

Software Quality Assurance Days  
10-я Международная конференция  
2-3 декабря 2011, Москва



# Аналитик и Тестировщик в одном лице – путь к качеству

Докладчик:

Максим Цепков (M.Tsepkov@custis.ru)

[www.custis.ru](http://www.custis.ru)

CUSTIS®

# О чем этот доклад?

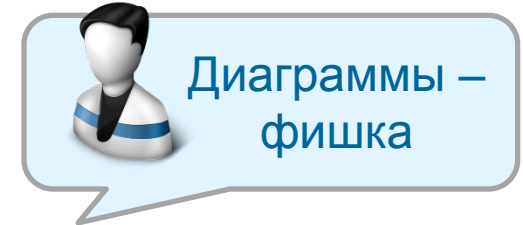
- Процесс разработки и разделение ролей:
  - Классика – водопад, разделение ролей – оттуда.
  - IT-отрасль меняется, меняются и модели.
- Есть альтернатива – модель аналитика-заказчика:
  - В команде – аналитики-тестировщики и разработчики.
  - **Делимся опытом успешного использования.**



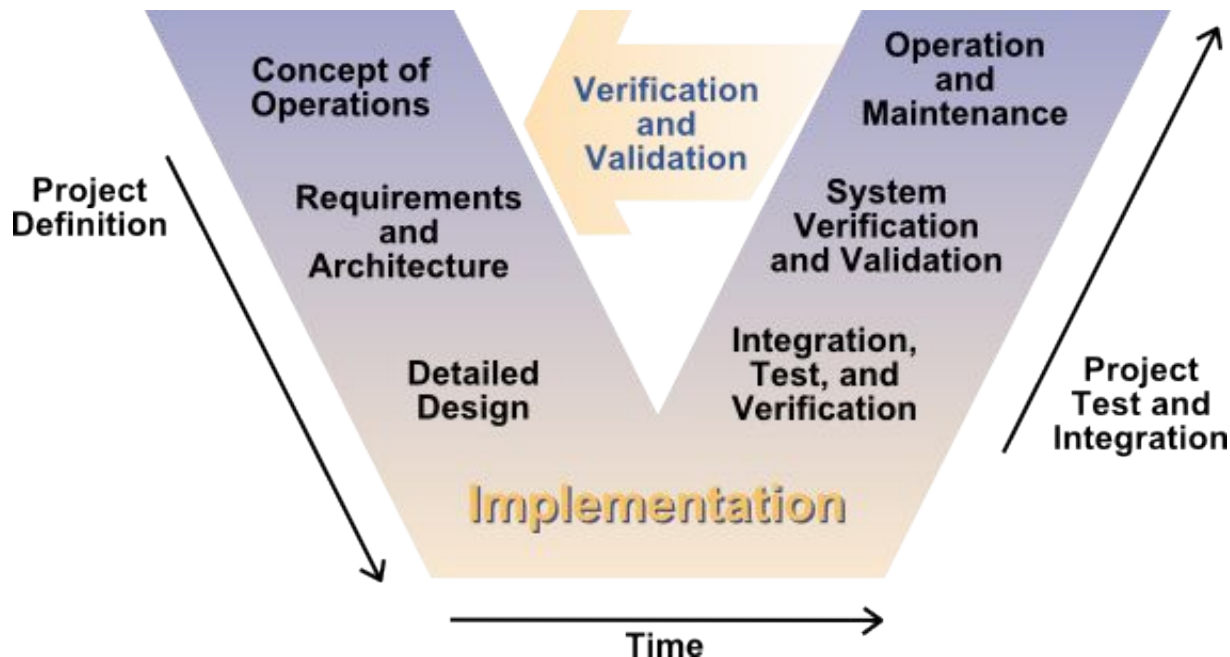
Смотрим на опыт других  
и вырабатываем свой подход.

# Визуальное представление ролей

□ Для эффективного обсуждения нужно графическое представление.



□ Это оказалось удобно делать на схеме V-модели.



# Процесс разработки и роли

# Agile – мировой тренд

Это просто факт

## □ Наблюдаемые признаки:

- Признание и использование Agile в ведущих IT-компаниях и в inhouse-разработке.
- Явное упоминание Agile в базовых документах ([SWEBOK](#), [PMBOK](#)).
- Россия – в русле мирового развития.

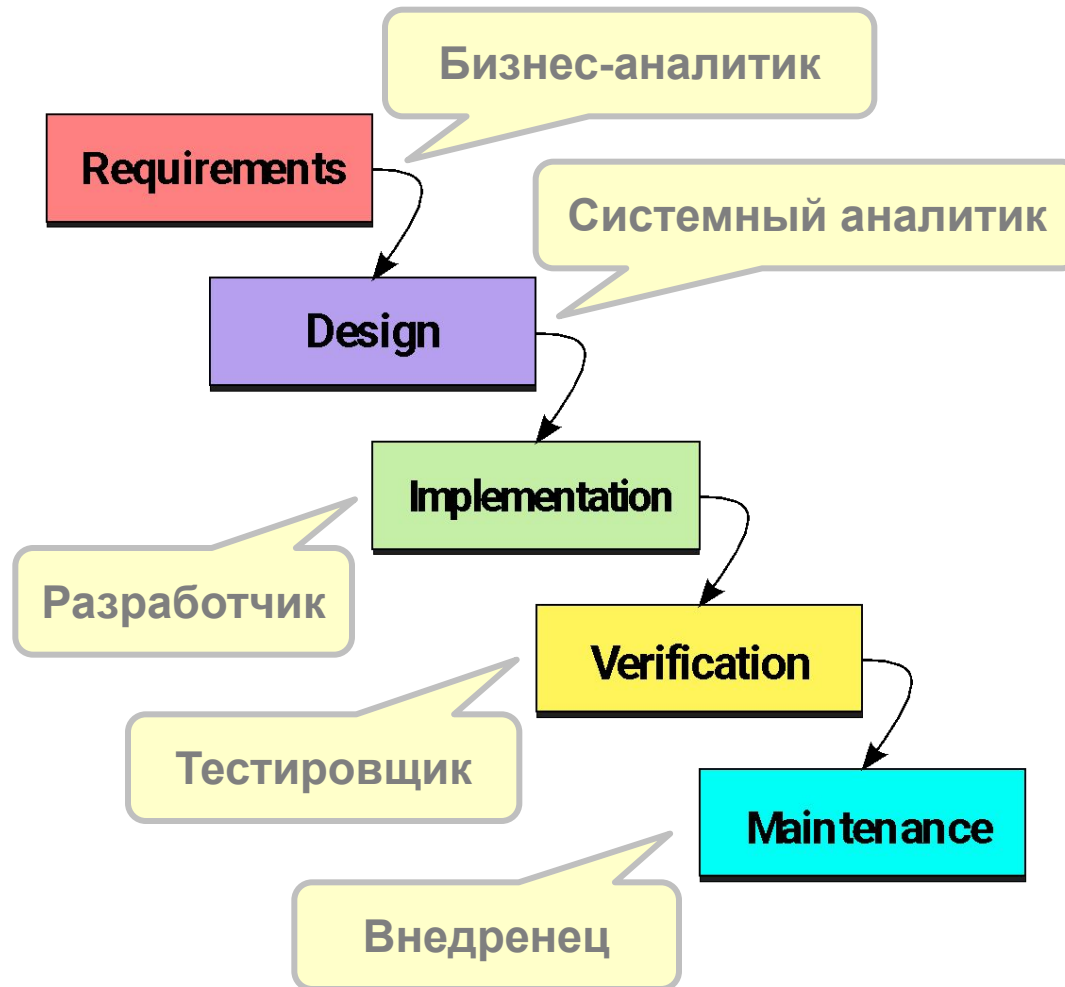
## □ Мечта о едином, эталонном процессе похоронена.

- Даже в варианте «возьмите только нужное» (PMBOK).

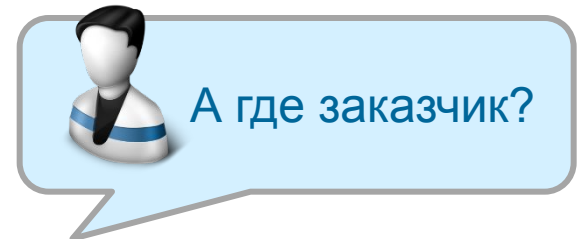
## □ Делаем процесс, адекватный проекту и компании!

- SCRUM/Canban/XP – лишь распространенные комбинации.
- Комбинируем известные успешные практики, придумываем свои.
- Фокус на эффективные коммуникации и автономность команды. <sup>5/21</sup>

# Водопад ушел – роли остались



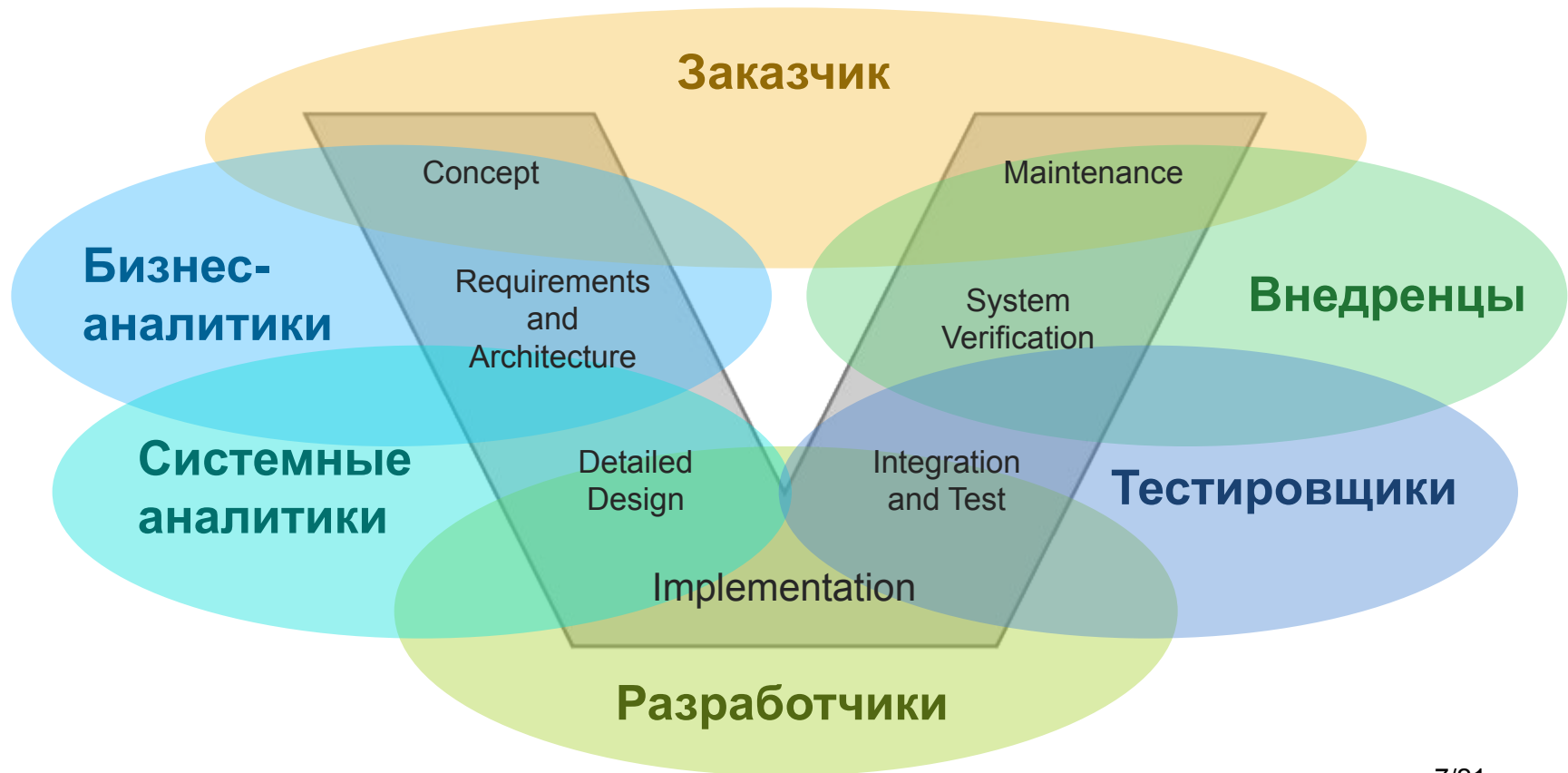
- Каждой стадии – своя роль.
- Роли выполняются разными людьми или командами.
- Передача работы – через артефакты на отдельных языках.




Модель водопада – [http://en.wikipedia.org/wiki/Waterfall\\_model](http://en.wikipedia.org/wiki/Waterfall_model)

# Роли водопада на V-модели

- ☹ Коммуникации – лишь с соседями.
- ☹ Длинный цикл общения ведет к потере информации.

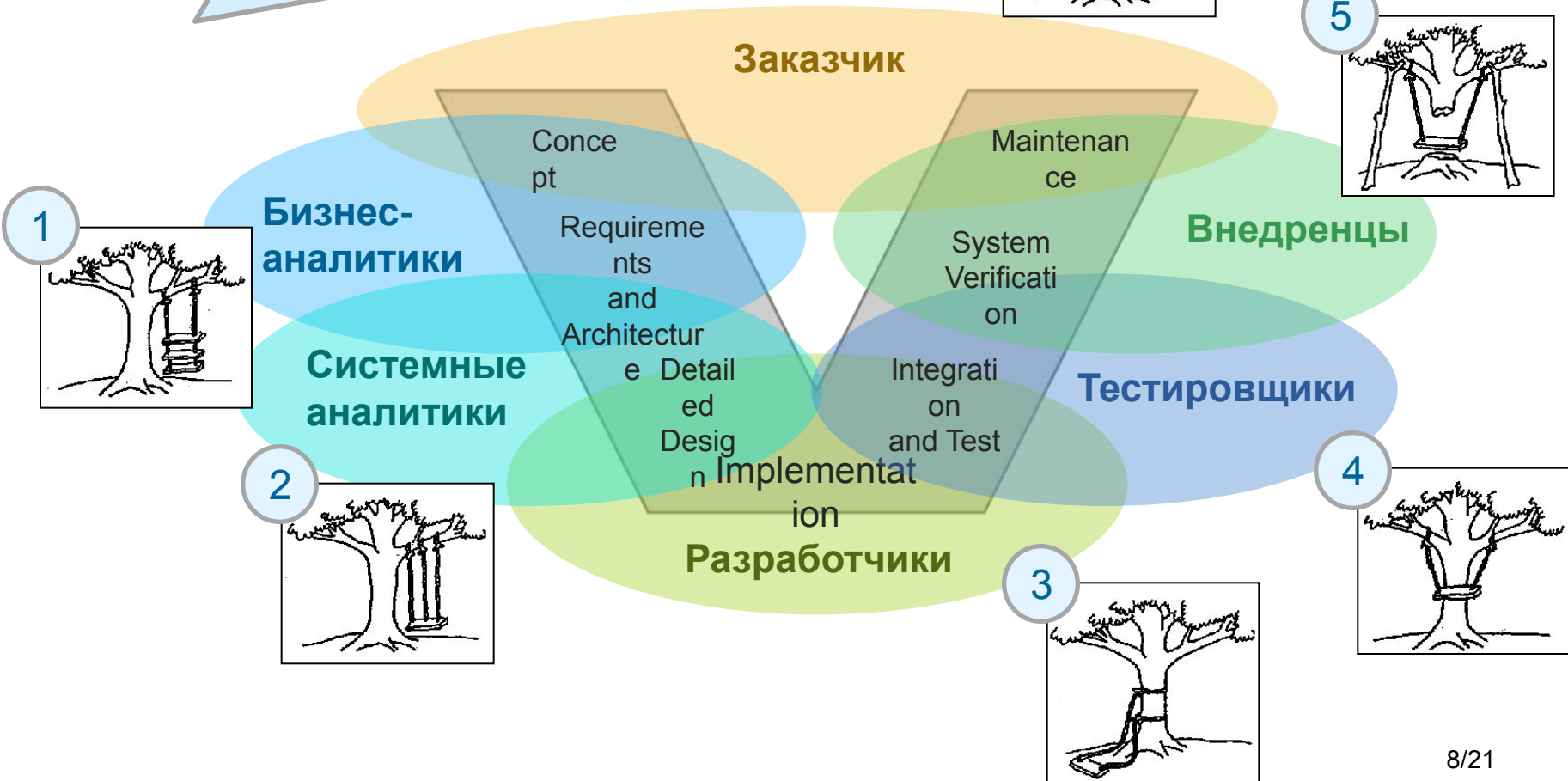
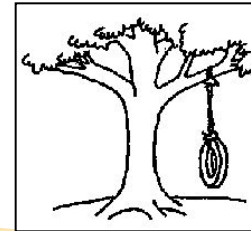


# Изменение видения проекта...



Старый известный образ...

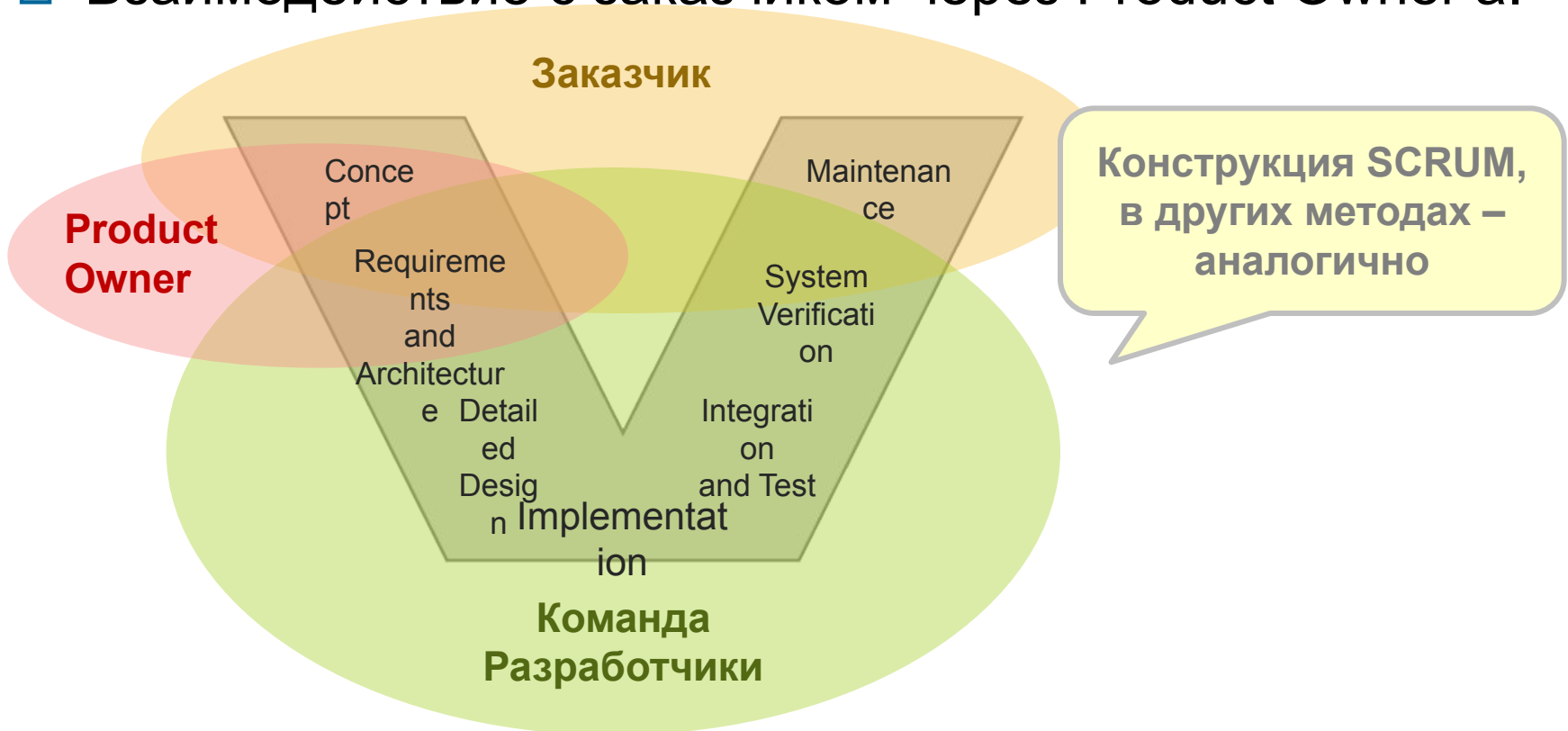
Что хотели





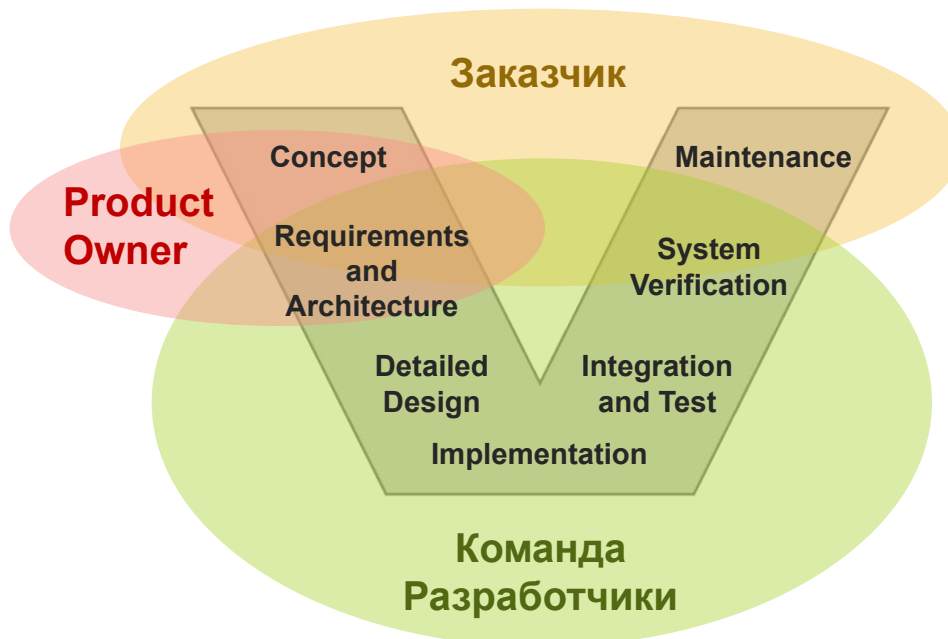
# Что предлагает Agile?

- Кросс-функциональная команда разработчиков.
- Взаимодействие с заказчиком через Product Owner'a.



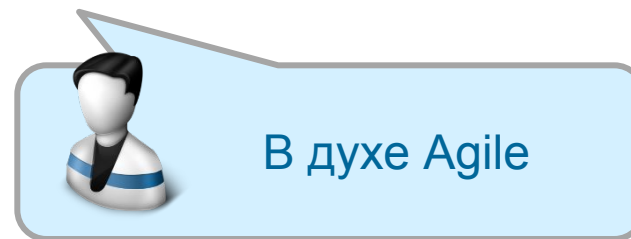
# Плюсы и минусы

- 😊 Эффективные коммуникации.
- 😊 Возможность быстрой обратной связи.
- 😞 Большая нагрузка на Product Owner'а.
- 😞 Расширение зоны ответственности Заказчика.
- 😞 Слишком разнообразная работа членов команды.



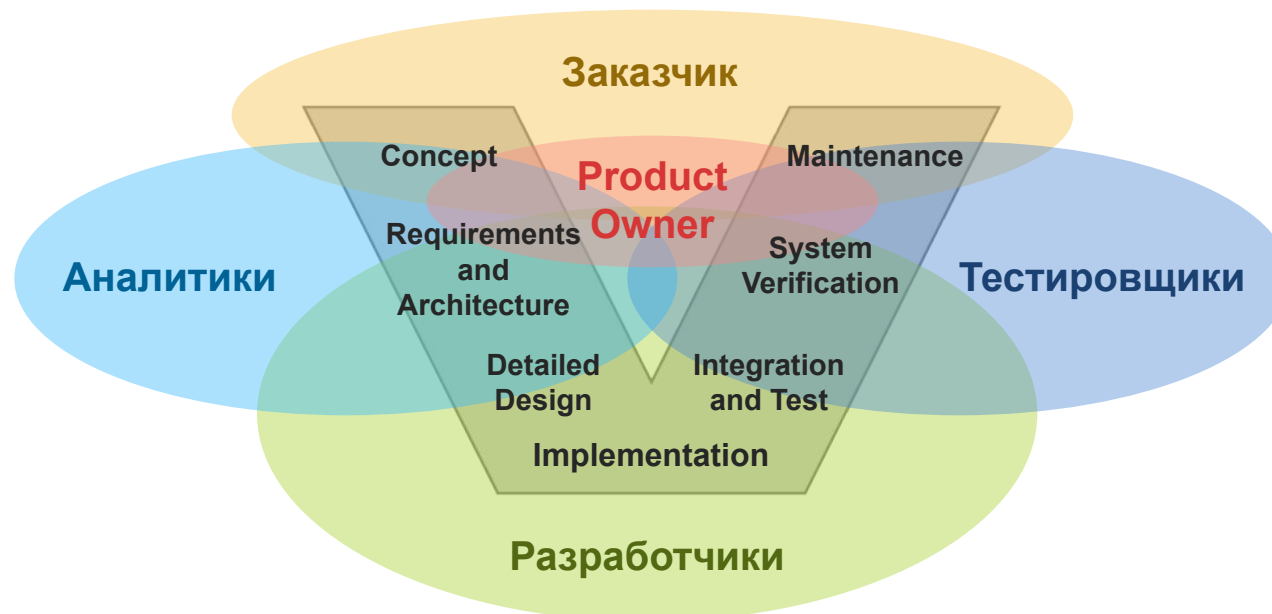
 Подходит далеко не для всех проектов.

# Ищем хорошие варианты



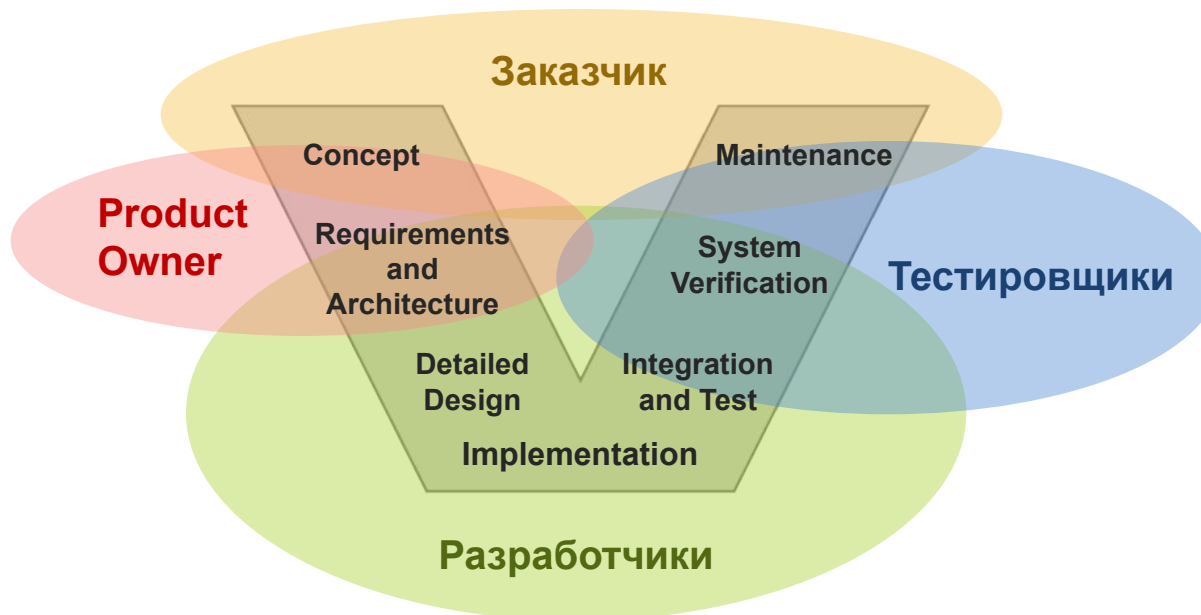
# Специализация внутри команды

- Кросс-функциональная команда не означает полной взаимозаменяемости, возможна специализация.
- 😊 Снижается нагрузка на Product Owner'а.
- 😞 Большое число ролей затрудняет коммуникации.
- 😞 Неравномерная нагрузка на роли в ходе проекта.



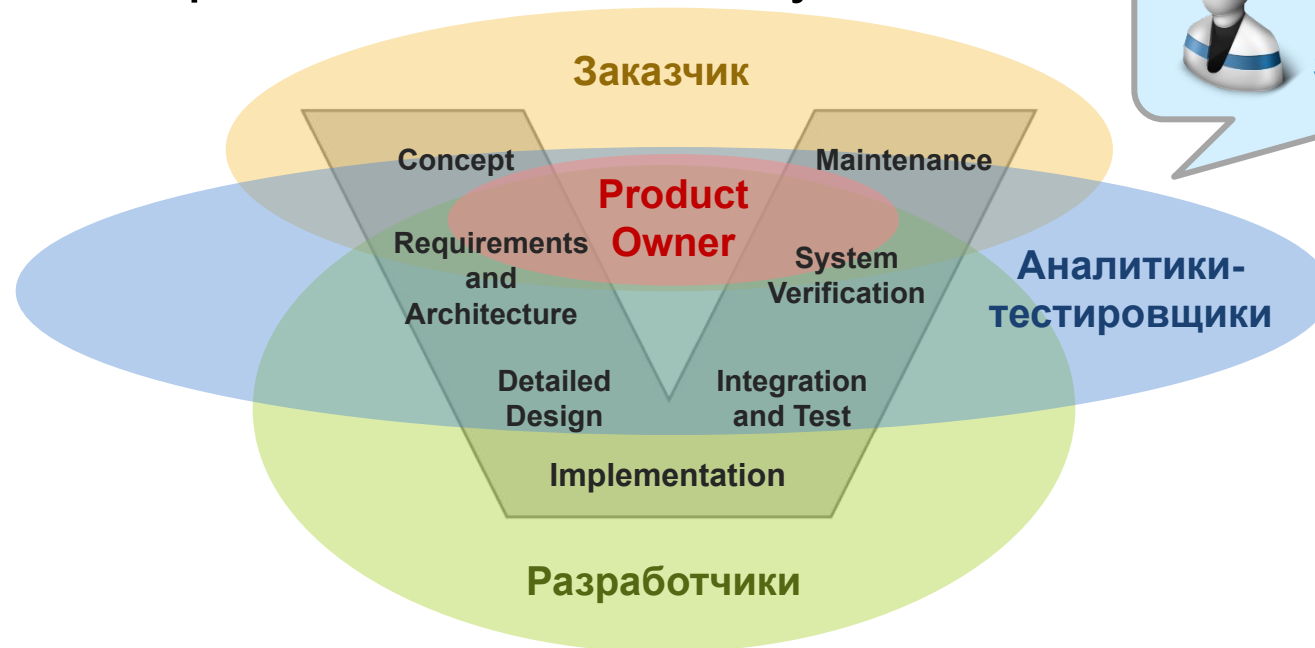
# Есть проекты, где аналитики мало

- Команда разработчиков и тестировщиков – распространенный вариант.
- 😊 Две роли – не много, но достаточно.
- ☹ Не подходит, когда аналитической работы много.



# Модель внутреннего заказчика

- Аналитики получают требования заказчика и формулируют задачу разработчикам.
- Они же принимают результат разработки и передают его заказчику.



Новое – хорошо забытое старое.

# Область применения модели

- Для проектов с полным набором активностей.
- CUSTIS – заказная разработка: обследование, постановка, разработка, внедрение, **развитие**.
- Для продуктовой разработки тоже применима.
- Модель распространена в мире



[Вернер на Req-Labs.](#)

# Преимущества модели

- ☺ Автономность команды разработки.
- ☺ Эффективные коммуникации внутри и с заказчиком.
- ☺ Быстрая реакция на требования заказчика (скорость поставки часто важнее качества продукта).
- ☺ Прием результата разработки аналитиком повышает соответствие системы ожиданиям заказчика.
- ☺ Две роли в команде – возможность дублирования.
- ☺ Равномерная нагрузка на роли в ходе проекта.



Все вместе дает высокое качество услуги для заказчика.



# Почему аналитика, тестирование, внедрение – схожая активность?

## □ Представить работу пользователя с системой:

- Бизнес-сценарии – основа требований и тестов.
- Основные активности пользователей, эргономика.
- Сложные случаи – для проектирования и проверки.

Это все – общие активности.

## □ Общение с Заказчиками и Пользователями:

- Выяснение их работы и ее особенностей.
- Уточнение при альтернативных и спорных моментах.
- Объяснение работы системы.
- Консультации по сложным случаям.



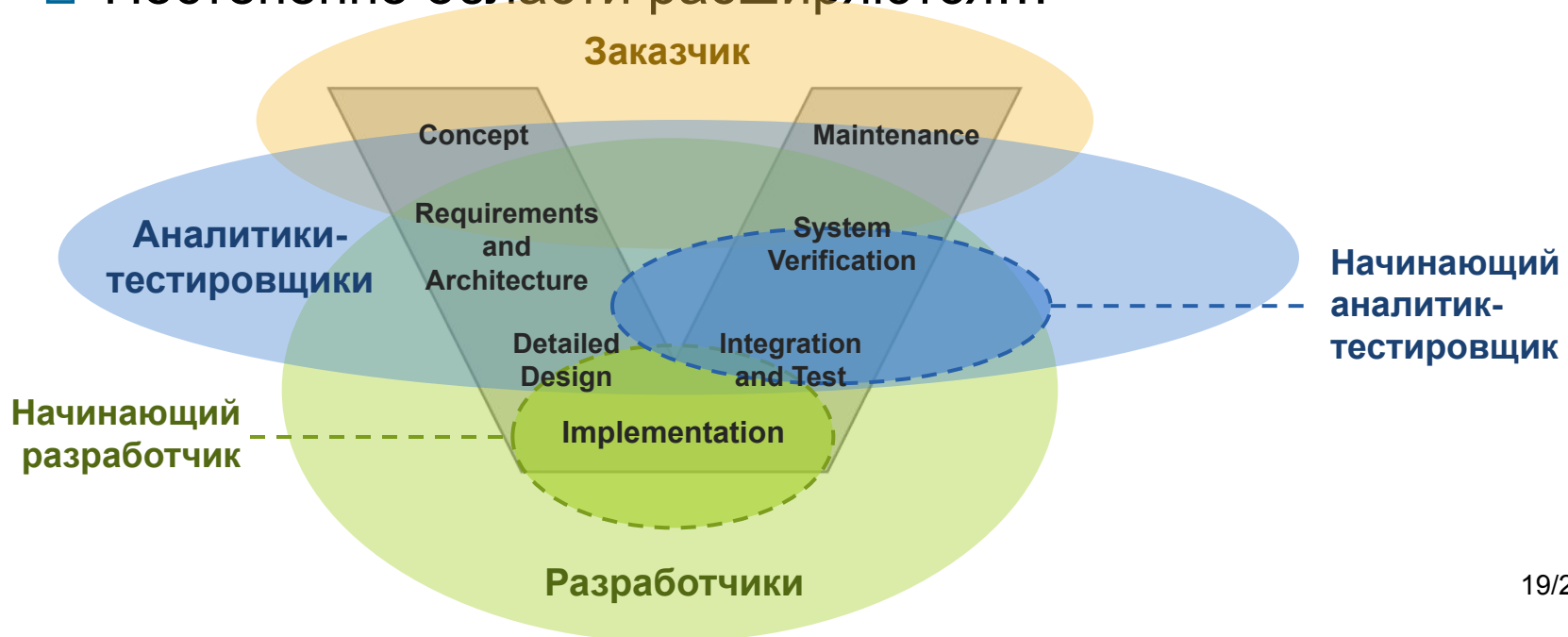
В аналитике и тестировании есть место и сенсорикам, и интуитам.

# Опыт CUSTIS – типовая команда\*

- Соотношение разработчиков и аналитиков – 2:1.
- 6–7 (4–11) человек: 4 разработчика, 2 аналитика и руководитель проекта (Product Owner).
- Члены команды могут заменять друг друга с учетом специализации. У руководителя тоже есть зам.
- Применение DDD дает единый язык общения.
- Часть разработчиков и аналитиков – начинающие, они растут и набирают опыт.
- По мере роста опытные сотрудники уходят в новые проекты, а новички – приходят.

# Рост новичков в команде

- Активность аналитика начинается с тестирования: освоение системы и бизнес-области.
- Активность разработчика начинается с реализации по проработанным постановкам.
- Постепенно области расширяются...



# Подводя итоги

## Общее:

- Создавайте разделение ролей исходя из проекта.
- Для визуализации хорошо использовать V-модель.
- Эффективные коммуникации – необходимы.

## Частное:

- Аналитик, тестировщик и специалист по внедрению и сопровождению в одном лице – эффективно.
- Скорость поставки доработки часто важнее, чем ее качество.



Спасибо!

Вопросы?

Максим Цепков

[M.Tsepkov@custis.ru](mailto:M.Tsepkov@custis.ru)

CUSTIS®