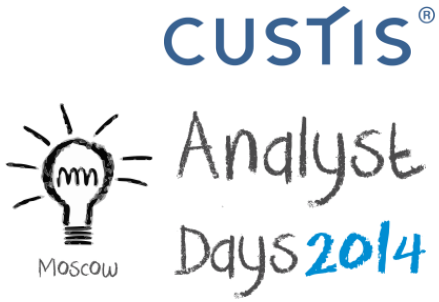


Domain-Driven Design: Модель вместо требований

Максим Цепков

Главный архитектор дирекции развития решений

Москва, 24 мая 2014 года

CUSTIS®

О чем этот доклад?

- Есть разные способы работы с требованиями
- Выбор конкретного зависит от проекта

- В сложных проектах уместна работа с моделями
- И DDD – наиболее эффективный способ для этого

DDD – ключ к построению сложных систем и их развитию вслед за потребностями бизнеса

CUSTIS®

Что будет в докладе?

- Теория
  - Кто и что проектирует – области и роли
  - DDD – единый язык проекта и одна модель системы
    - Модели были давно, но две: бизнес-области и системы
    - Единый язык проекта создает общее поле понятий
    - И позволяет работать с одной общей моделью
    - Но в большом проекте следует выделять контексты
- Практика
  - Единый язык в конкретных примерах
  - DDD в корпоративных приложениях
- Заключение

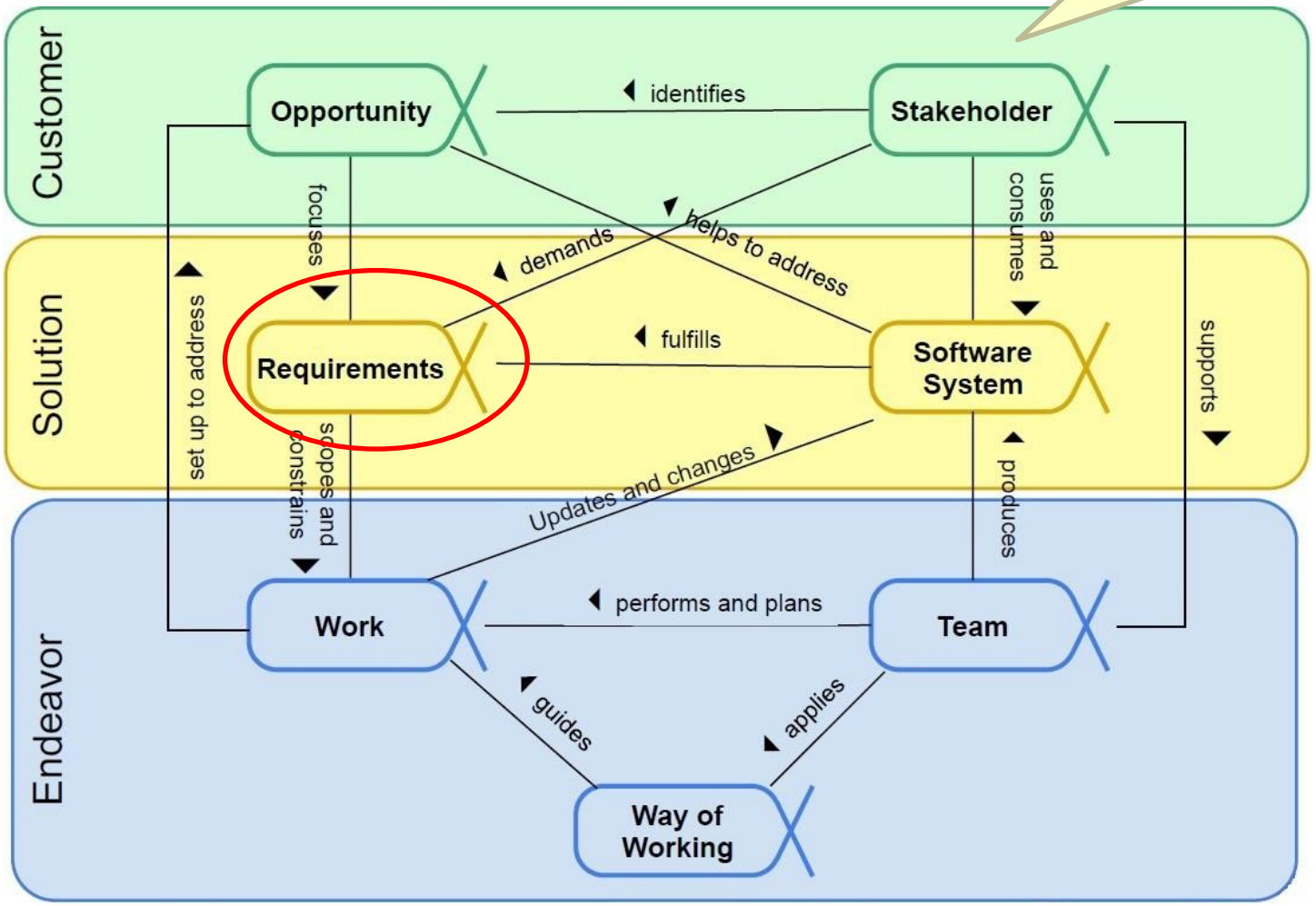
CUSTIS®

Начнем с теории

CUSTIS®

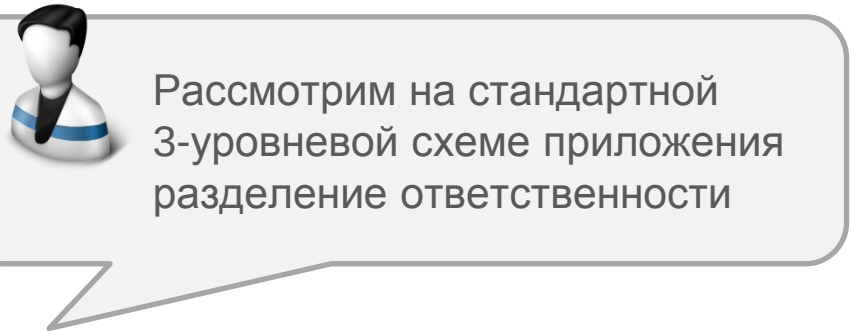
А что такое требования?

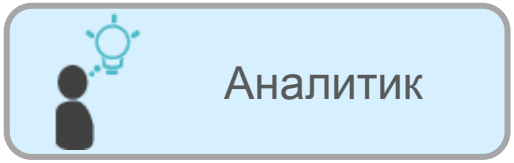
Essence (OMG, SEMAT)
Kernel Alphas



CUSTIS®

Что мы проектируем?



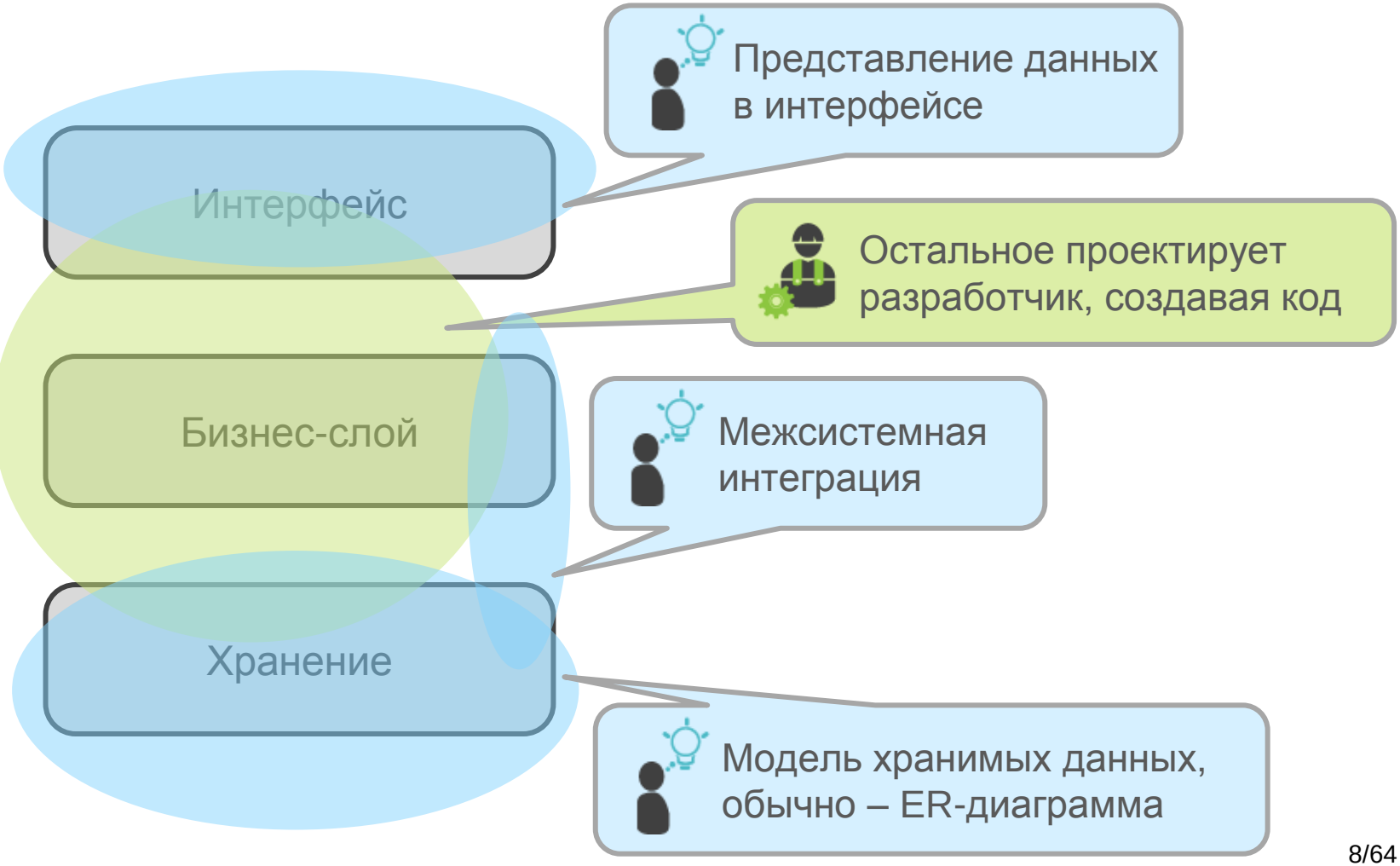


CUSTIS®

Проектирование от внешних требований

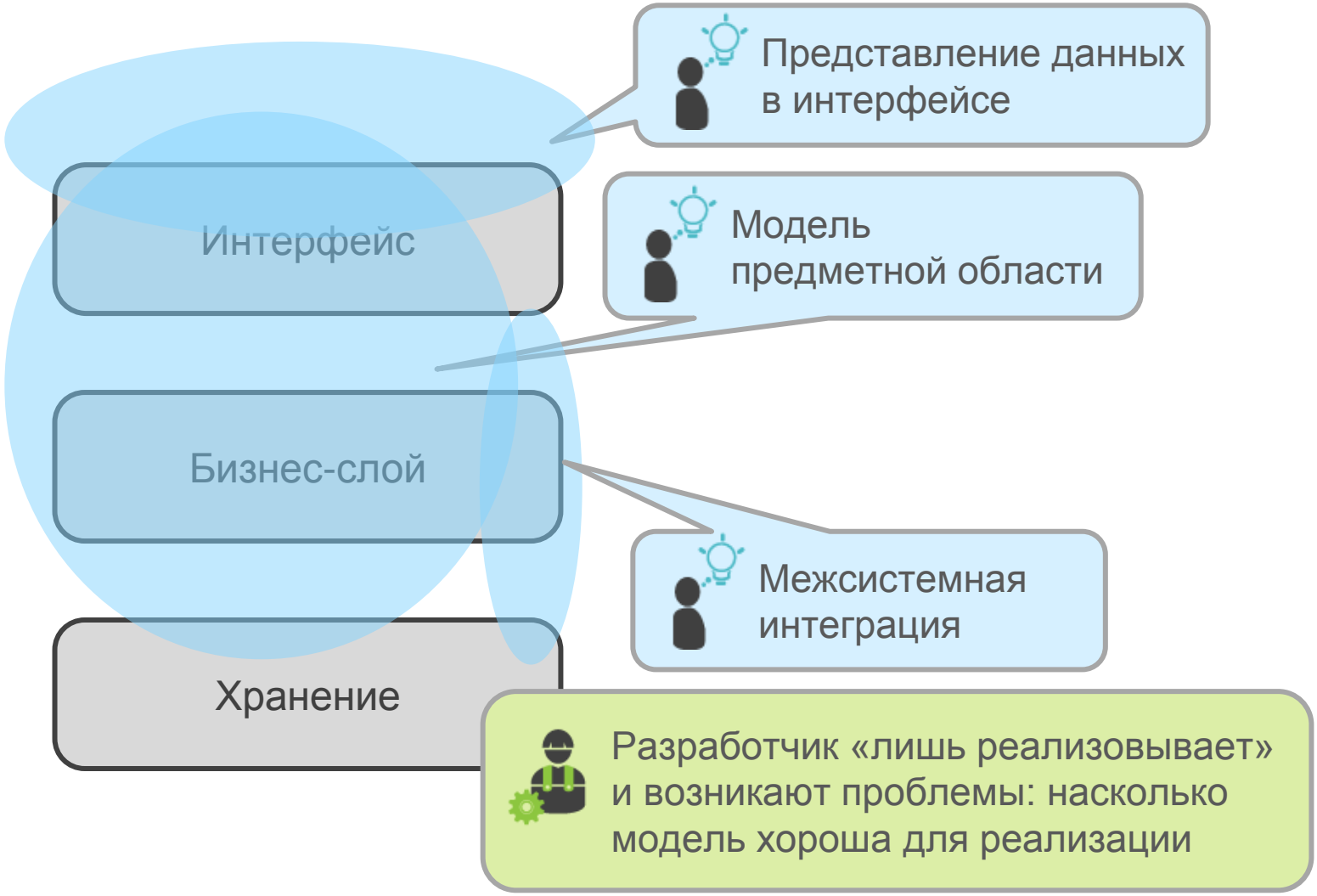
CUSTIS®

Проектирование от данных



CUSTIS®

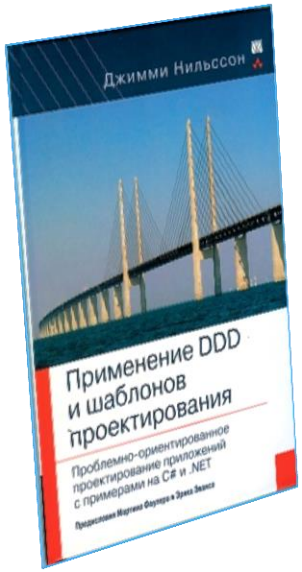
Проектирование по модели



CUSTIS®

DDD: как оно начиналось

- Концептуальная книга Эрика Эванса
  - на английском – в 2003 г.
  - на русском – только в 2010 г.



- Практическая книга Джимми Нильссона
  - на английском – в 2006 г.
  - на русском – в 2007 г. (почти сразу!)

CUSTIS®

Основная идея DDD

CUSTIS®

Единый язык (ubiquitous language)

- Построен на основе терминов предметной области
- Его понимают ИТ-специалисты и эксперты бизнеса
- На нем описана модель ИТ-системы и ее место в бизнес-процессах

Понятия единого языка:
клиент, накладная, платеж, долг –
из предметной области

ООП –
это парадигма
моделирования

CUSTIS®

А где здесь ООП?

- Парадигма моделирования определяет
  - Элементы языка
  - Способ их соединения в сложные конструкции
  - Визуальный образ для представления
  - Способ отражения модели в код

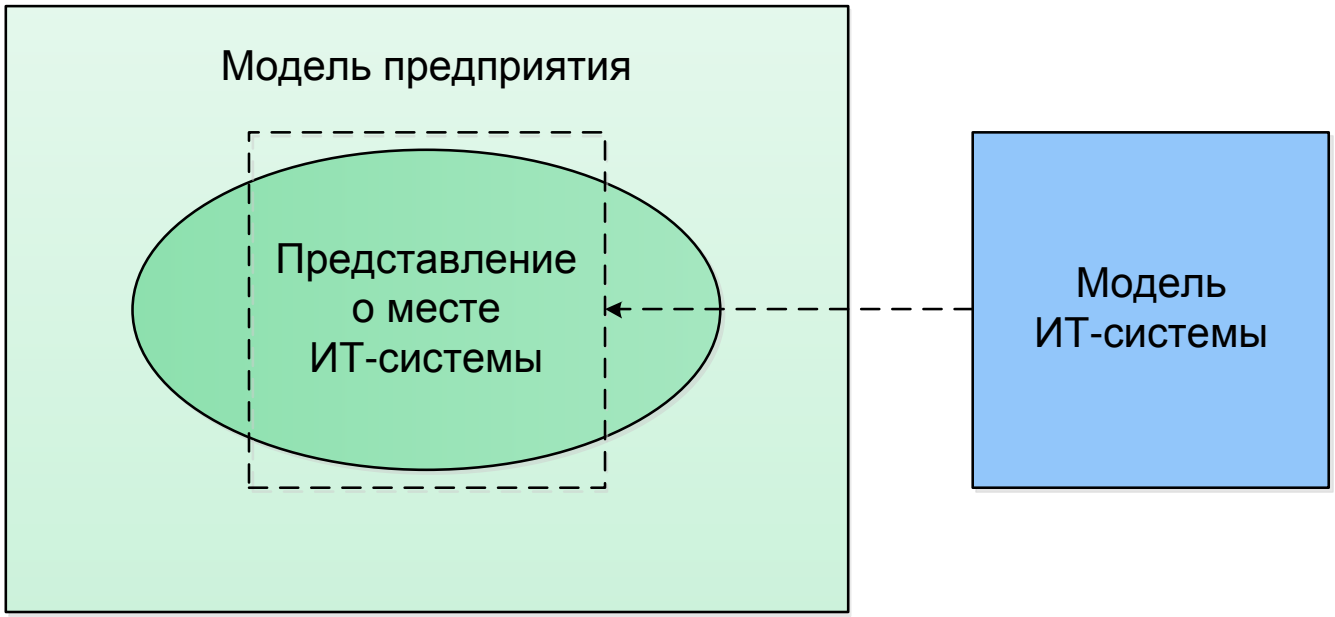
Объекты
с атрибутами
и методами

Диаграмма классов
и другие UML-диаграммы

Типы, соответствующие
бизнес-объектам

Я сосредоточусь на разработке модели,
а не на ее реализации

Зачем нужен единый язык?

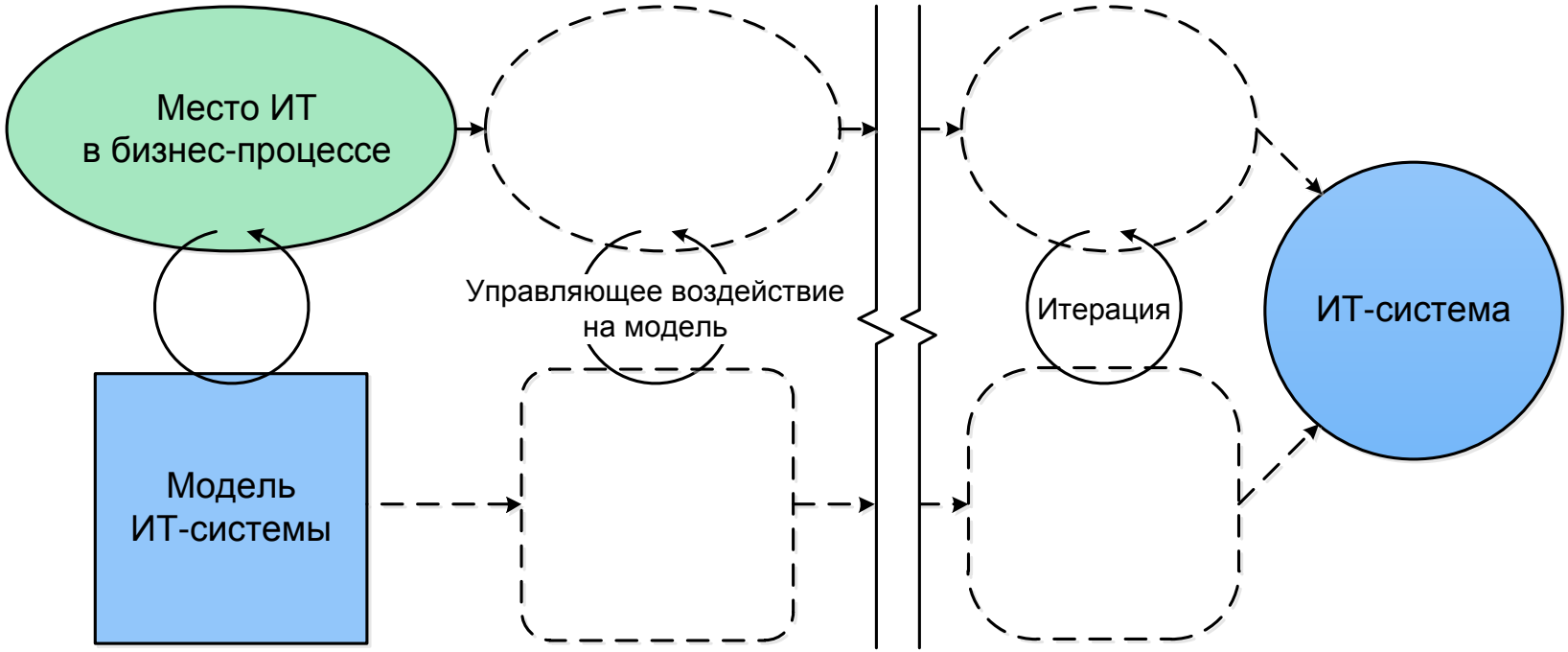


Модель системы не соответствует представлению бизнеса о ее месте в модели предприятия

«Не то чтобы совсем не попал,
но только не попал в шарик…»

CUSTIS®

Итерационное развитие модели



Единый язык позволяет совместить модель системы с представлениями бизнеса о ее месте

CUSTIS®

Единая модель

- Аналитик собирает требования и строит модель: сначала – предметной области, затем – системы
- Артефакты модели описывают и систему, и ее использование в бизнес-процессах предприятия
- Разработчик реализует модель
- Артефакты модели можно проследить в коде

Модель предметной области **становится** моделью системы

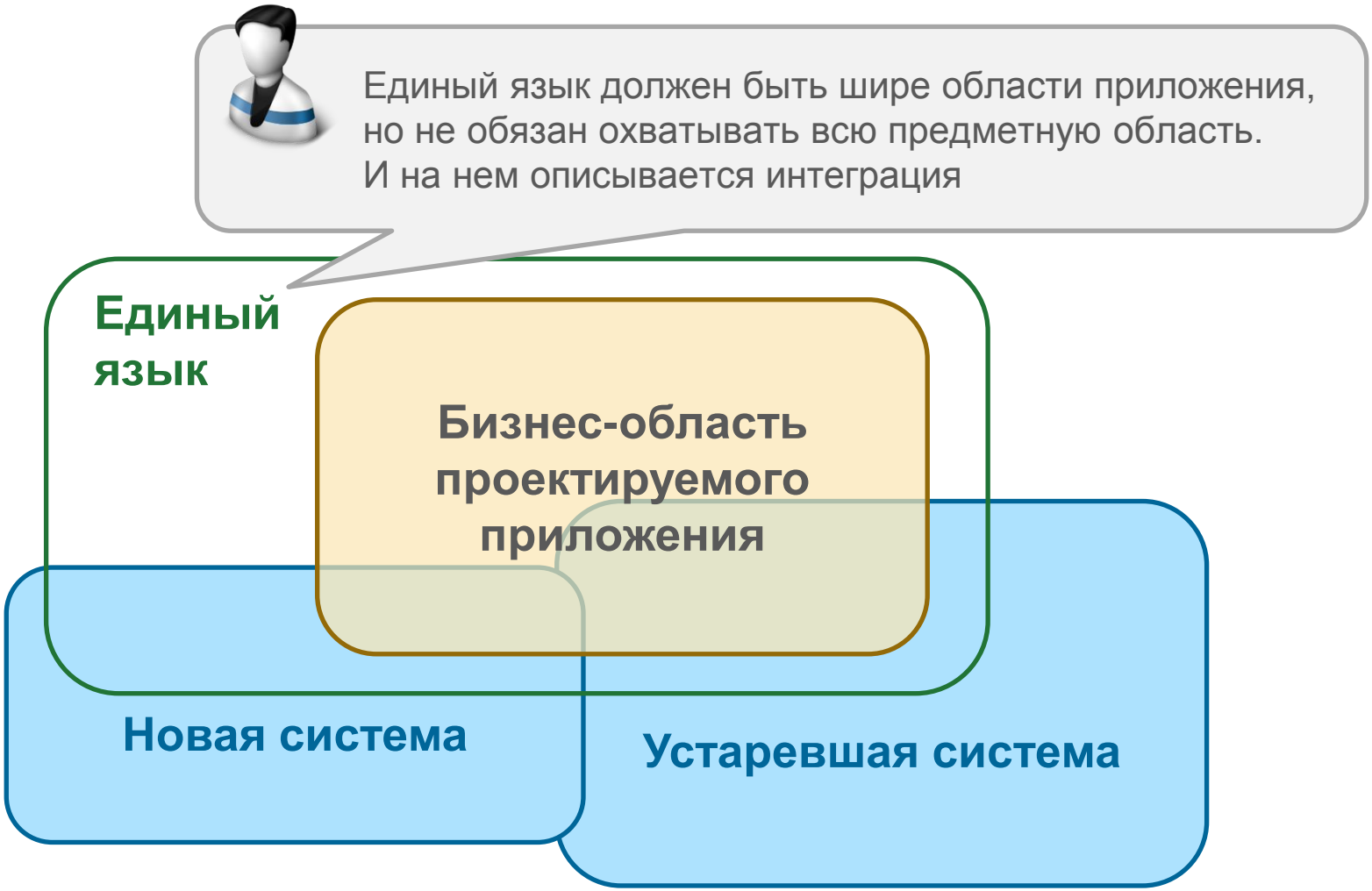
CUSTIS®

Плюсы единой модели

- ☺ Верификация постановок бизнес-специалистами
- ☺ Общее понимание требований на стороне бизнеса
- ☺ Обсуждение модели бизнесом и IT, поиск баланса в сложных решениях
- ☺ Перенос моделей из других предметных областей
- ☺ Бизнес представляет потенциальные возможности системы и сложность различных доработок
- ☺ На этапе эксплуатации – эффективное общение бизнес-пользователей и IT без квалифицированных переводчиков-аналитиков

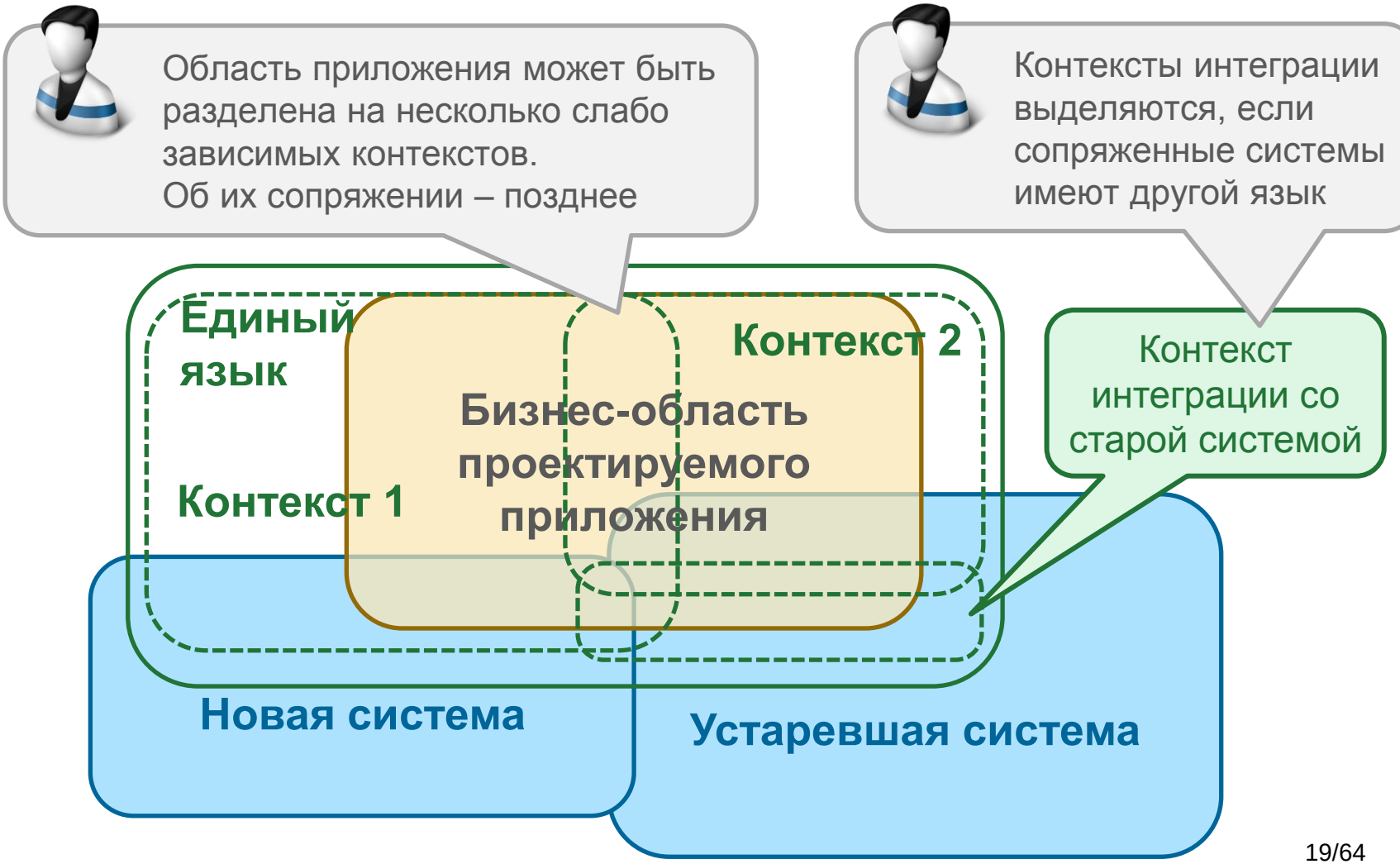
CUSTIS®

Границы единого языка



CUSTIS®

Контексты и их карта



CUSTIS®

Единый язык и слои приложения

Интерфейс работает с объектами предметной области. Язык расширяется понятиями конструирования интерфейса – экраны, таблицы, кнопки…

Единый язык ориентирован на описание модели для бизнес-слоя, его объекты отражаются в реализации

При описании интеграции и хранения в единый язык добавляются понятия описания потоков данных, транзакций и другого межсистемного взаимодействия

Пример:
Виза на документы

Задача

Задача касается конкретного документа

- В процессе обработки документа (сделки, договора, контракта) обеспечить проверку и одобрение определенными сотрудниками или отделами

CUSTIS®

Выбор проектного решения

CUSTIS®

В чем проблема?

- Шаблоны обладают разной гибкостью и сложностью
- Для выбора нужно понимать требуемый баланс гибкости и сложности решения

Традиционный подход

- На этапе сбора требований аналитики формулируют задачу для конкретного документа
- Исходя из этого в системе проектируется решение
- ☹ Выбранное решение отражает текущую ситуацию

Решение надо принимать с учетом потенциального изменения бизнес-процессов

CUSTIS®

Примеры

- Проверка сделки казначейством и отделом рисков – не получается выполнять параллельно
- Юристы отозвали одобрение кредита, а служба безопасности на него опиралась – связи между визами не контролируются системой
- Настройку виз для одобрения договоров с недвижимостью передали в IT из-за сложности

CUSTIS®

Решение в рамках DDD

- Представляем шаблоны, описанные применительно к конкретному документу, показываем разницу
- Бизнес-специалисты оценивают потенциальную потребность в реализации бизнес-процессов

☺ Решение принимается с учетом перспективы

CUSTIS®

В чем единый язык?

- Для решения модель надо описать понятно бизнесу
  - Можно описать обобщенные решения для документов, «состояния» и «визы», и на них ссылаться
  - Можно описывать каждый случай отдельно
- Термины должны быть понятны заказчику
  - Например, «визированием» могут считать одобрение документа, требующее только просмотра, а если требуется дополнительная работа, то это называется «обработкой» или «проверкой»
- Общий шаблон надо «перевести» на язык проекта

CUSTIS®

Результат

- ☺ Мы используем известные шаблоны решений
- ☺ Заказчик оценивает вариант решения не только с точки зрения текущих потребностей, но и из предположений о развитии бизнес-процессов
- ☺ Проектируя изменения бизнес-процессов, заказчик представляет потенциал гибкости системы и принимает решения с учетом этого

CUSTIS®

Точки расширения

Вопрос: Как обеспечить легкое развитие системы при изменениях правил проверки и одобрения документа?

Ответ: Для этого нужны точки расширения.

- Стратегии – именованные алгоритмы обработки, включаемые в определенных точках
  - Проверки или обработка при переходе в состояние
  - Алгоритм определения состояния документа по визам
- Стратегии дополняем спецификациями – декларативными описаниями условий, применяемых для выбора стратегий по параметрам документа
  - Декларативное описание графа перехода документа и набора виз, связанных с переходами

CUSTIS®

DDD
для корпоративных приложений

Часть 1. Общая схема

CUSTIS®

Типичное корпоративное приложение

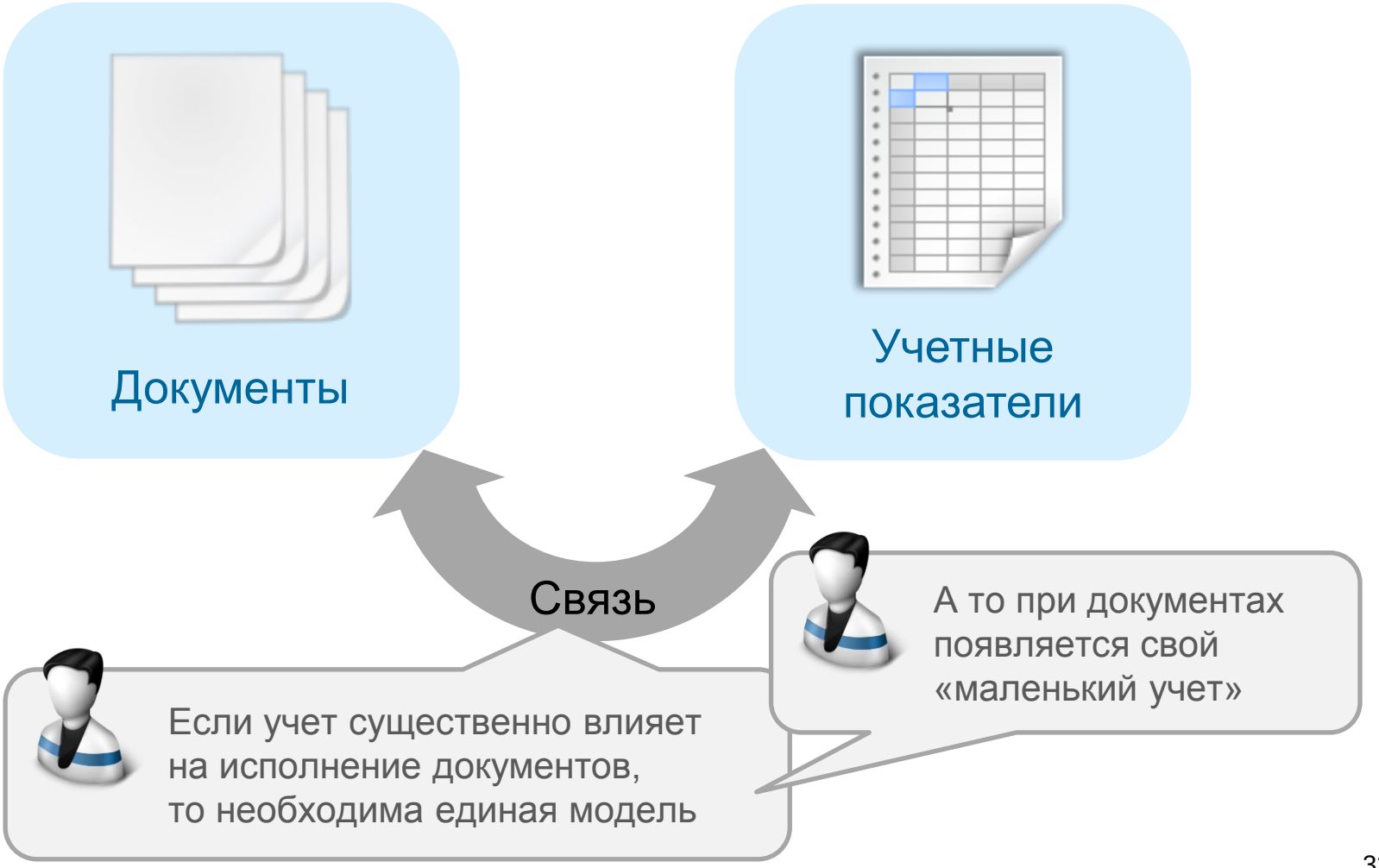
- Пользователи создают документы
- По необходимости заполняют справочники
- Потом документы исполняют
- При этом меняются учетные данные
- Которые влияют на исполнение документов
- И отражаются в отчетах

Объектная модель

Жизненный цикл документа

Учет –
не объектная модель

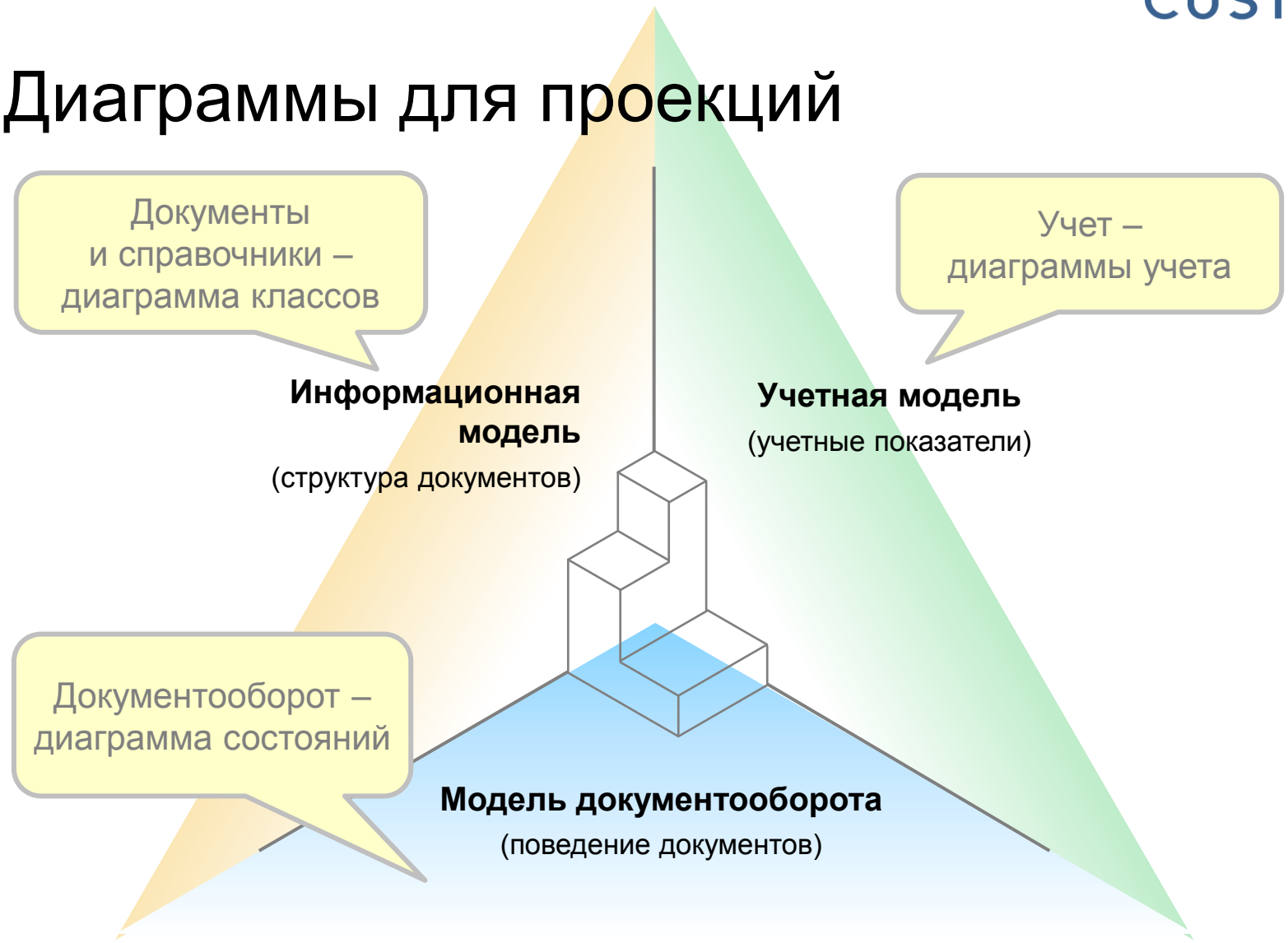
Одна модель или несколько?



Три проекции модели системы

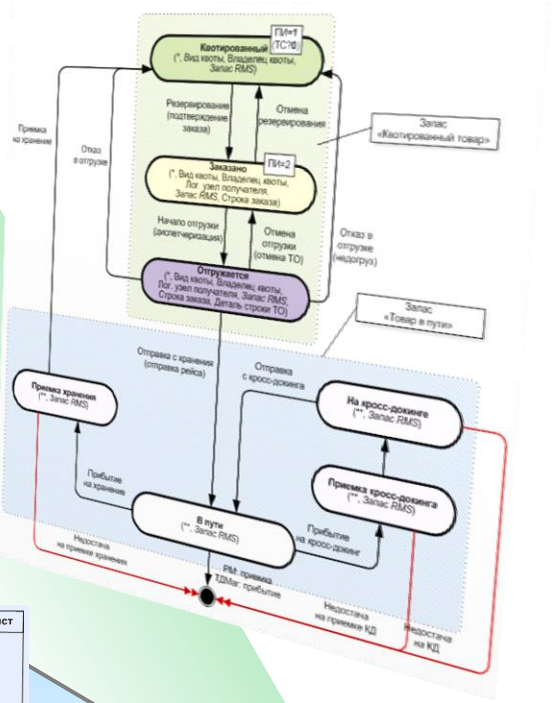
CUSTIS®

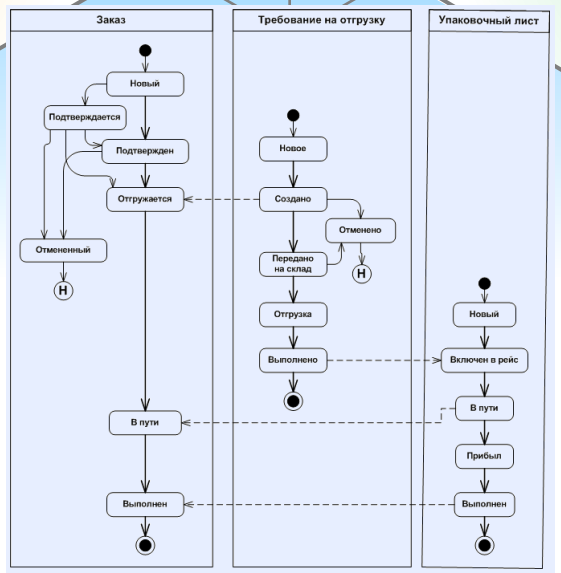
Диаграммы для проекций



CUSTIS®

Диаграммы для проекций





CUSTIS®

DDD
для корпоративных приложений

Часть 2. Учетная модель

CUSTIS®

Учетная модель – не объектная

Сложность объектного представления учета:

- Нет идентификации единичного объекта
- Работа идет с показателями, текущее значение которых меняется
- Изменение числового значения может менять состояние с точки зрения принятия бизнес-решения
- Часто интерес представляют агрегаты, а не отдельные значения

Представление учета оказалось за рамками UML. Для него не придумано эффективного представления

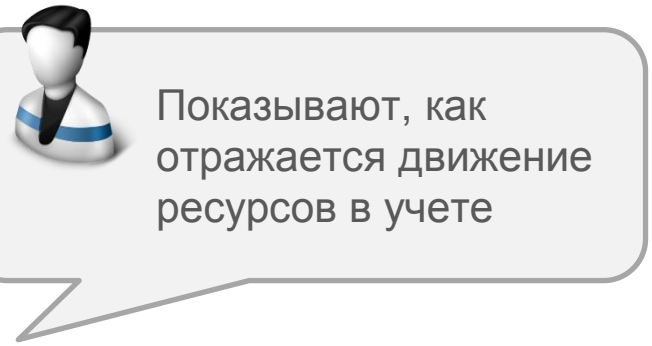
CUSTIS®

Учетная модель

- Для представления учетной модели мы придумали диаграммы учета
- Они показывают элементы учета
  - Какие есть синтетические счета и их аналитику
  - Как проводки перемещают ресурсы по синтетическим счетам
  - С какими событиями связано исполнение проводок

Статья «Диаграммы учета: мост между бухгалтером и разработчиком» (журнал «Бухгалтер и компьютер», №5-2011)

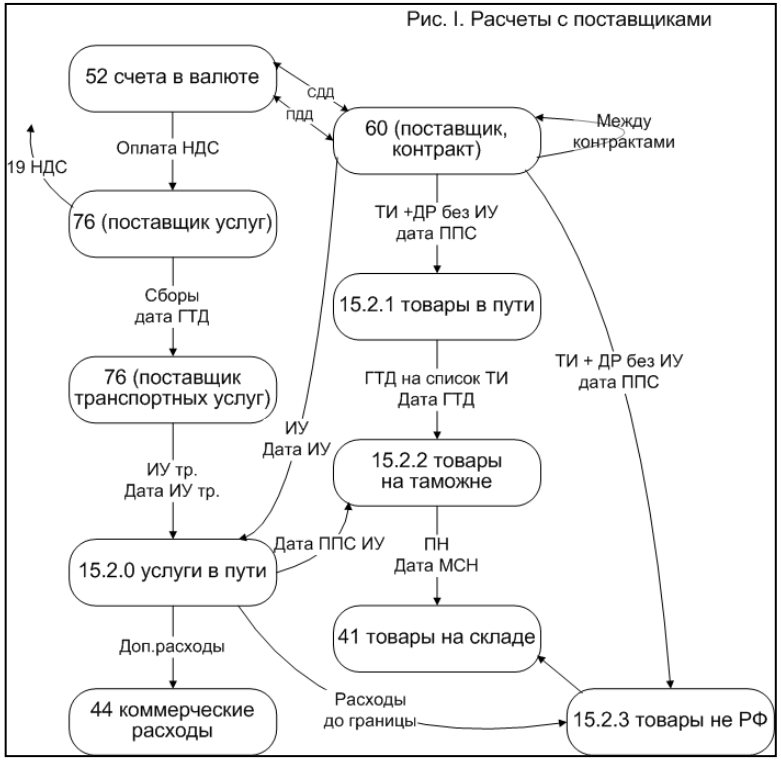
Диаграммы учета

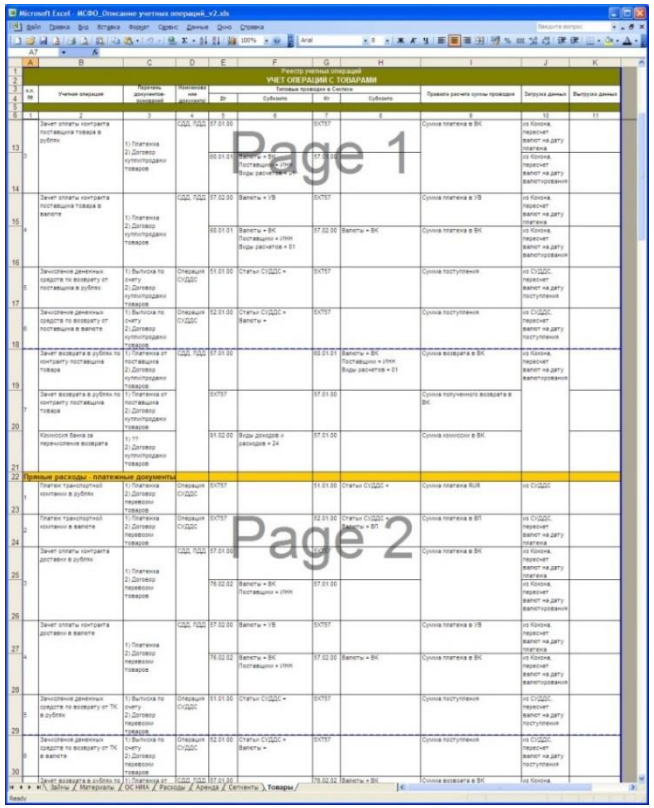


Диаграммы учета для бизнеса

Бухгалтеры могут применять диаграммы учета для разработки учетной политики.

Они нагляднее, чем Excel





CUSTIS®

DDD
для корпоративных приложений

Часть 3. Документооборот

CUSTIS®

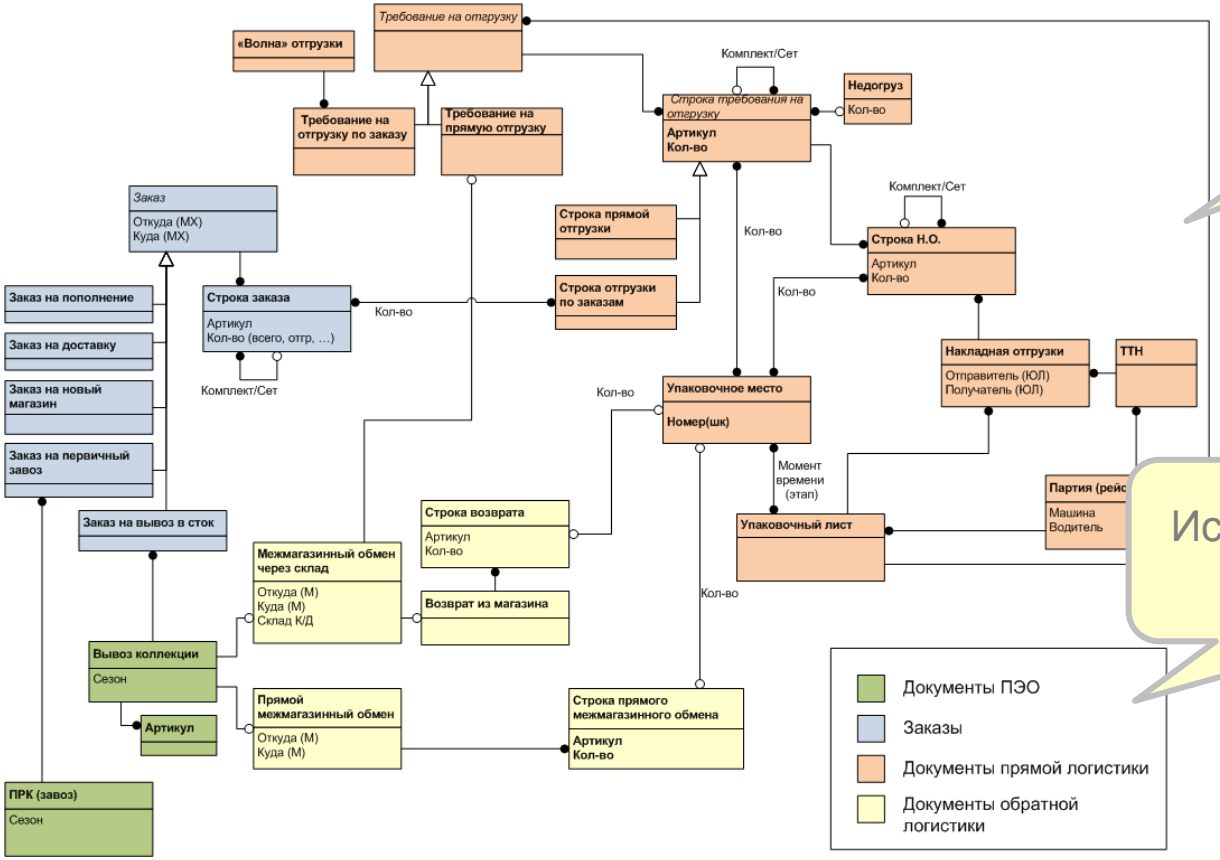
Хорошо работает объектная модель

- Объекты с атрибутами и методами – элементы единого языка для предметной области и способ их соединения в сложные конструкции
- Для отражения документооборота используем состояние документа
- Диаграмма классов и диаграмма состояний UML – визуальный образ для наглядного представления
- Объекты в программе – способ отражения модели в реализацию

CUSTIS®

Модель должен понимать заказчик

Классы соответствуют бизнес-объектам – заказчик видит знакомые названия



Представляем диаграммой классов

Используем цветовые выделения

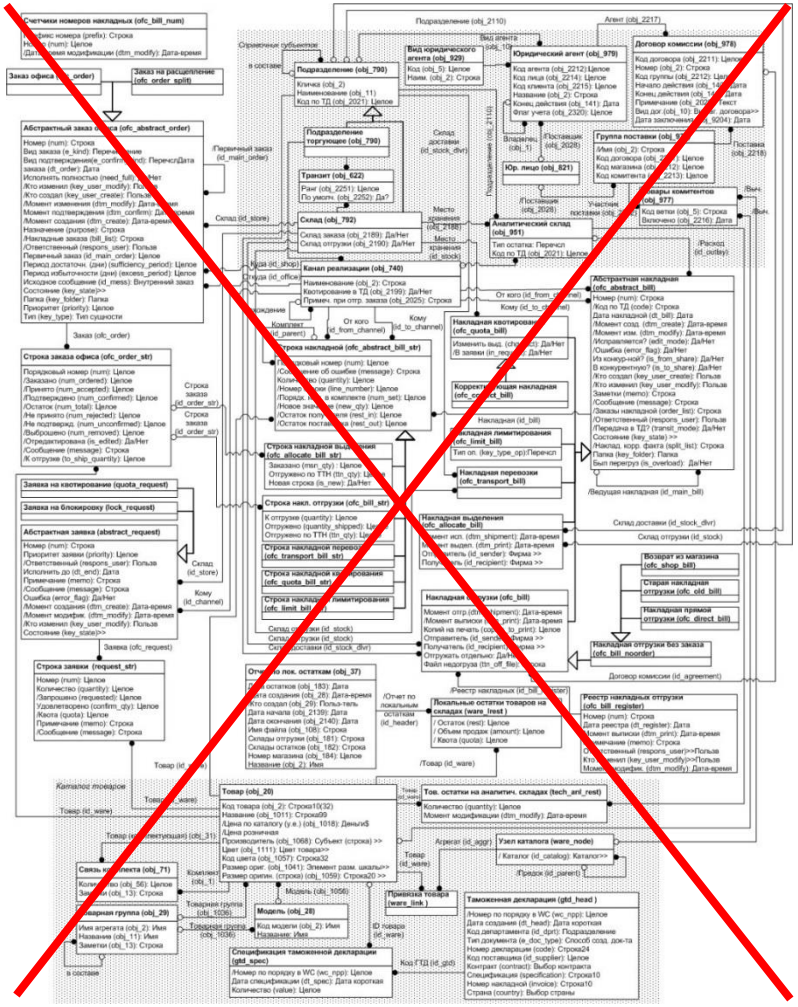
CUSTIS®

Не нужно усложнять диаграмму

☹ Не нужно

- Стараться изобразить все классы на одной диаграмме
- Отображать вспомогательные атрибуты
- Использовать технические коды
- Использовать сложную нотацию

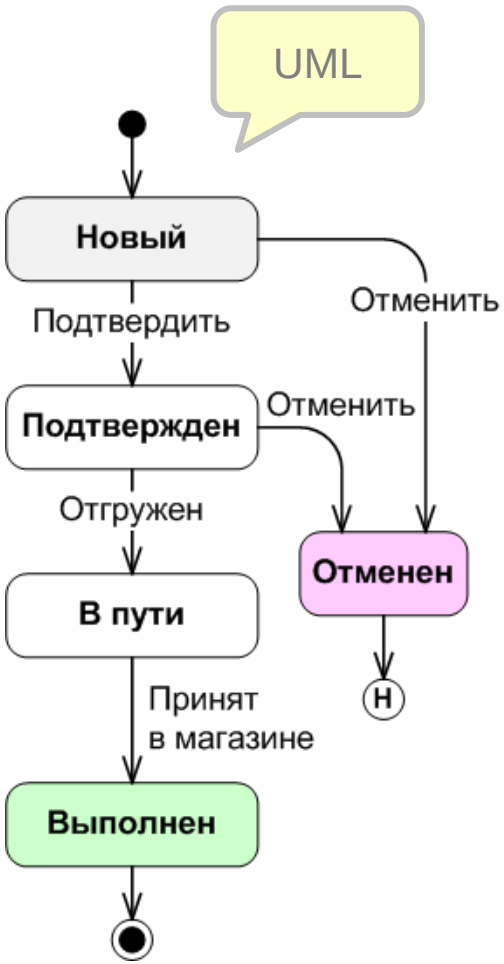
☺ Диаграммы должны понимать заказчики, а не только разработчики



CUSTIS®

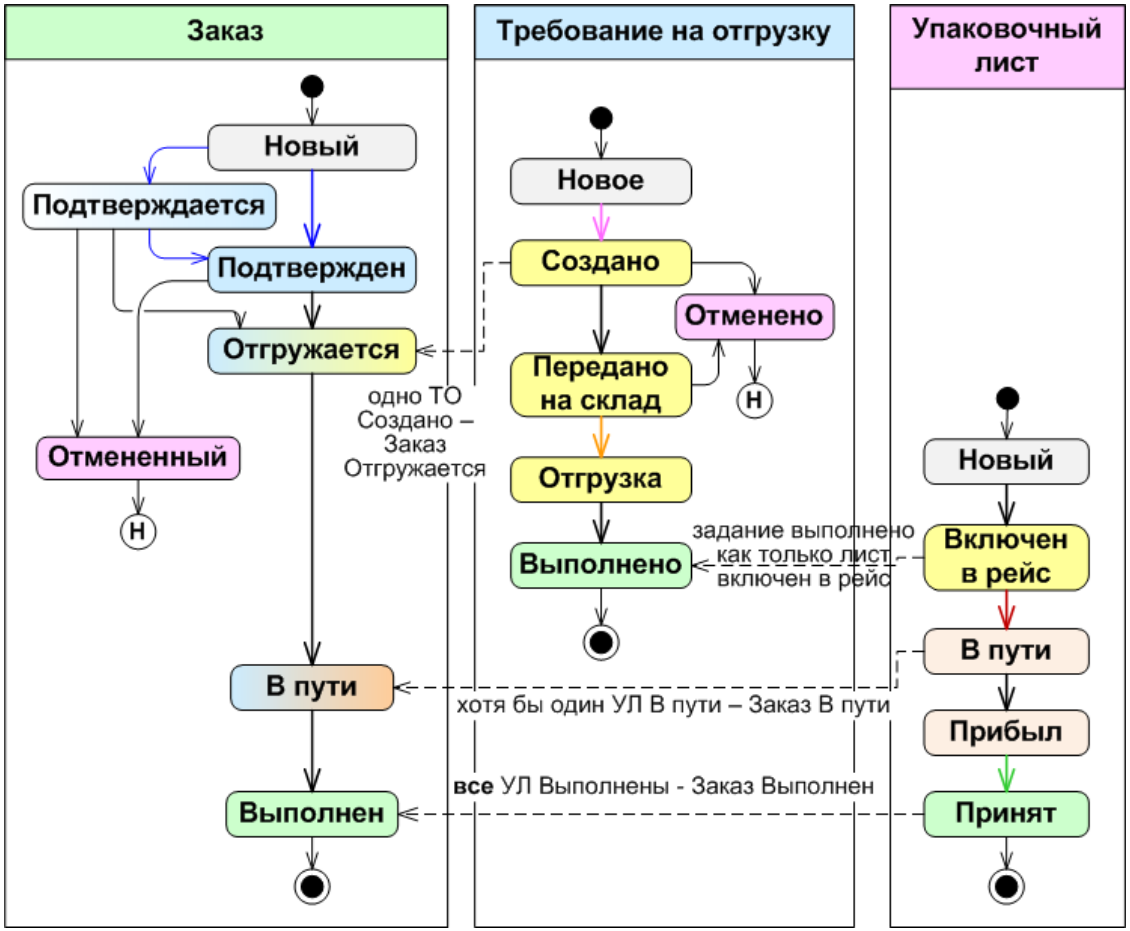
Модель документооборота

- Документу приписываем состояние, оно определяет этап жизненного цикла
  - Какие действия можно совершать и кому
  - Кто отвечает за обработку документа
- Возможные изменения состояний документа образуют граф переходов – State machine diagram



Язык ООП «с расширениями».
Названия состояний и переходов – на языке бизнеса

Наглядно представляет сложный документооборот



CUSTIS®

Точки расширения в логике переходов

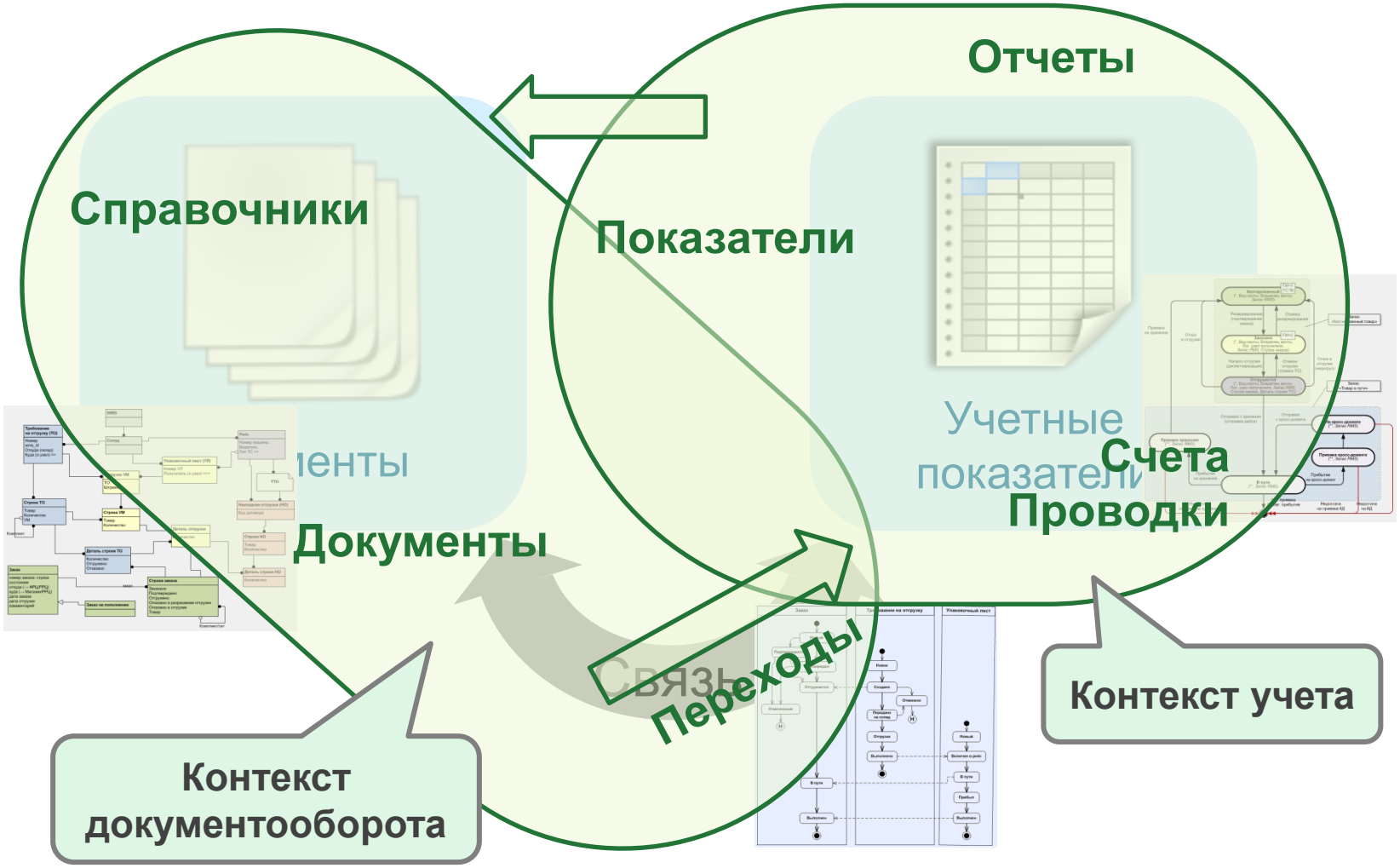
- Сочетание декларативного и императивного
- Типизация документов, графы при наследовании
- Если граф переходов настраивается – надо уметь подхватывать настройки в интерфейсе и логике
- Если в любом случае требуется кодирование – в чем будет выгода от настроек?
- Развитие правил обработки на переходе
  - В коде через наследование объектов
  - Через стратегии в точках расширения
  - Через декларативное описание правил выбора стратегий
- Настройка прав доступа

CUSTIS®

DDD
для корпоративных приложений

Часть 4. Разделение контекстов

Контексты в единой модели



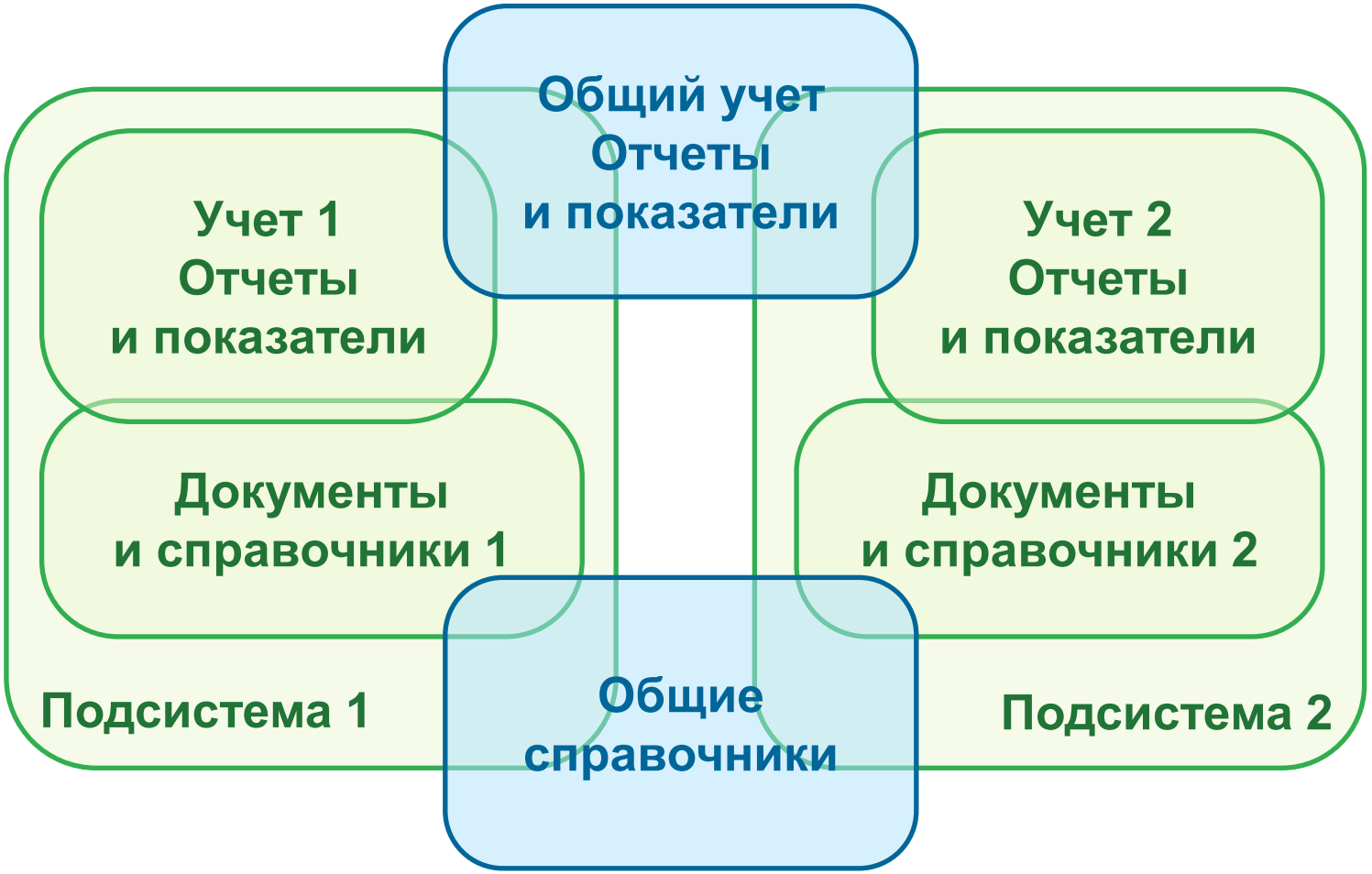
CUSTIS®

Вертикальная декомпозиция

- Розничный магазин
  - Продажи и Склад магазина
  - Продажи, Склад магазина, Заказы товара
- Торговая компания
  - Закупки и Продажи
  - Закупки, Продажи и Склад
- Оптовые продажи
  - Заказы и Отгрузки
  - Заказы, Отгрузки, Оплаты

CUSTIS®

Вертикальная декомпозиция



CUSTIS®

Сопряжение контекстов

- **Примеры**
  - Магазин: со складом и без склада, продажа с заказом
  - Логистика и склад: много систем с заказами товара
  - Банк: Главная книга и много систем по банковским продуктам
- **Подходы**
  - Смысловое ядро
  - Общее ядро
  - Абстрактное ядро
  - Подключаемые компоненты
  - Крупноблочная структура
  - Уровни разделения обязанностей

CUSTIS®

Еще пример:
Долг клиента

CUSTIS®

Задача

Ведение взаиморасчетов с клиентами
по продажам

- Обеспечивающее ведение управленческих лимитов
- И отражение расчетов в бухгалтерию

CUSTIS®

Проблема:
Смешение языков на бизнес-уровне

- Менеджеры по продажам: долг – основа для управленческих решений. Увеличивается, как только приняли решение об отгрузке, уменьшается, когда признали претензию

- Бухгалтеры: долг – в соответствии с ПБУ, с учетом оформления документов и прохождения процедур

- Следствие: управленческий и бухгалтерский долг имеют систематические различия

CUSTIS®

Процесс отгрузки товара

CUSTIS®

Проблемы двух пониманий долга

- ☹ Контроль правильности учета запаздывает
- ☹ Менеджеров не получается контролировать через бухгалтерию
- ☹ Имеются две разные суммы долга, что затрудняет принятие управленческих решений
- ☹ Для сверки с клиентом и решения проблем менеджеры должны вникать в ПБУ

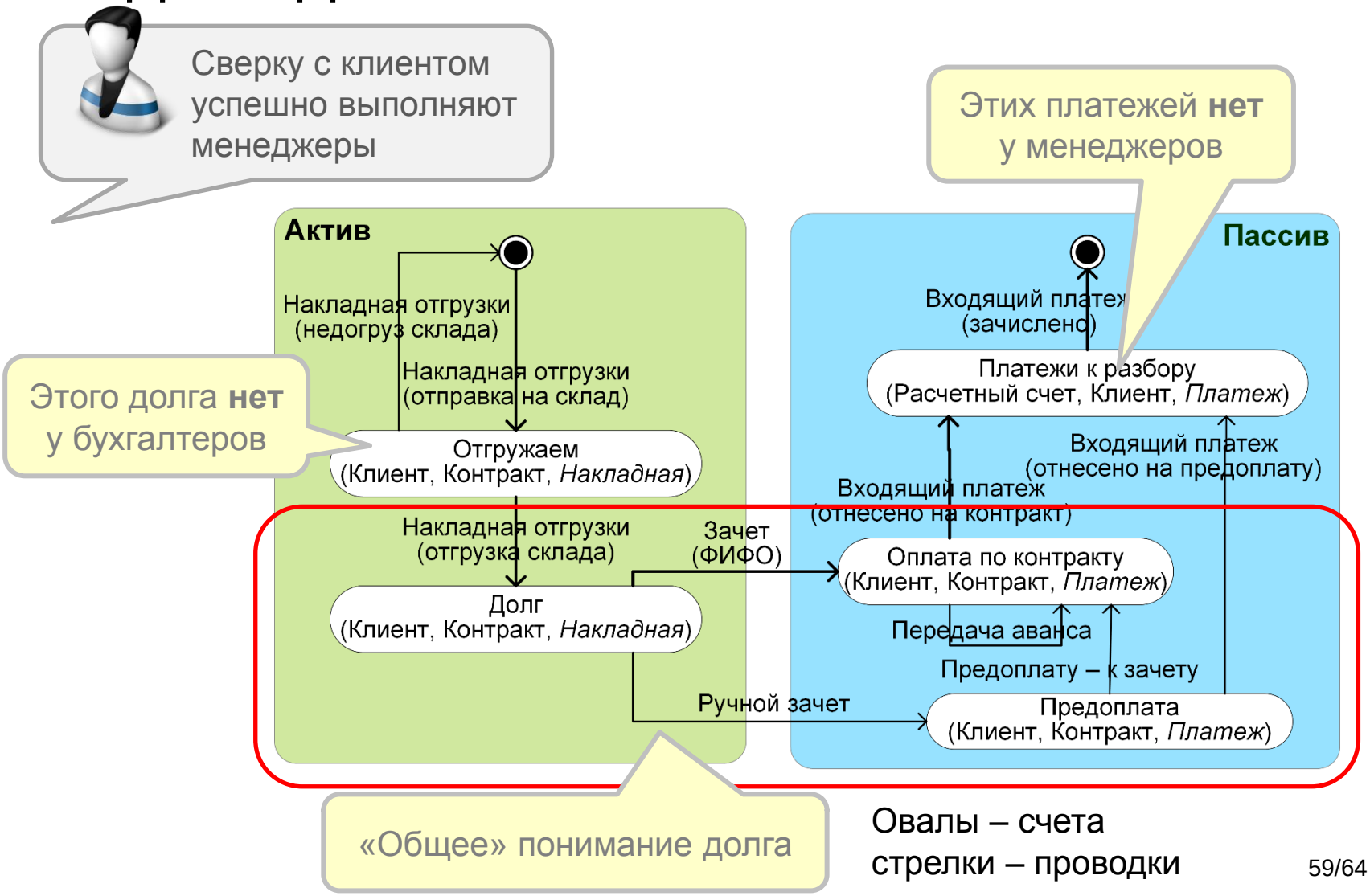
CUSTIS®

Решение от DDD

- Вырабатываем единый язык, описывающий долг в терминах, понятных менеджерам и бухгалтерам
- Строим модель, которая показывает управленческий и бухгалтерский долг и их различие
- Ситуации, в которых долг различается, описываем на едином языке, согласуем со специалистами
- Вырабатываем требования по контролю различий, а также по устранению несущественных различий

☺ Результат – общий взгляд у бизнес-специалистов на предметную область, описанный в модели, которая реализуется разработчиками

Модель долга клиента



CUSTIS®

Результат

- ☺ Согласовано понимание предметной области у различных бизнес-специалистов
- ☺ Реализована модель, отвечающая интересам всех заинтересованных сторон проекта

CUSTIS®

Заключение

CUSTIS®

Практики проектирования применяем для бизнес-анализа

- Ограниченные контексты и их изоляция
- Способы выделения общего в контекстах
- Стратегии и Политики
- Выделение общих объектов
- Абстракция через интерфейсы

Технические практики наполняются бизнес-смыслом и служат для общения с заказчиком

CUSTIS®

Что же достигает DDD?

- Единый язык + Единая модель:
  - Дают надежную основу для общения всех участников проекта при принятии решений
  - Успешно заменяют мелкую россыпь требований
  - Позволяют эффективно развивать сложную систему
- Это требует дополнительных усилий:
  - Формирование единого языка
  - Понимание разработчиками предметной области
- По опыту, результат окупает усилия

CUSTIS®

Спасибо!

Вопросы?

Максим Цепков

maks@custis.ru

www.facebook.com/mtsepkov