

Одиннадцатая независимая научно-практическая конференция «Разработка ПО 2015»

22 - 24 октября, Москва



# Паттерны организации эффективной разработки ПО

Безуглый Дмитрий  
ООО «Системный подход»

## Безуглый Дмитрий

- **Управление продуктами, проектами, командами, процессами**
- НТТУ «КПИ» 1998 год. ВМКС . Инженер - Системотехник
- Около 20-лет опыта участия в проектах по созданию и развитию различных систем и продуктов.
  - Max масштаб проекта 50 инженеров, около 30 чел-лет. Перевод на Scrum-of-Scrum
  - Max бюджет проекта 2,5 млн долл.
  - Max ROI проекта 400%
- Основатель компании «Системный Подход» с 2008 года
  - Тренер/Консультант
  - Более 900 участников тренингов
  - Десятки реализованных проектов
- Экспертная фасилитация стратегических сессий





"Мир(организация) такой, какой он есть,  
потому что он таким стал" ...

# Жизненный цикл корпоративных приложений



# В поисках «Серебряной пули»

- За что вы так не любите простые решения ?
- У них бывают очень сложные последствия ...
  - С. Лукьяненко

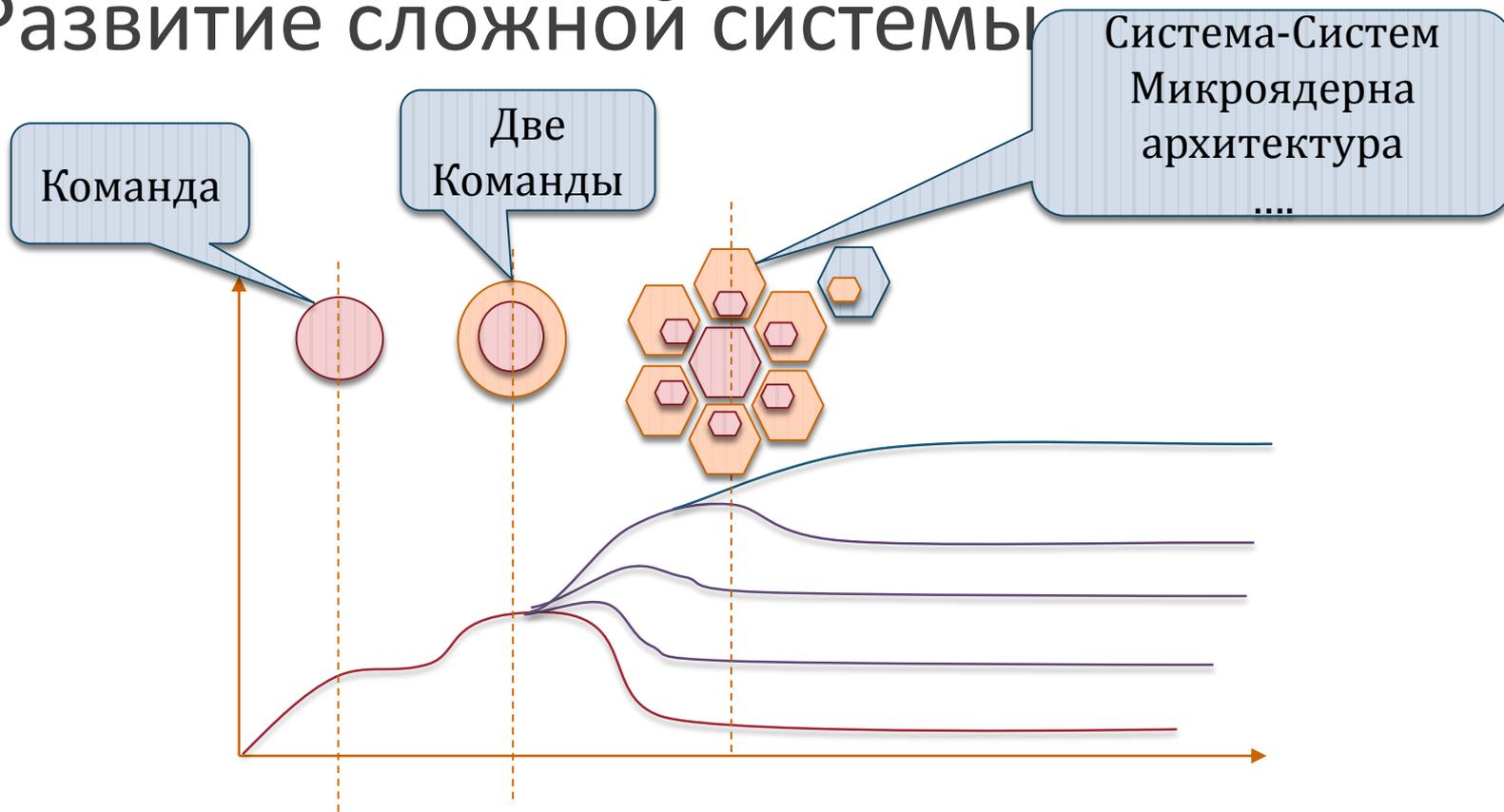


# Закон Конвея

- «Структура созданной системы отражает структуру связей в команде/коллективе задействованной в ее создании»
- Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations

— Melvyn Conway, 1967

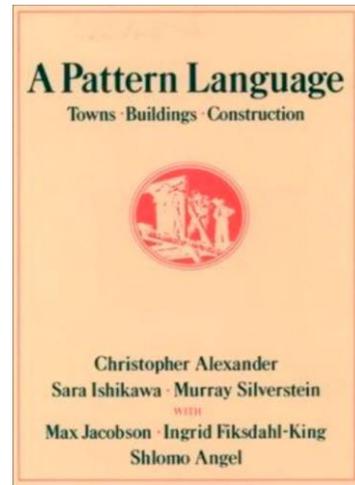
# Развитие сложной системы



# Паттерн

- Каждый шаблон описывает проблему, которая возникает снова и снова в нашей среде, а затем описывает ядро решения этой проблемы, таким образом, что вы можете использовать это решение миллион раз, даже никогда не реализуя одним и тем же образом дважды.
- "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice"

• Alexander K. A  
Pattern Language. Oxford, 1977



# Описание паттерна

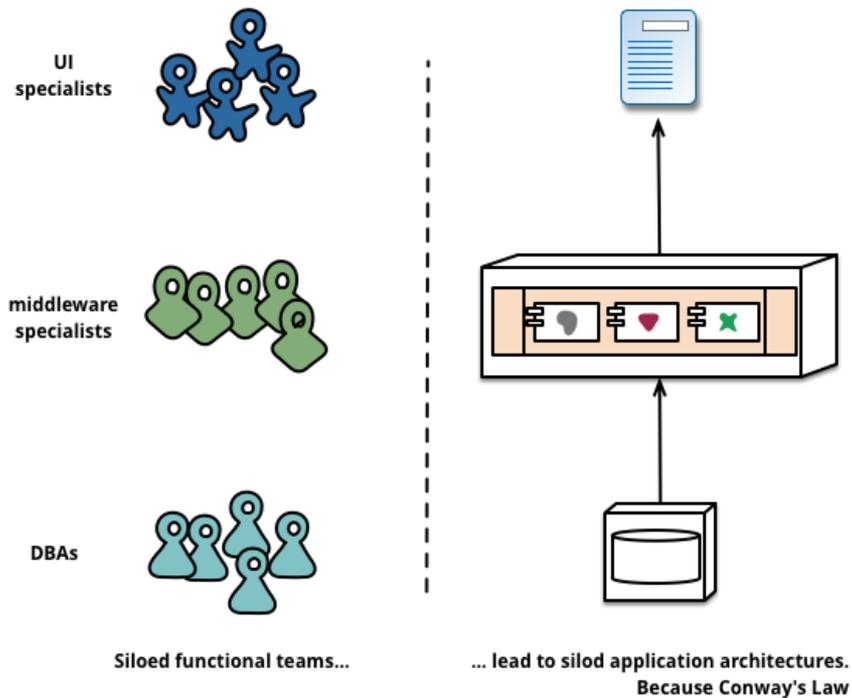
- Контекст
- Проблема
- Силы
- Решение
- Взаимодействие с другими шаблонами
- Эффект

# Сервисы, с разделением на слои абстракции (Abstract Layer)

- Характерно
  - Каждый уровень предоставляет свой сервис
  - Каждый уровень скрывает нижележащие сервисы
  - Каждый уровень имеет хорошо определенные интерфейсы
  - Данные синхронизируются с помощью вызовов, идущих между слоями
- Когда применять
  - В сложных гетерогенных системах
- Пример
  - Модель OSI/ISO
  - Стандартная трехуровневая архитектура



# Трёх-звенная архитектура



“

Процесс – алгоритм позволяющий сделать  
успех повторяемым

*Гай Кавасаки*

# Нейрологические уровни

## Дилтс



## Организация

- Миссия
- Бизнес модель
- Культура +/- Стратегия
- Компетенции
- Процессы
- ...

# Закон Пирамид

- Изменение, происходящее на более низком Логическом уровне, почти никогда не приводит к изменениям на более высоком Логическом уровне; а изменение, происходящее на более высоком уровне, почти всегда приводит к изменениям на более низких уровнях



# Классификация

## По предназначению:

- Управление целью ( Зачем ?)
- Управление Содержанием ( Что делаем ?)
  - В том числе управление качеством (валидация и тестирование)
- Управление процессом ( Как делаем ? )
- Управление временем ( Когда ?)
- Управление людьми ( Кто делает ?)
  - В том числе компетенция, мотивация и текучка
- Управление знаниями
  - В том числе вопросы масштабирования команд

## Контекст применения:

- Персональные
- Проектной команды
- Программы или целого отдела (Центра компетенции)
- Организация

# Примеры паттернов

---

# Команда

- Общая ответственность за результат
- **Самоорганизующаяся**
- Магическое число 7



# Паттерн «Команда»

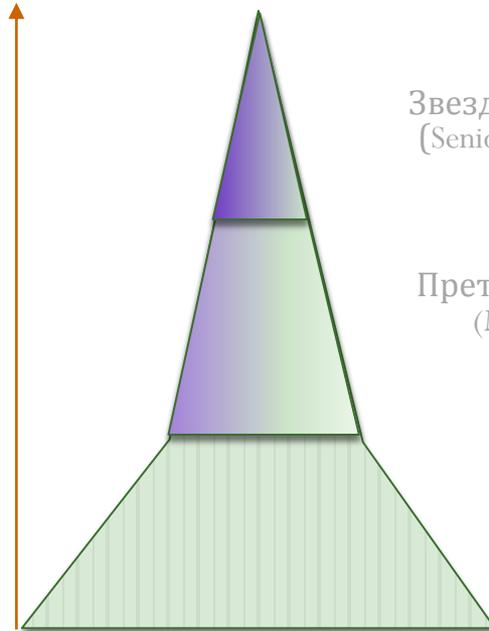
- Контекст
  - Организация работы группы
- Проблема
  - Достижение уникального результата для компании
- Силы
  - Стандартный процесс/культура компании не подходит для реализации задачи
  - Цикл формализации и внедрения процессов больше чем необходимая скорость изменений
  - Люди обладают поддержкой коллективной работы в небольших группах на инстинктивном уровне

# Паттерн «Команда»

- Решение
  - Самоуправление. Обеспечение условий для самостоятельного выбора процесса. Отказ от его формализации
- Взаимодействие с другими шаблонами
  - Может быть составной частью коллектива, Система Систем
- Эффект
  - «+» Возможно создание уникального процесса достижения необходимого результата
  - «+» Эффективность групповой работы
  - «-» Команда устойчива но **ОГРАНИЧЕННА** во времени
  - **Не масштабируется**

# Коллектив. Модель здоровой конкуренции

Компетенция  
и компенсация



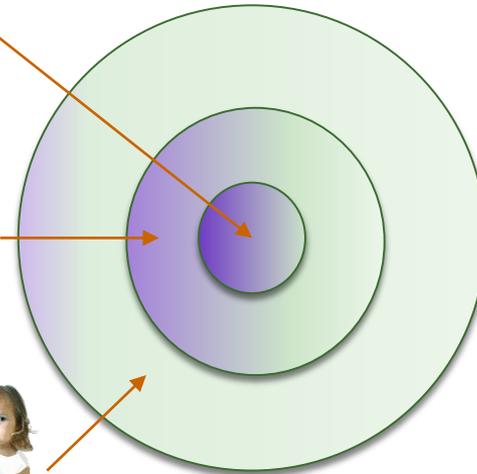
Звезды  
(Senior)



Претенденты  
(Middle)



Новички  
(Junior)



Э. Салливан Время - деньги. Создание команды разработчиков программного обеспечения.

10/23/2015

# Паттерн «Коллектив»

- Контекст
  - Организация работы направления ( От 25 до 500 людей)
- Проблема
  - Обеспечение наивысшей индивидуальной/организационной компетентности
- Силы
  - Команды не устойчивы во времени
  - Конкуренция разрушает команды
  - Команды не масштабируются

# Паттерн «Коллектив»

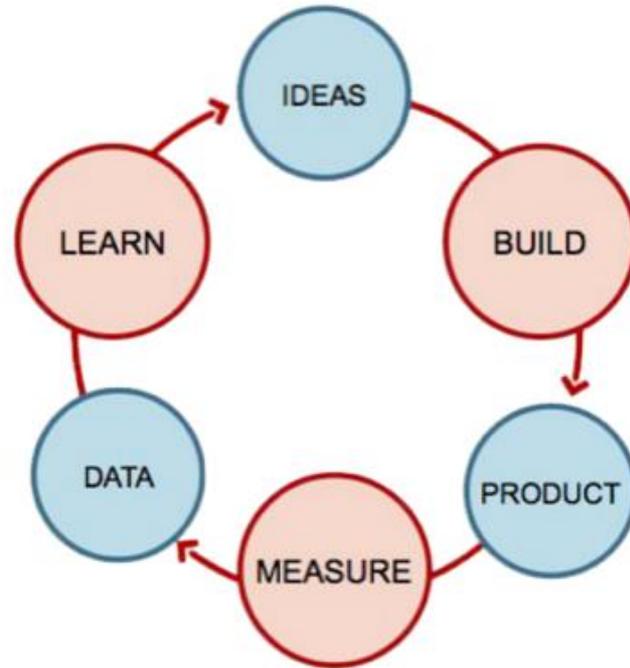
- Решение
  - Создание условий здоровой профессиональной конкуренции
- Взаимодействие с другими шаблонами
  - Может включать команды, «Иерархия целей»
- Эффект
  - «+» Устойчивость во времени
  - «+» Достижение высокого уровня экспертизы
  - «+» Переносимость и масштабируемость опыта
  - «-» Повышенные требования к компетентности руководителей

# Паттерны но не так.

---

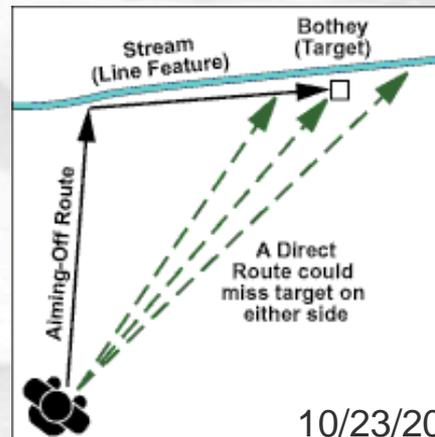
design pattern misuse

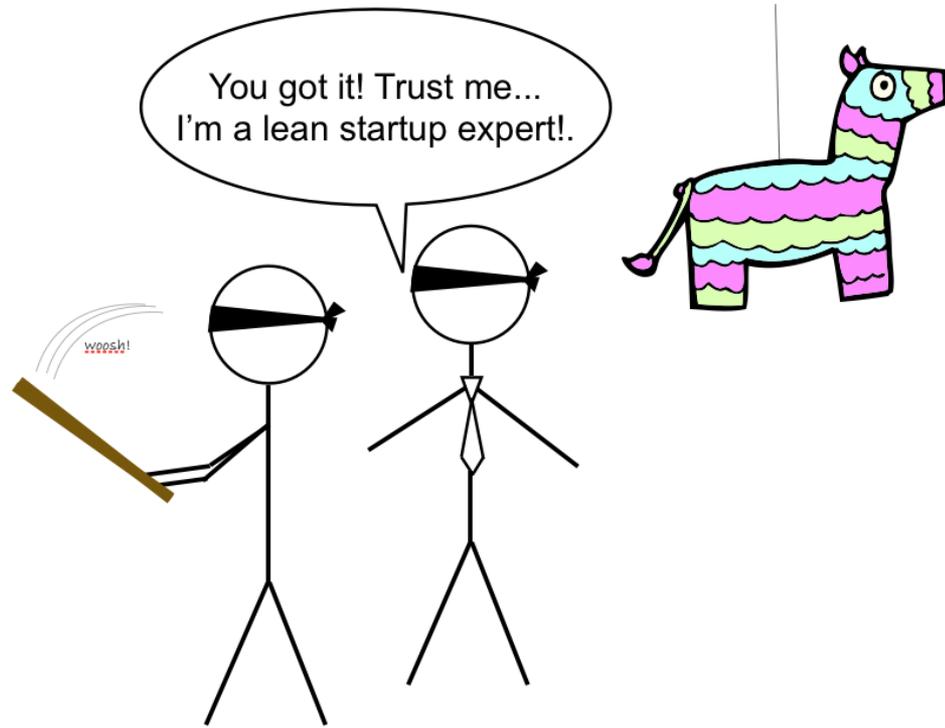
# Lean cycle



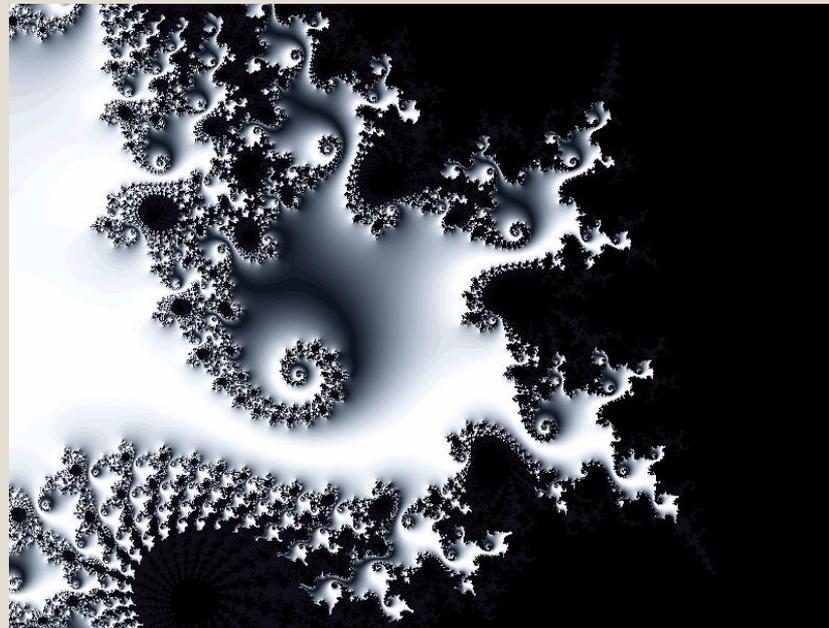
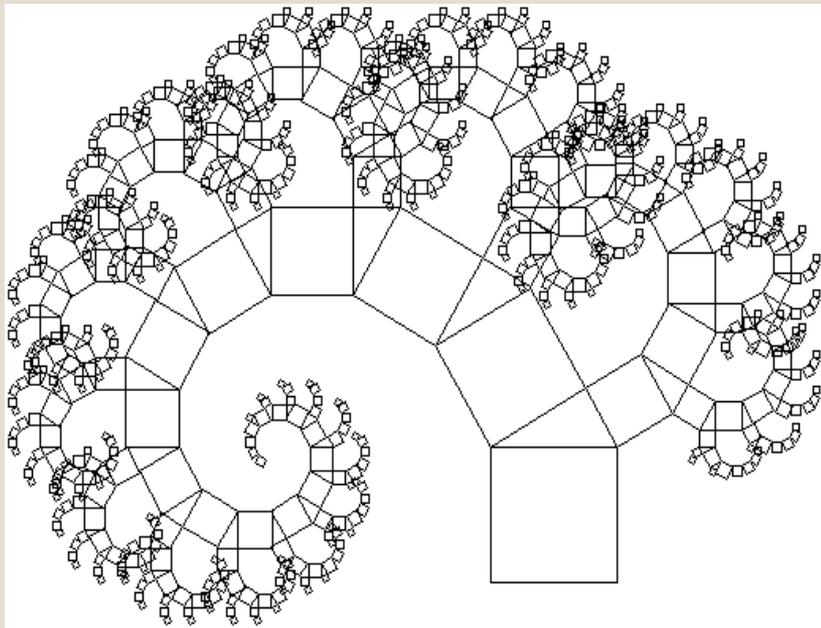
“

Lean & Agile это методология валидации гипотез, А не стратегия достижения цели





Если у Вас не получилось то Вы не TRUE ...  
( Не достаточно веры)

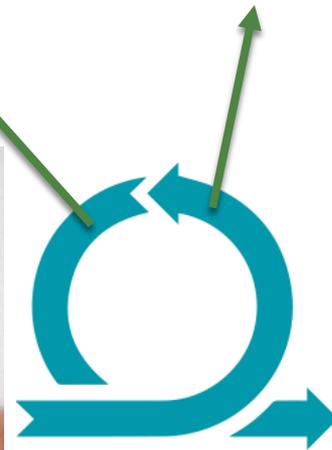
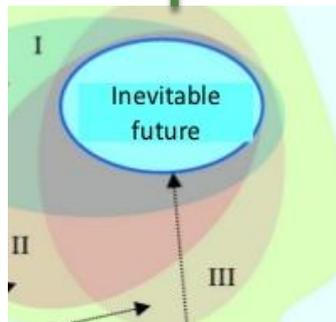


Фрактал как один из примеров паттернов

# Примеры комбинаций паттернов

---

Framework



“

*Универсальной организационной  
структуры/процесса  
ориентированной на создание  
продукта/системы существовать  
не может*

# Краткие выводы

- Паттерны передают опыт успешного решения задач
- Честные паттерны передают **Недостатки**, также как и преимущества
- **Применение паттернов никогда не может быть осуществлено механически**
- Паттерны позволяют осуществлять осознанное и эффективное проектирование через обсуждение альтернатив

# Каталоги паттернов

## Проектирование ПО

- **Patterns of Enterprise Application Architecture**
- Core J2EE Patterns
- Enterprise Integration Patterns
- Microsoft Enterprise Solution Patterns
- Microsoft Data Patterns
- Microsoft Integration Patterns
- Domain Driven Design
- **Analysis Patterns**
- Data Model Patterns
- **Gang of Four**
- POSA

## Создание Организаций

- Кто в банду ?

# Спасибо за внимание !

**Дмитрий Безуглый**

- <https://www.facebook.com/dmitry.bezuglyy>
- [bdl@system-approach.ru](mailto:bdl@system-approach.ru)

**ООО «Системный Подход»**

- <https://www.facebook.com/SystemApproach>
- [www.system-approach.ru](http://www.system-approach.ru)