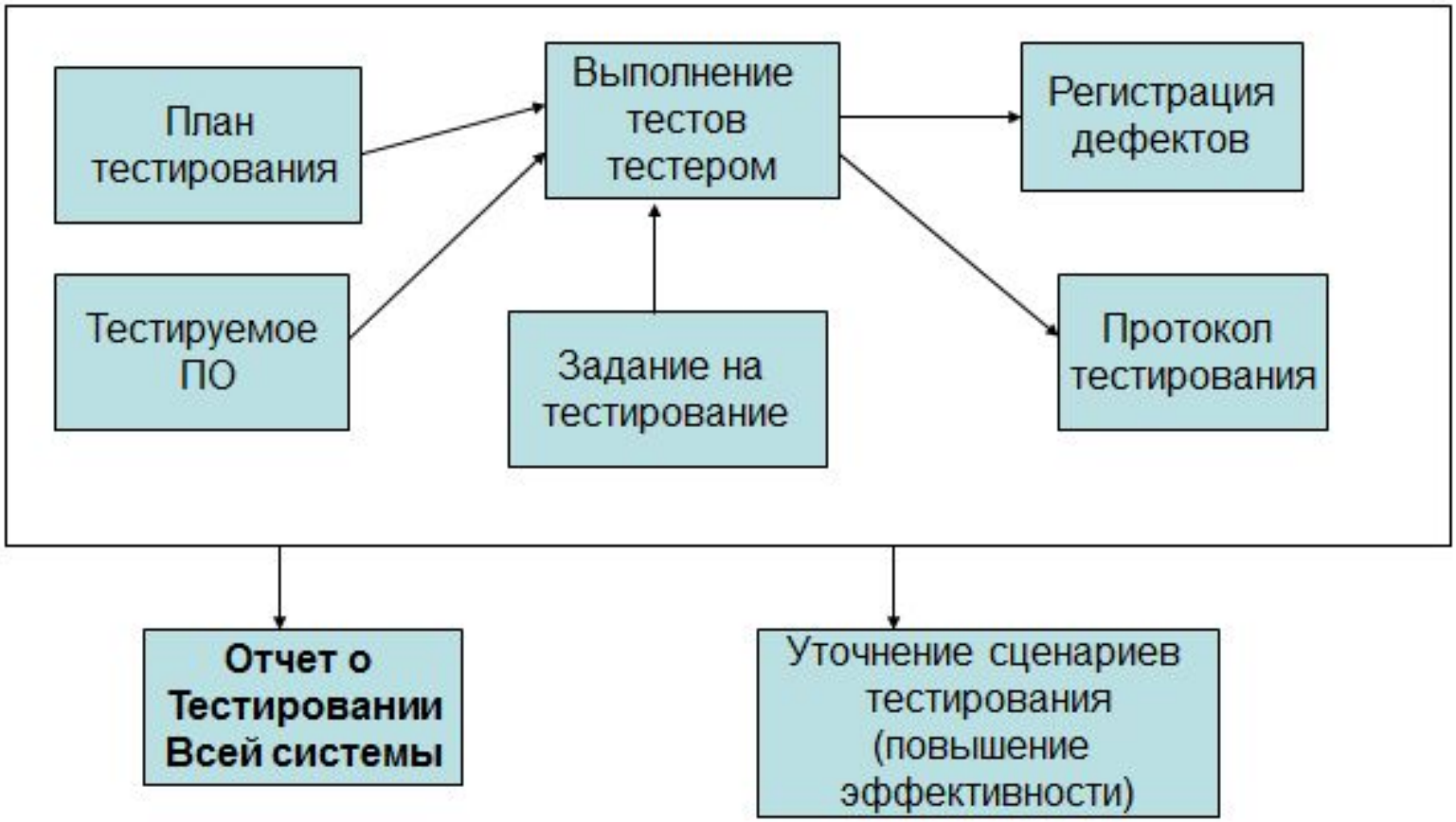
Особенность модели:

каждый следующий этап проектирования начинается после полного завершения работ по предыдущему этапу.

Компания хороших идей Bellintegrator

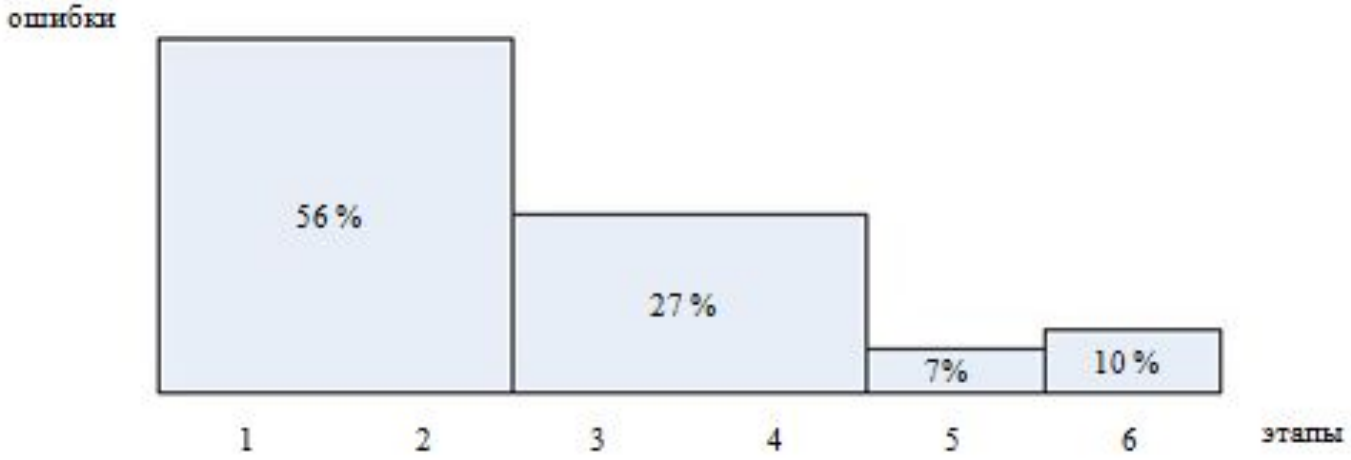
Модели разработки ПО

Тестирование в каскадной модели

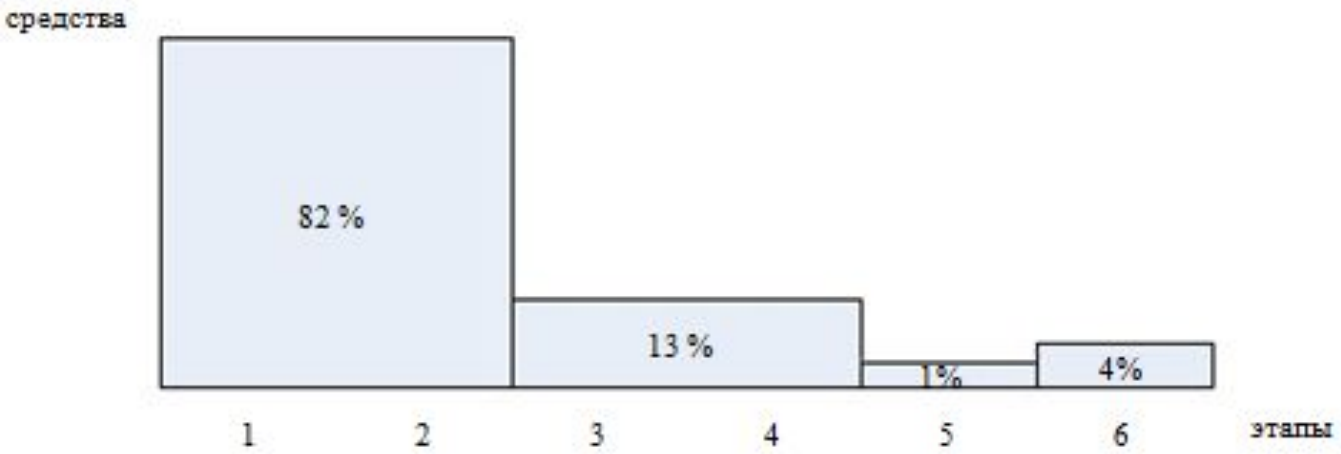


Модели разработки ПО
Каскадная модель

1



2



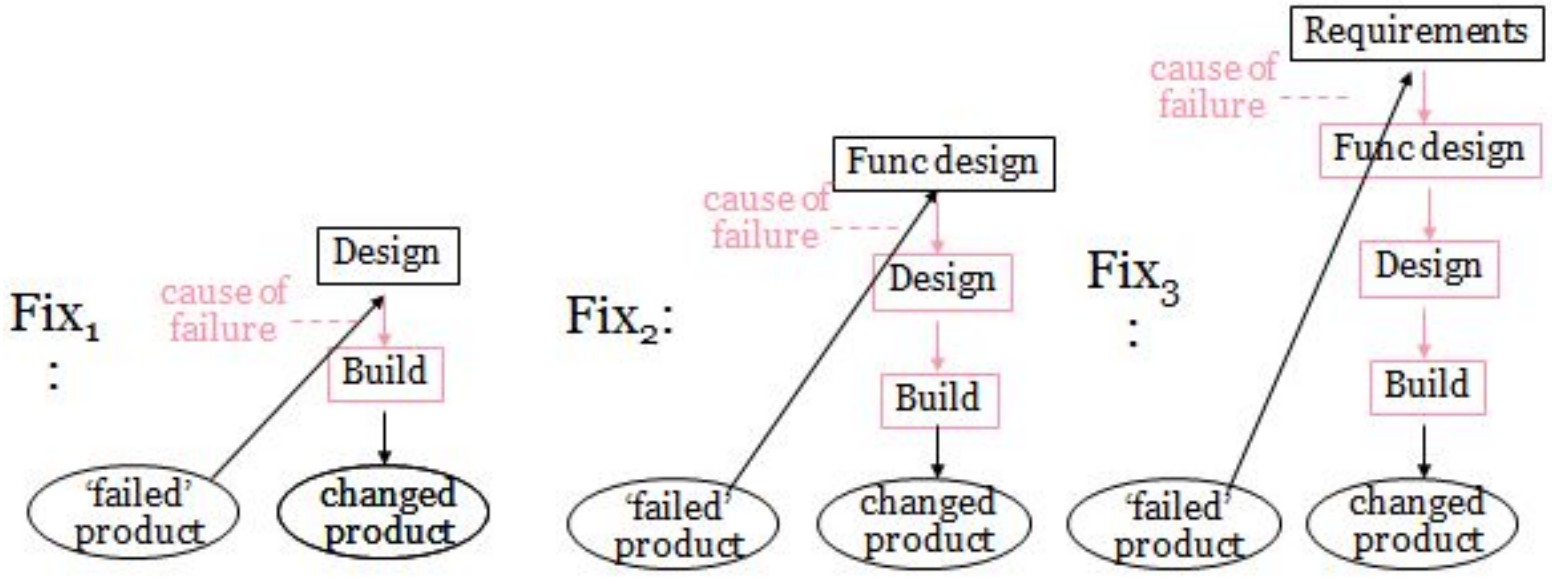
Компания хороших идей
Bellintegrator

Модели разработки ПО

V-модель разработки

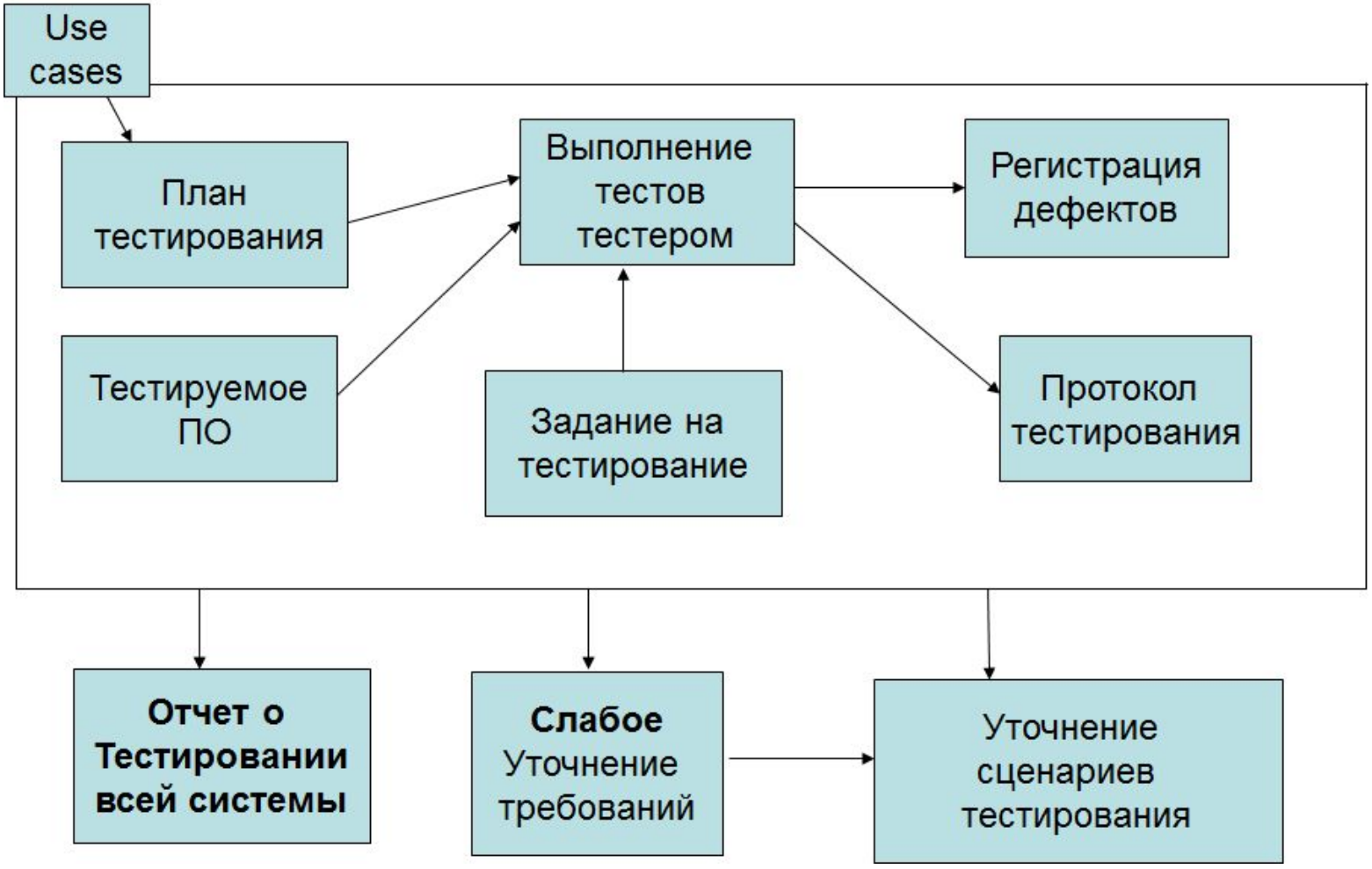
Модели разработки ПО
Вариации доработок

- Fix_1 - изъян в коде программы; переделывается только код
- Fix_2 - изъян в спецификациях (технический дизайн); меняются и спецификации, и код
- Fix_3 - изъян в архитектуре/функциональном дизайне; меняются архитектура, спецификации и код



Модели разработки ПО

Тестирование в V-модели

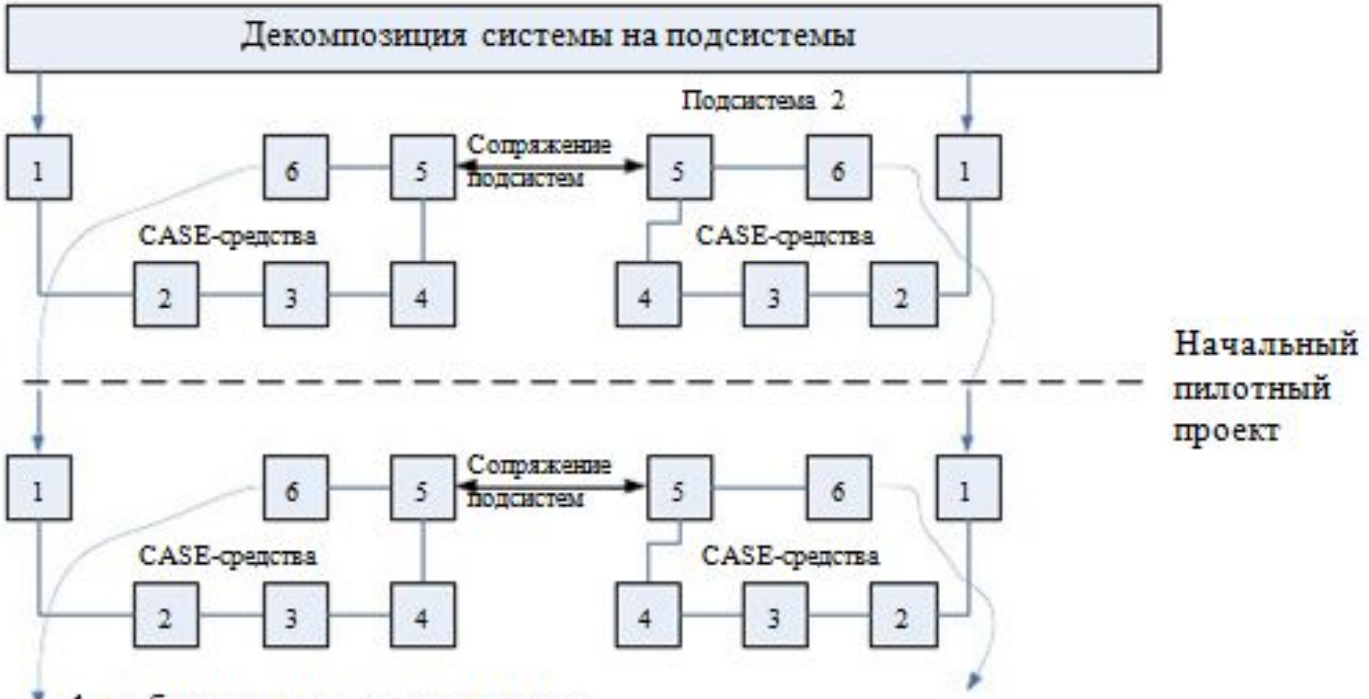


Модели разработки ПО
Эволюционные модели

- Мульти-каскад
  - последовательные каскады
  - параллельные каскады: одновременная разработка в нескольких направлениях;
  - требует интеграции; фактически подразумевается в V-модели
- Прототипирование
  - получить быстро нечто работающее, оценить и спланировать дальнейшую разработку
  - варианты: proof-of-concept; архитектурный прототип;
  - Стимуляция пользователя на участие в разработке
  - Поэтапное уточнение требований
- Наращивание, метод пробных разработок
  - первоначальная разработка постепенно наращивается
  - если прототип не отбрасывается, то прототипирование частный случай наращивания
  - Периодическая стабилизация продукта
  - Большой объем тестирования
  - Вовлеченность заказчика

Модели разработки ПО
Спиральная модель

Модели разработки ПО
Спиральная модель

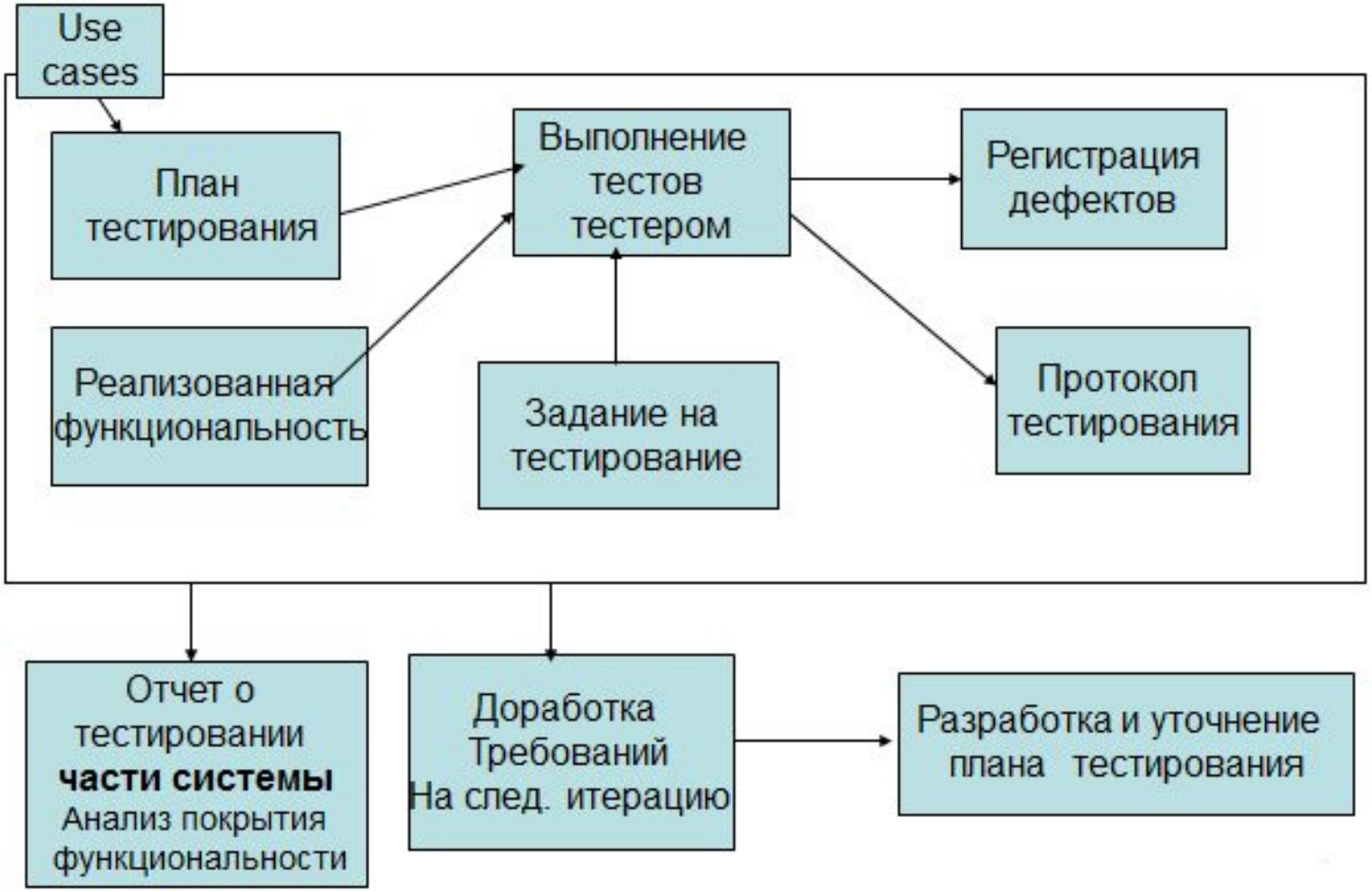


1. выбор архитектуры подсистем
2. выявление информационных потребностей конечных пользователей системы
3. концептуальное проектирование
4. логическое проектирование
5. отладка подсистем
6. сопровождение подсистемы

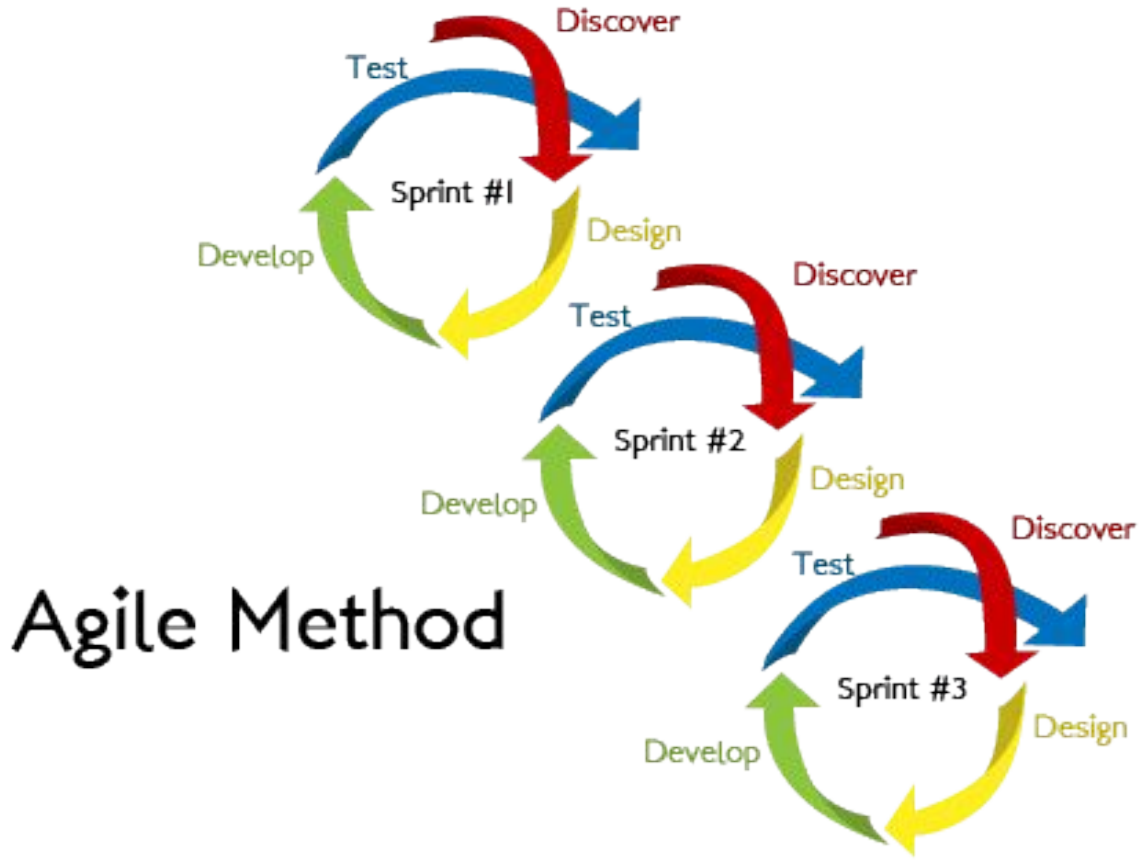
Компания хороших идей Bellintegrator

Модели разработки ПО

Тестирование в спиральной модели



Модели разработки ПО
Методология Agile



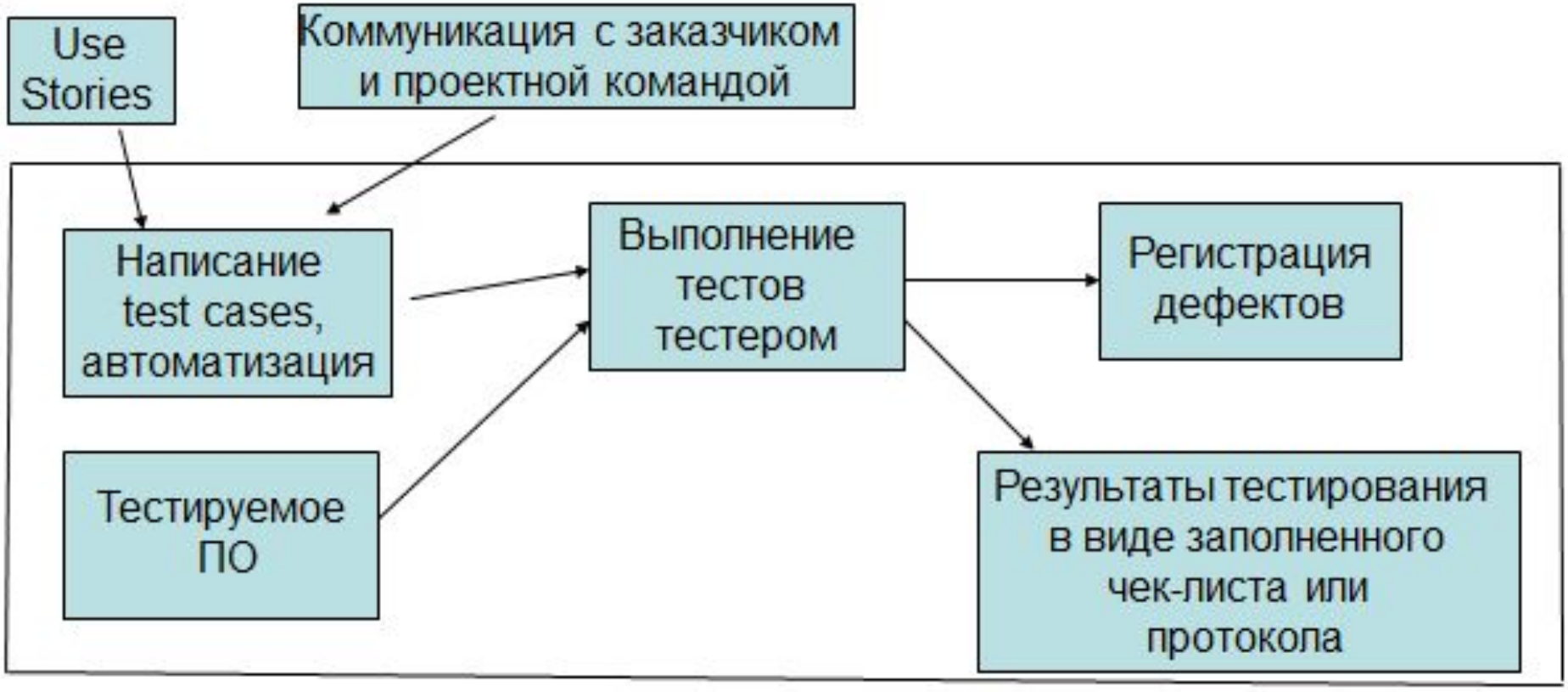
Компания хороших идей
Bellintegrator

Модели разработки ПО
Agile. Используемые подходы

- Вовлеченность заказчика
- Пользовательские истории как база тестирования
- Короткие рабочие циклы
- Test-driven development (unit tests)
- Acceptance tests
- Автоматизация тестирования
- Учет нужд тестирования при проектировании и разработке
- Большой объем регрессионного тестирования
- Приемочные тесты – форма документирования функциональности системы
- Изолированность модулей

Модели разработки ПО

Agile. Тестирование в agile

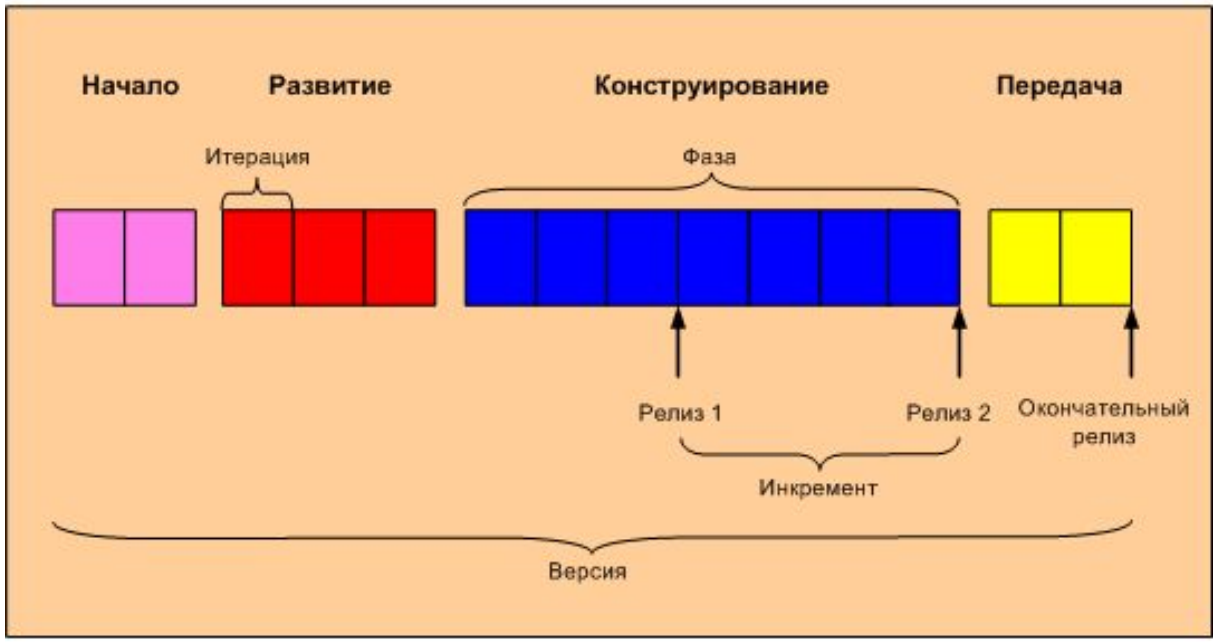


Модели разработки ПО

RUP. Rational Unified Process

Модели разработки ПО

RUP. Фазы разработки RUP



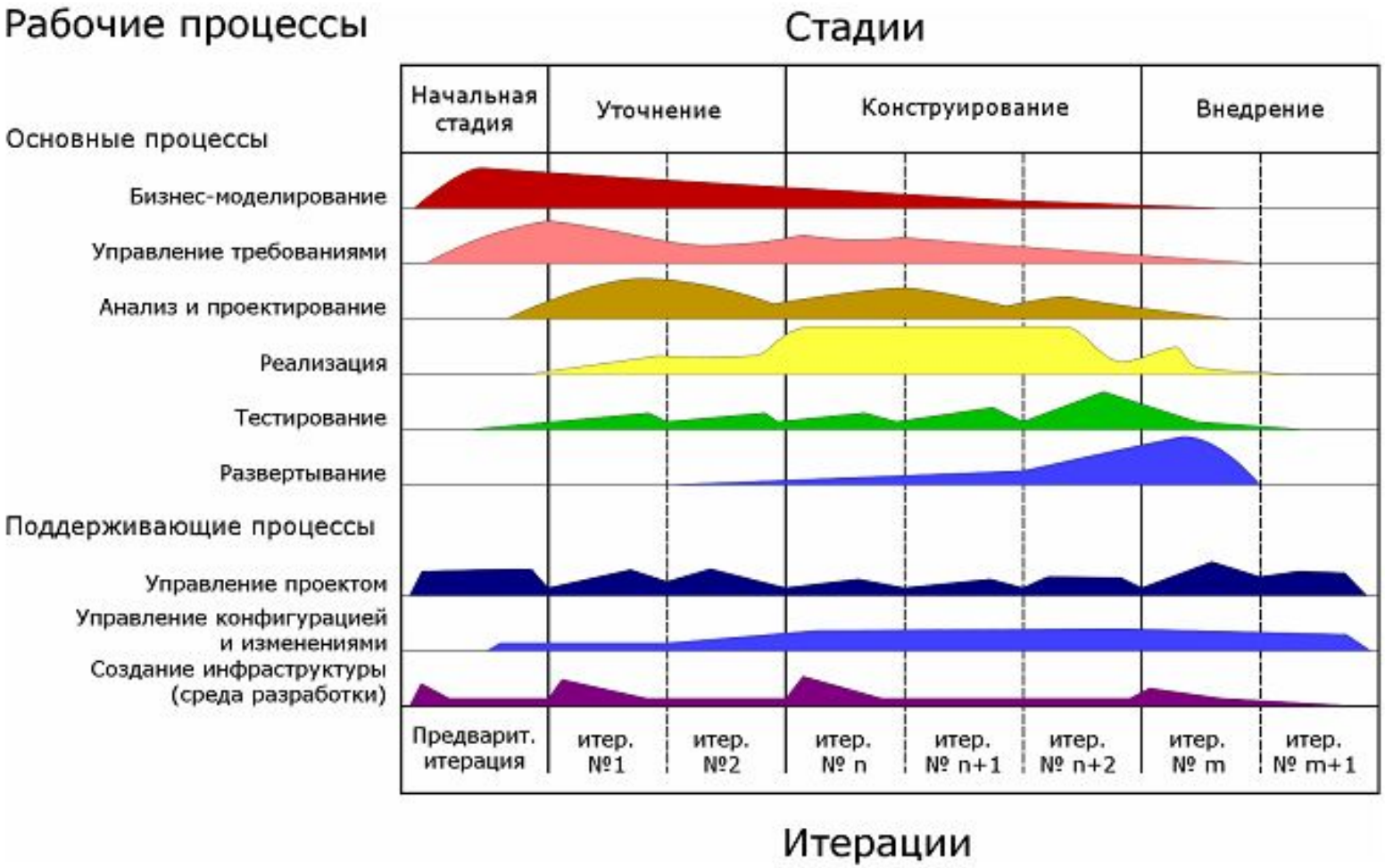
1. Начальная стадия (Inception) **3. Построение (Construction)**

2. Уточнение (Elaboration) **4. Внедрение (Transition)**

Компания хороших идей
Bellintegrator

Модели разработки ПО

RUP. Процессы и стадии



Модели разработки ПО

RUP. Цели тестирования в RUP

- Поиск и документирование дефектов качества;
- Общие рекомендации относительно качества;
- Проверка выполнения основных предположений и требований на конкретных примерах;
- Проверка, что продукт функционирует так, как было запроектировано;
- Проверка, что требования выполнены соответствующим образом.

Компания хороших идей Bellintegrator

Модели разработки ПО

RUP. Роли тестирования в RUP

Роль	**Описание**
Test Manager, Test Project Manager	▪Обеспечивает управление работами ▪Осуществляет оценку трудозатрат, рисков ▪Получает необходимые ресурсы ▪Обеспечивает управленческую отчетность ▪Мониторинг результатов тестирования
Test Designer	Разрабатывает план тестирования, проектирует тестовые данные ▪Разрабатывает модель тестирования, определяет приоритеты ▪Оценивает эффективность сценариев тестирования
Tester, Test Engineer	▪Выполняет тесты, автоматизирует ▪Фиксирует результаты ▪Восстанавливает тесты и систему после сбоев ▪Воспроизводит условия дефектов и сбоев ▪Документирует дефекты и запросы на изменение
Test System Admin, DB Administrator, DB Manager	▪Администрирует систему управления тестированием и дефектами ▪Инсталлирует, администрирует и управляет доступом к тестовым данным

Компания хороших идей Bellintegrator

Модели разработки ПО

RUP. Достоинства RUP в тестировании

- Итерационная разработка повышает вероятность обнаружения ошибок.
- Концентрация на наиболее критических требованиях к ПО.
- Ориентация на архитектурные задачи – устранение архитектурных рисков
- Следование архитектурным шаблонам – повышение качества и скорости реализации системы
- Принцип достаточно хорошего качества
  - поиск и устранение следующей ошибки сейчас обойдутся дороже, чем возможные потери Заказчика при проявлении ошибки и затраты на ее устранение в будущем
- Статистический анализ результатов тестирования
- Большой объем тестирования и улучшение качества системы
- Использование современных технологий –> улучшение качества
- Автоматизация регрессионного тестирования
- Инструментальная поддержка
- Управляемость

Компания хороших идей
Bellintegrator

Цикл тестирования ПО

Компания хороших идей
Bellintegrator

Цикл тестирования ПО

Основные этапы

1. Анализ
2. Разработка стратегии тестирования
и планирование процедур контроля качества
3. Работа с требованиями
4. Создание тестовой документации
5. Тестирование прототипа
6. Основное тестирование
7. Стабилизация
8. Эксплуатация

Компания хороших идей
Bellintegrator

Цикл тестирования ПО

Основные этапы. Анализ

Даже самый качественный код не спасет ваше приложение, если требования к ПО не были должным образом систематизированы или же были плохо документированы.

Цикл тестирования ПО

Основные этапы. Стратегия

Качественно разработанная стратегия позволяет избежать хаотичных или избыточных тестовых процедур, гарантируя при этом своевременный контроль качества всех компонентов системы

Цикл тестирования ПО

Основные этапы. Тестовая документация

Основная цель — сделать объем и ход выполнения задач прозрачным и понятным для заказчика.

Необходимо позаботиться о своевременном создании и регулярном обновлении соответствующей документации, фиксируя все шаги проделанной работы.

Компания хороших идей
Bellintegrator

Цикл тестирования ПО

Основные этапы. Тестирование прототипа

Своевременные изменения, выполненные на концептуальном уровне во время прототипирования, помогают предотвратить дорогостоящие переделки системы на стадиях разработки.

Цикл тестирования ПО

Основные этапы. Тестирование прототипа

Компоненты системы	**Глубина/тип тестирования**	**Область тестирования**
Модульное тестирование Интеграционное тестирование Системное интеграционное тестирование	Приемочное тестирование «Позитивное» тестирование «Негативное» тестирование Исследовательское тестирование Регрессионное тестирование	Функциональное тестирование Тестирование производительности (в т.ч. нагрузочное тестирование, стресс-тесты) Тестирование графического интерфейса пользователя Тестирование удобства пользования Тестирование безопасности Тестирование базы данных Тестирование совместимости (в т. ч. кроссплатформенное)

Компания хороших идей
Bellintegrator

Цикл тестирования ПО

Основные этапы. Стабилизация

Тестирование проводится в условиях, наиболее приближенных к реальным (или даже в условиях эксплуатации).

Цикл тестирования ПО

Основные этапы. Эксплуатация

Даже после ввода системы в эксплуатацию тестирование все еще выполняет важную роль на стадии поддержки.


Компания хороших идей
Bellintegrator