

Аналитик и legacy: как разобраться в устройстве старой системы?



Максим Цепков

Главный архитектор решений CUSTIS

Навигатор по миру Agile, бирюзовых организаций и спиральной динамики

<http://mtsepkov.org>

23-24 мая 2025

Санкт-Петербург

Немного обо мне

- Создание и внедрение больших корпоративных систем (более 25 лет)
 - Знание практик операционного управления и ведения проектов в крупных коммерческих и государственных организациях и банках
 - Опыт управления проектами в ИТ: от инженерного подхода и РМВОК – к современным Agile-методам (с 2007 года)
 - Опыт перестройки процессов организаций при внедрении систем
- Навигация в менеджменте цифрового мира
 - Agile и менеджмент самоуправления: бирюзовые организации, холакратия и социократия ([книга](#), [статьи](#) и [выступления](#))
 - Модель спиральной динамики (с 2013 года) и другие модели soft skills, **модели личности** и самоопределения ([книга](#), [статьи](#) и [выступления](#))
 - СМД-методология и развитие СРТ при переходе в цифровой мир



На моем сайте mtsepkov.org – мои выступления и много других материалов

О чём этот доклад?

Задача: разобраться в устройстве старой работающей системы

- Чтобы спроектировать доработку в ней нового функционала
- Чтобы заменить мягко новой системой
- Чтобы рядом построить систему, которая добавит новые функции

Знания часто утеряны, а документы – устарели, хотя система живет

Я расскажу свой опыт работы с enterprise-решениями

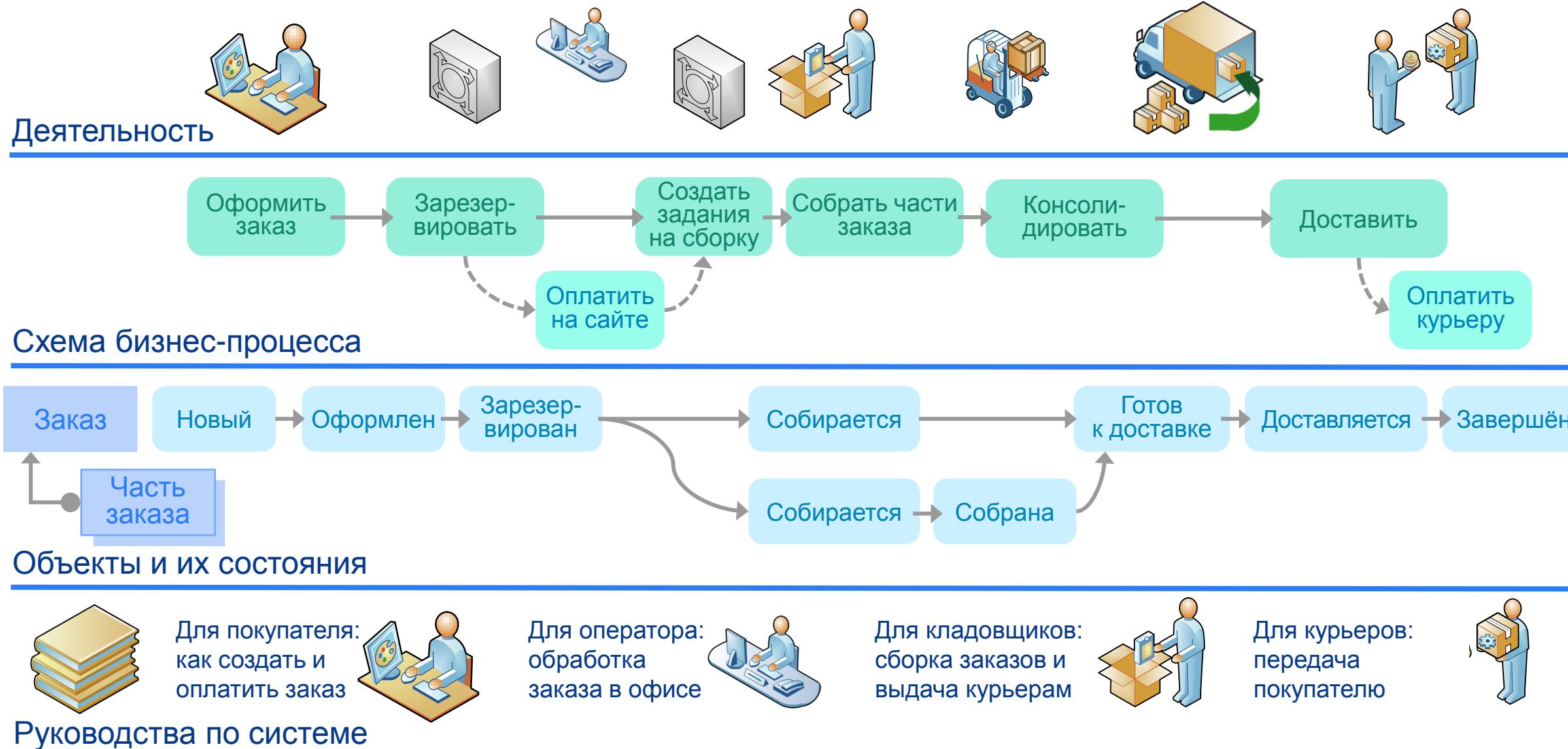
- Какова типовая схема устройства старых enterprise-решений?
- Откуда брать информацию, каких искать стейкхолдеров?
- Как описывать бизнес и его поддержку в ИТ-системах?



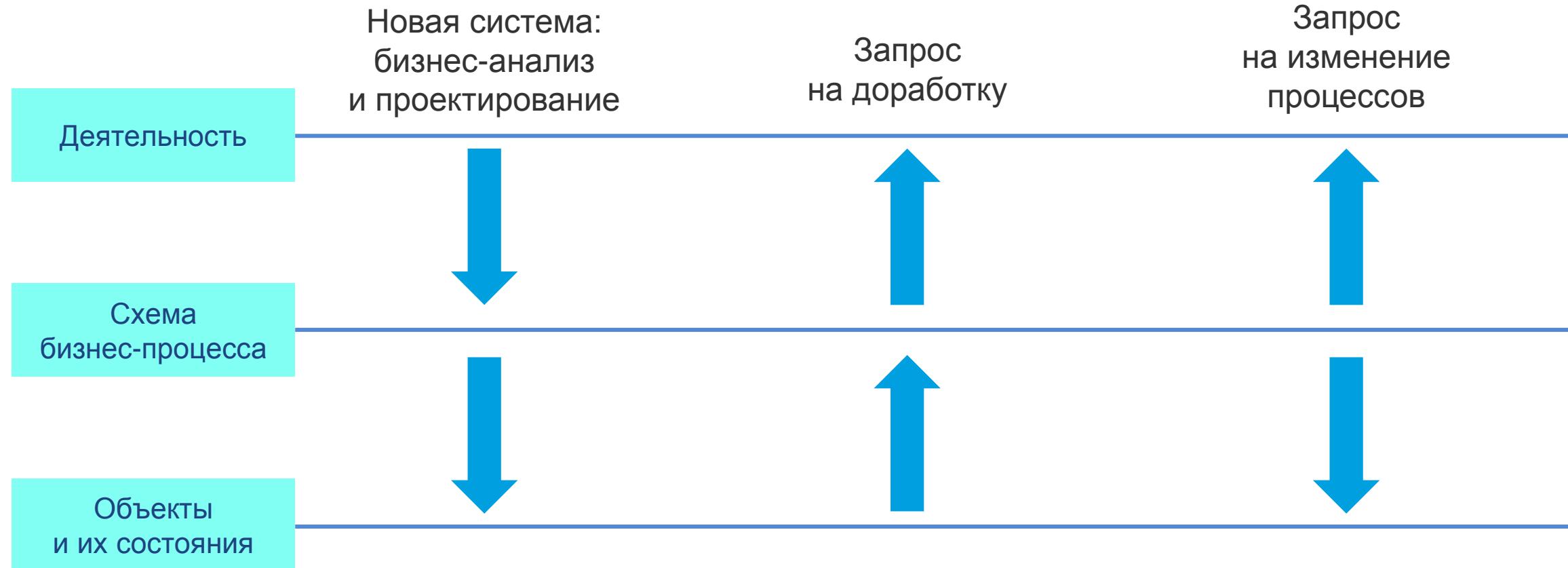
Я работал в разных предметных областях: торговля, банки, производство.
Методы – аналогичны, большинство примеров из торговли, потому что там проще.

Какая нужна информация?

Уровни представления (интернет-заказ)



Разные задачи автоматизации



Об учебниках: три категории постановок

1. Разработка новой системы поверх фрагментарной и малой автоматизации:
 - выясняем модель бизнес-процессов,
 - создаём модель системы,
 - работаем над её встройкой в процессы, изменяя их.
2. Доработка существующей системы: проектируем изменения модели существующей системы и её встройки в процессы.
3. Разработка новой системы, заменяющей существующую:
 - существующие процессы несут отпечаток старой системы, **его надо снять**,
 - проектируем новую систему и процессы,
 - проектируем работу на переходном этапе, обеспечивая мягкую замену.



Большинство методов анализа и проектирования создавались, когда бизнес-процессы были слабо автоматизированы, в расчёте **на первую ситуацию**. Сейчас мы имеем дело **со второй и третьей**.

Если пришли в живой проект – ищите, кто расскажет!

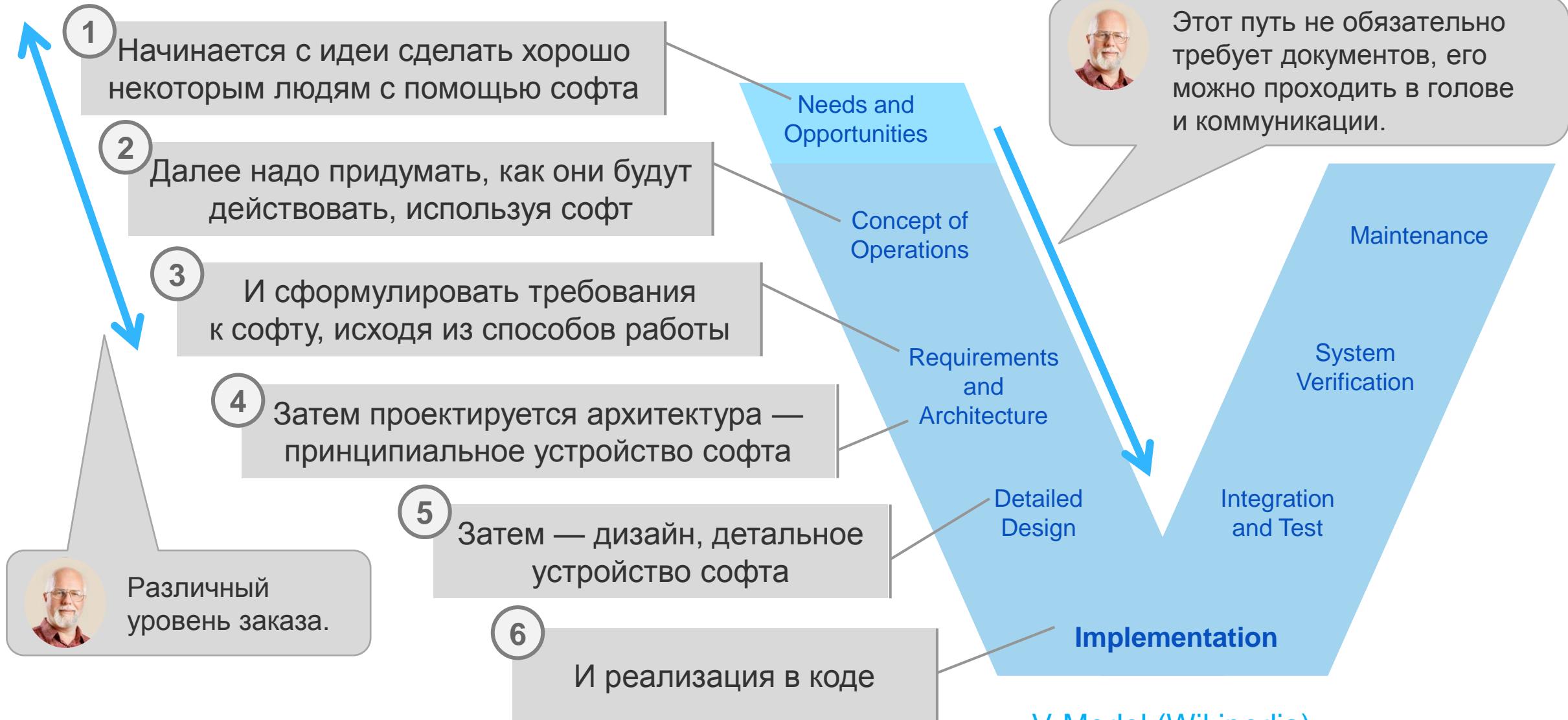
Обычно есть **две команды**: бизнес-заказчик и команда разработки

- Узнать про команду заказчика и их разделение ответственности, их отношение к системе и команде проекта
- Узнать разделение ответственности в команде разработки и опыт людей
- Узнать историю проекта, найти человека, который в проекте давно и попросить рассказать - он часто может поделиться старыми историями

Откуда узнать про легаси у заказчика?

- Общие представления про бизнес, включая публичные отчеты самой компании или статьи о векторах ее развития – там видишь специфику
- Сопоставить описания бизнес-процессов с инструкциями по системе и экранными формами – дает терминологию и проблемы, даже устаревшее
- Сопоставить инструкции с описанием стандартного продукта (1С, SAP)
- На интервью – спрашиваем о проблемах и просим показать в системе, иначе легко недооценить проблему, например, невозможность исправить документ на середине workflow
- Просим у пользователей реальные документы, и лучше – не отдельные образцы, а выгрузки массивов документов, чтобы посмотреть в Excel

Пробелы в документах — норма



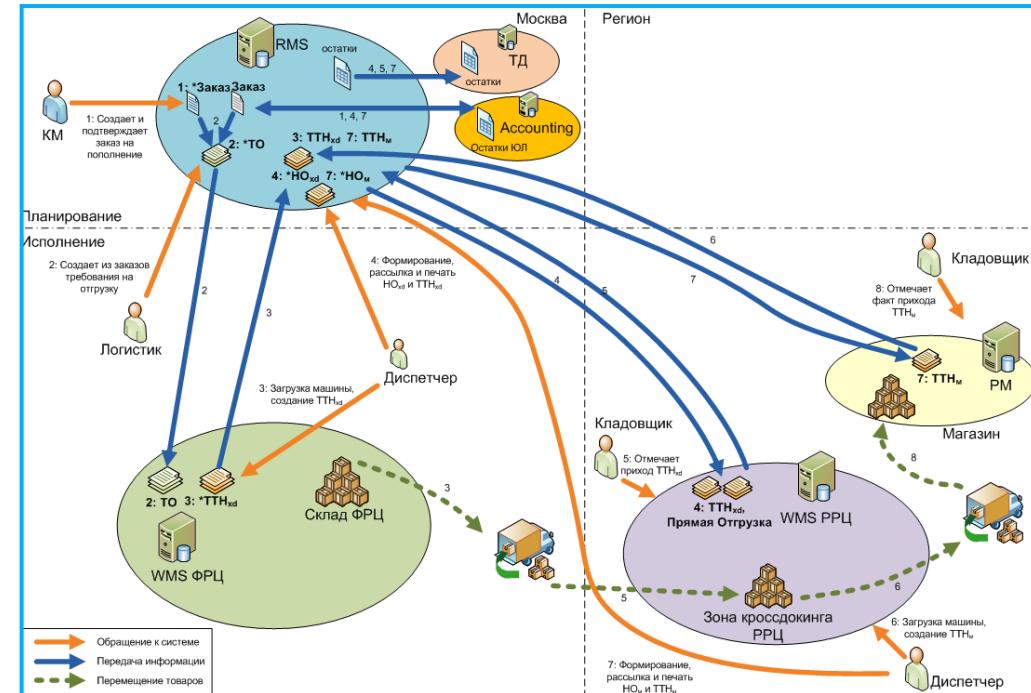
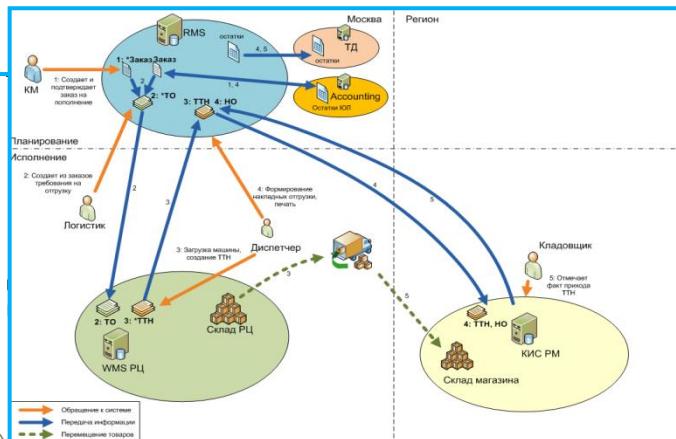
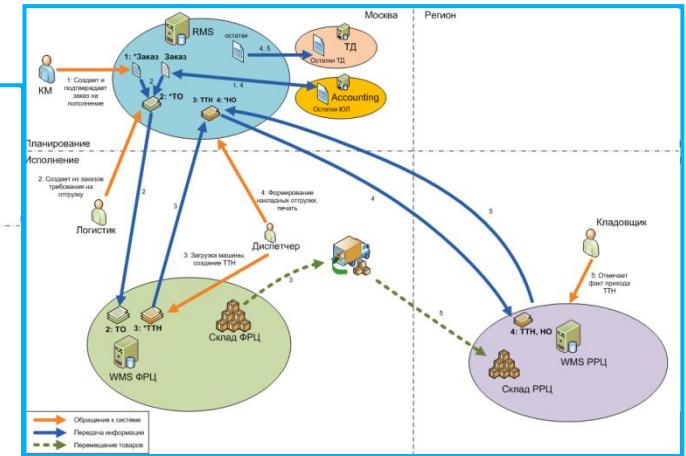
Понять устройство бизнеса



Строим модель, а не просто рисуем
схемы и описываем бизнес-процессы

Неформальная модель бизнеса

Мы можем построить формальную модель процесса и его реализации, но не всегда заказчик может её проверить. Часто он доверяет, а на внедрении вскрываются проблемы. Решение — вернуться в неформальную модель или показывать прототипы.

**2****3**

Снабжение магазинов: на неформальных моделях показывали кейсы работы будущей системы.

Формальная бизнес-модель и реальность

Оптовые продажи магазинам и торговым сетям

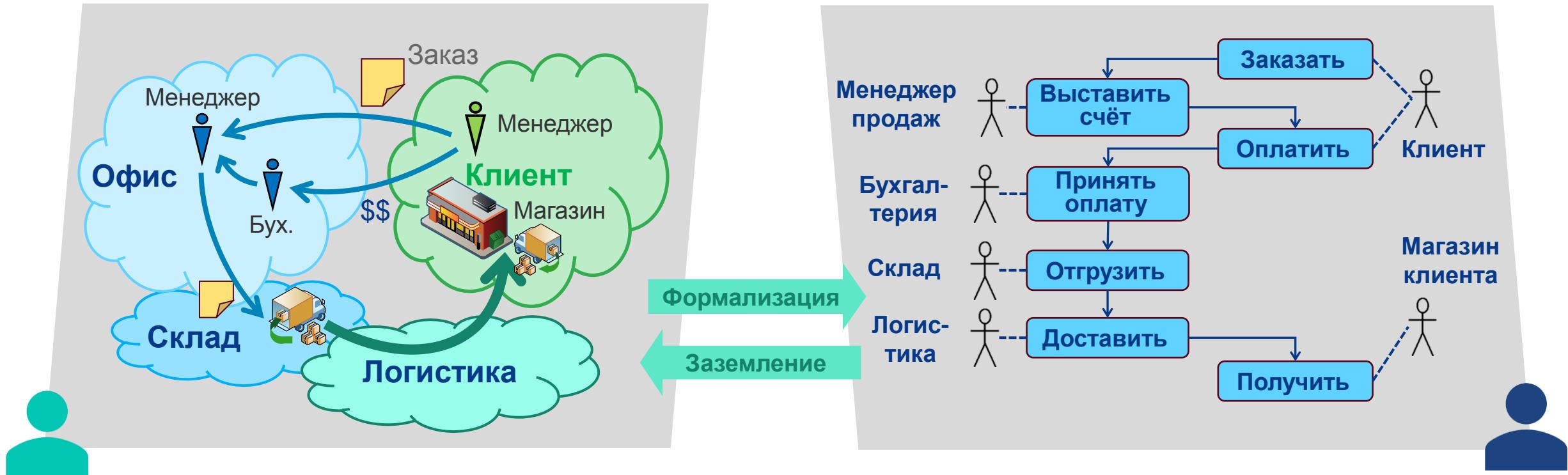
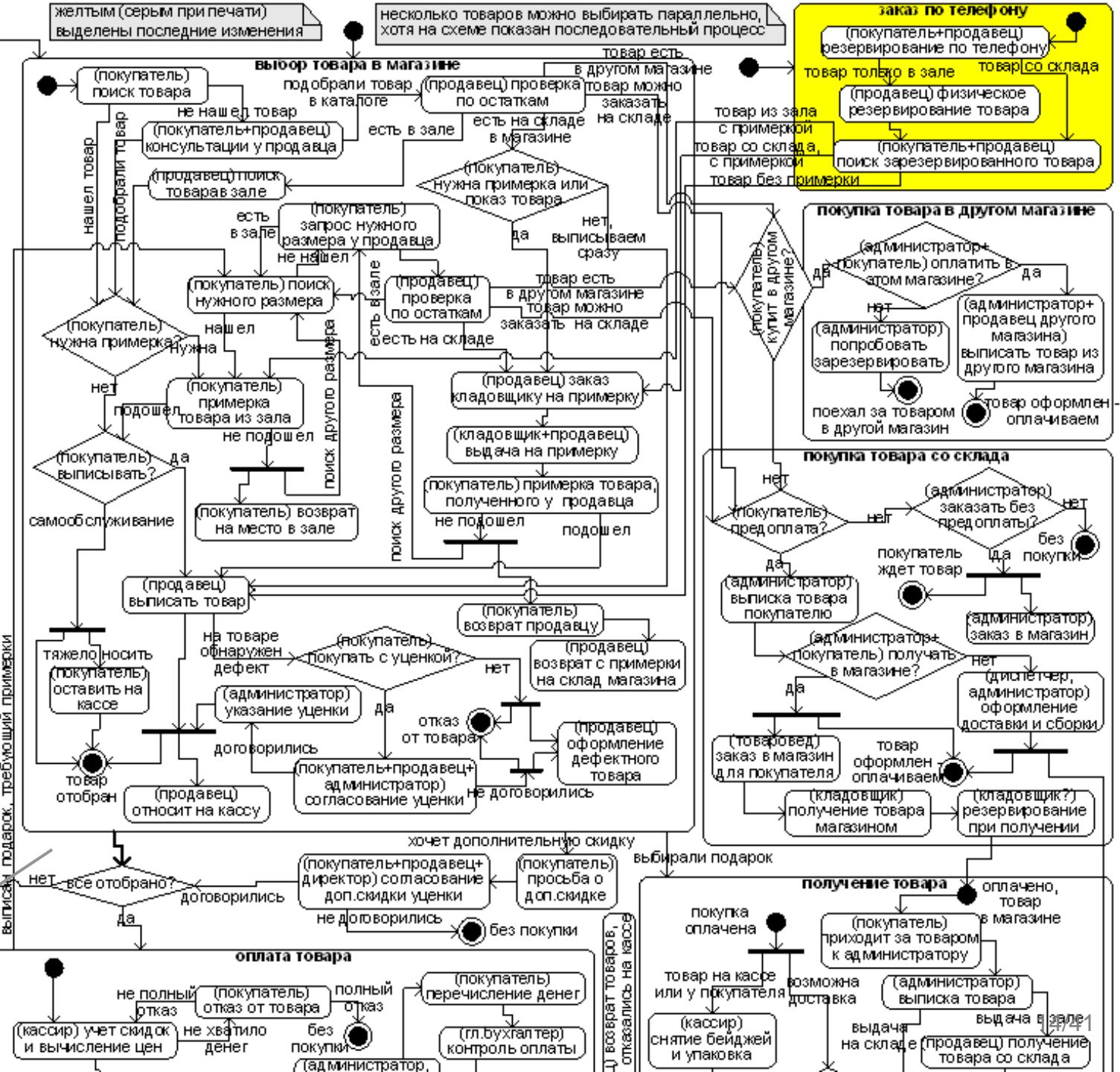


Схема бизнес-процессов – это модель происходящего в реальности.
Видим повседневную деятельность за формальной бизнес-моделью!

Диаграммы деятельности

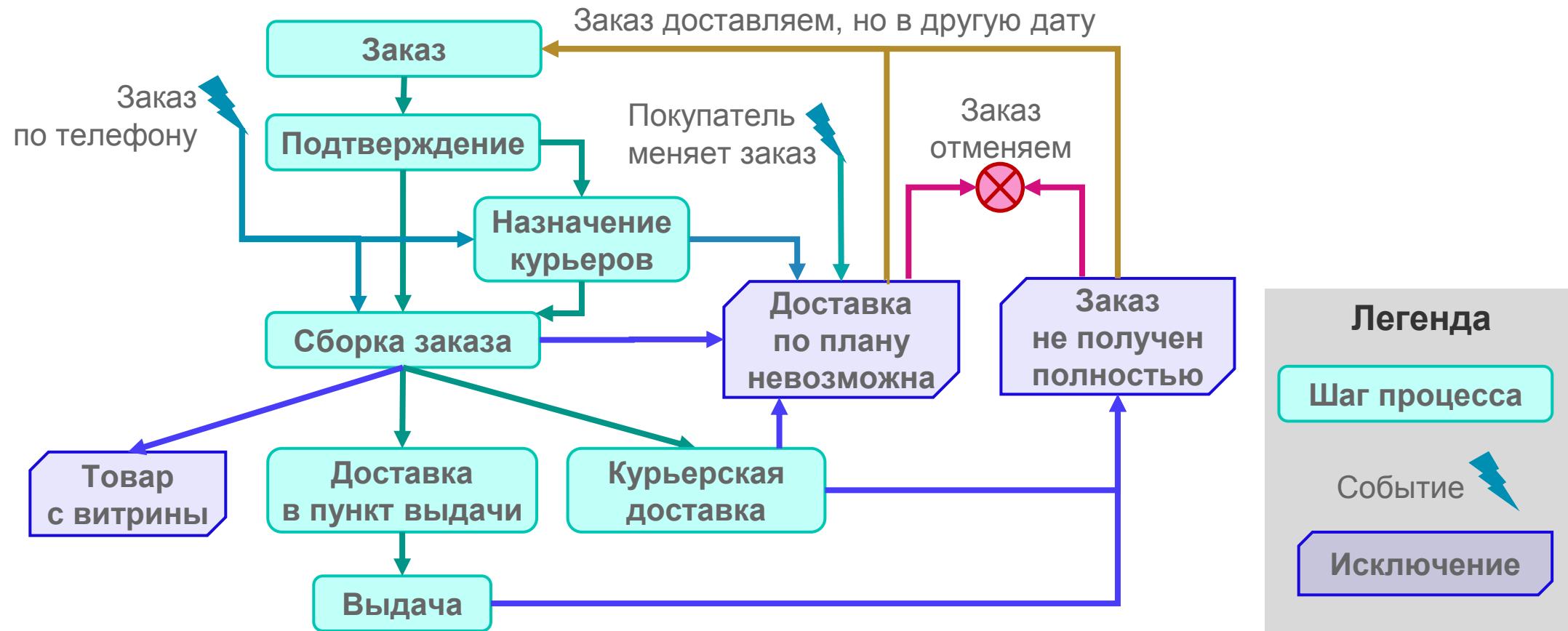
- Даже в «очевидном» процессе обслуживания покупателей выясняешь много особенностей
- Заказчик проверяет детали лучше, чем по тексту
- Однако, столь сложных схем лучше избегать, разделять их на несколько

Это activity diagram UML, отличия от BPMN не существенны. Начиналось с простой схемы, она усложнялась за счет побочных ветвей. Люди уже были в контексте, схема работала, поэтому не перерисовали



Исключения: Case Management

Процесс обработки заказа интернет-магазина с исключениями из моего доклада «[Process и Case Management в информационной системе: от автоматизации As Is к поддержке развития бизнеса](#)».



Не живите в матрице!

- Аналитики часто воспринимают деятельности через документы и объекты информационной системы
- Это же свойственно многим специалистам бизнеса
 - Бухгалтерам и другим специалистам, работающим только с документами в системе
 - Многим менеджерам, которые руководят «из кабинета»
- Это проявляется во время интервью, а до работающих руками – не доходит

Способы борьбы:

- Представляйте реальную деятельность, а не схему из квадратиков
- Доходите до тех, кто «делает руками», смотрите работу, спрашивайте их



В **Lean** есть практика **гэмба**: менеджер должен регулярно присутствовать в месте, где идет рабочий процесс. Аналитикам это тоже полезно.

Слабо структурированная деятельность

- Есть довольно много примеров бизнес-функций, для которых сложно построить описание в виде процесса, при том что вовлечено много людей:
 - выверка бухгалтерской отчётности с решением обнаруженных проблем;
 - планирование расписания при недостатке ресурсов с переговорами;
 - поиск баланса между текущей работой, ремонтом (техдолг) и развитием;
 - а также обработка исключений и особых случаев.
- Не надо пытаться описать подобное языком бизнес-процессов!
- Альтернативой может быть координация по целям: критерии продвижения к цели и её достижения.



Это верно для любого метода: надо уметь видеть границы,
и не пытаться применять его не по назначению.

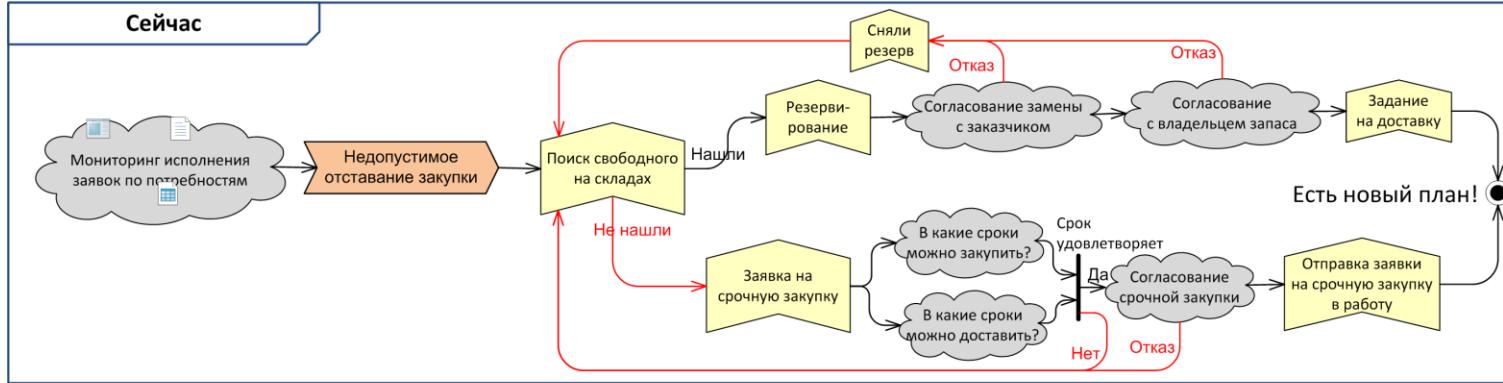
Апологеты методов часто выступают за повсеместное применение.

Описание бизнеса – схема бизнес-процессов?

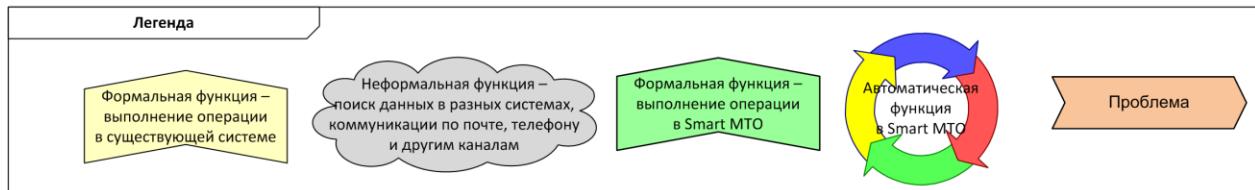
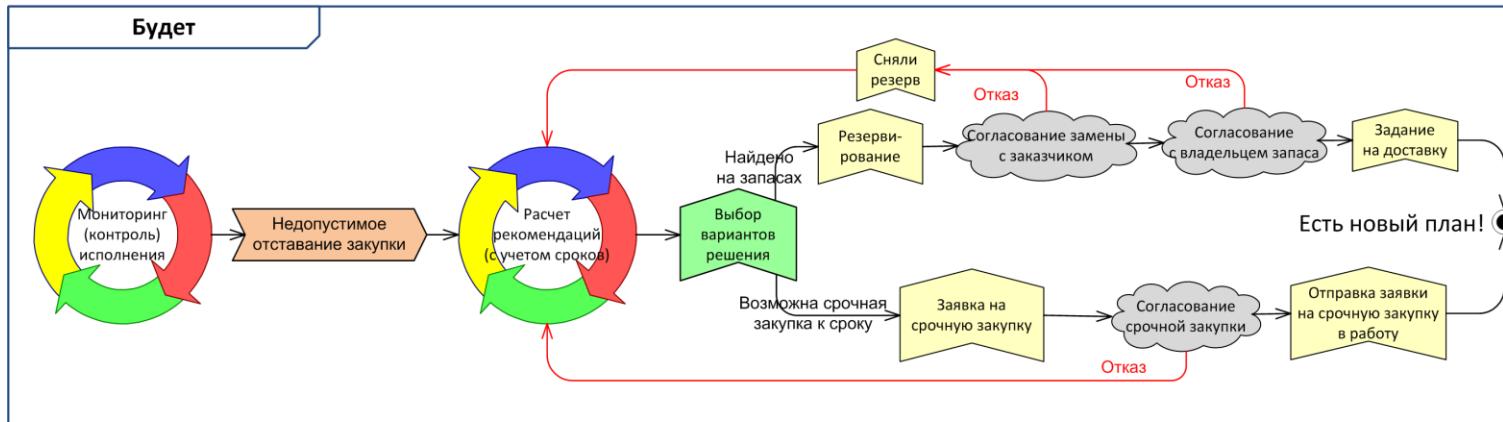
- Схемы процессов описывают бизнес-архитектуру лишь частично:
 - на них нет целей бизнеса и цепочек ценности;
 - они не показывают сопряжение бизнеса и софта;
 - далеко не всю деятельность можно описать языком бизнес-процессов.
- Они быстро устаревают, жизнь меняется быстрее:
 - схемы слабо соответствуют реальным процессам в компаниях;
 - описания конфликтуют с инструкциями пользователей по работе с системами;
 - об этом я говорил ещё в 2011 году в докладе [«Описание бизнес-процессов — waste?»](#).
- Есть много разных способов описания бизнес-архитектуры, надо представлять спектр вариантов и выбирать подходящие:
 - не стоит использовать описания бизнес-процессов лишь потому, что «так принято»;
 - для описания архитектуры ИТ-компании часто используют task flow, это альтернатива;
 - но если основа автоматизации — BPMN-движок, то бизнес-процессы надо использовать.

Smart MTO – изменения процесса

Процесс мониторинга исполнения и перепланирования при проблемах



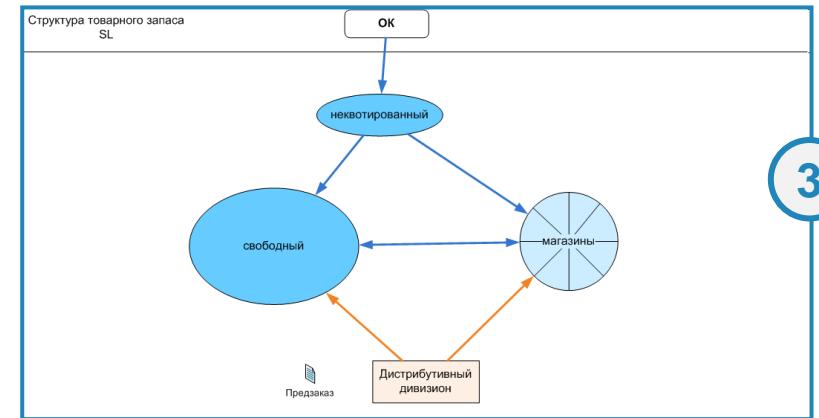
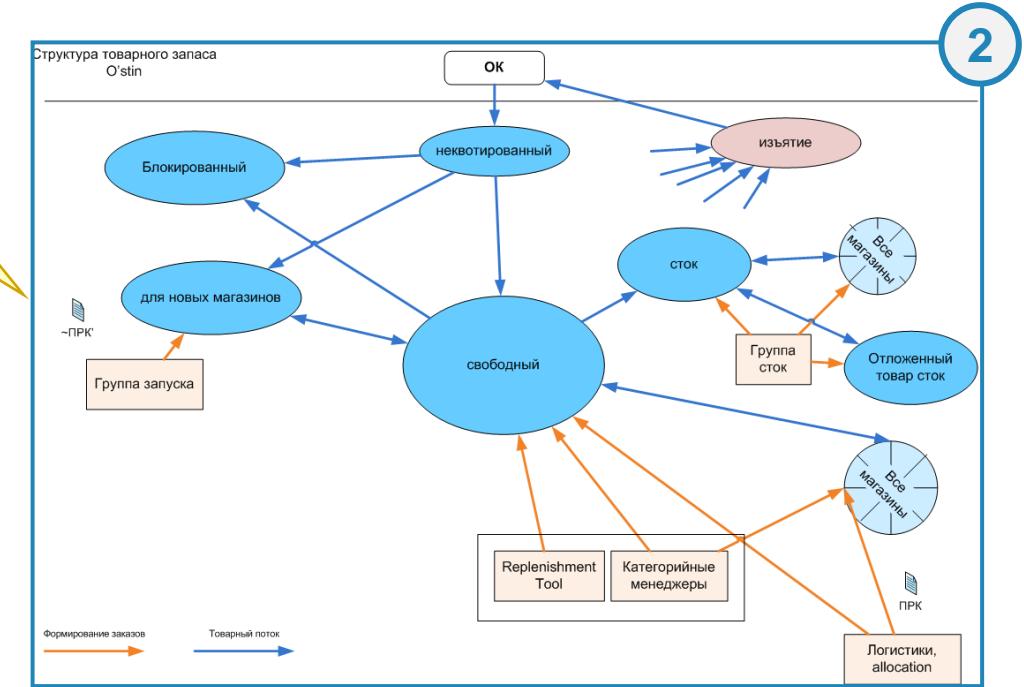
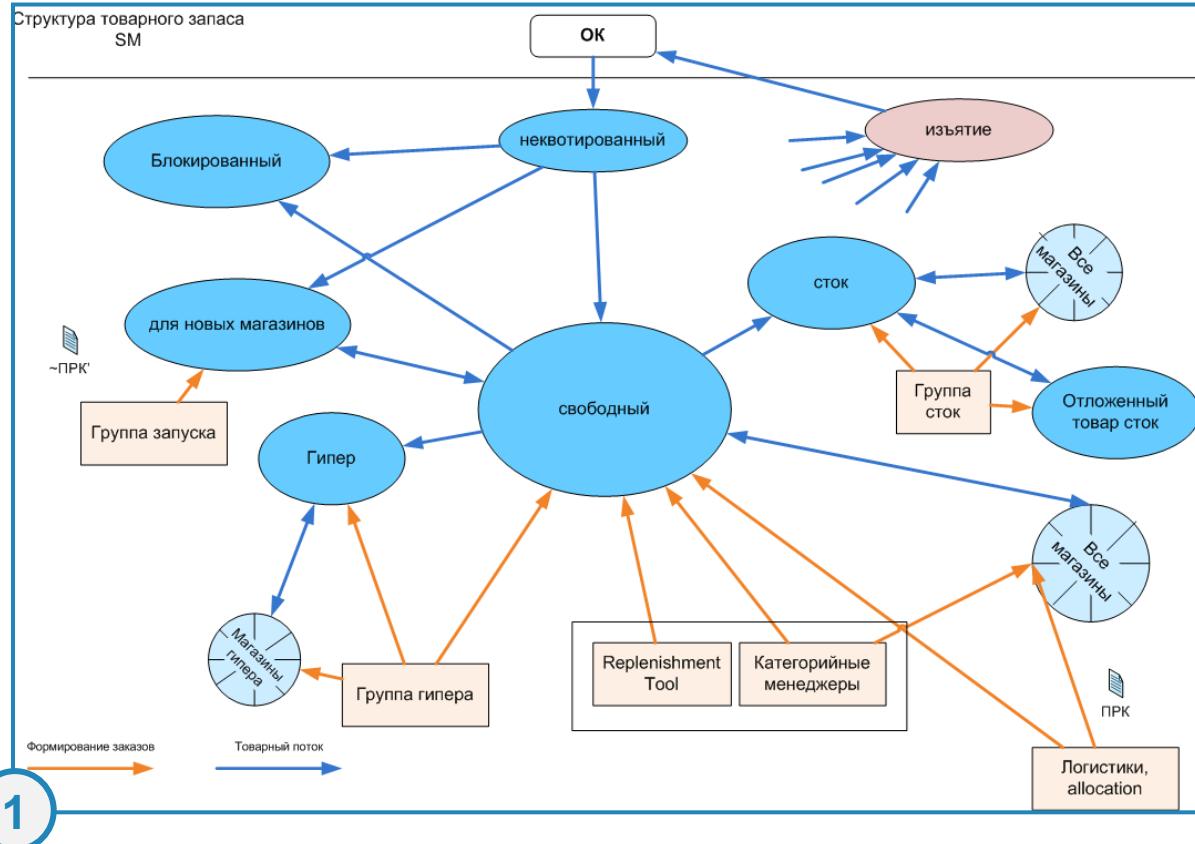
Чтобы схема ясно показала изменения, значки должны различаться. Не бойтесь отступить от формальной нотации, ищите наглядные представления!



Визуальный образ описывает назначение

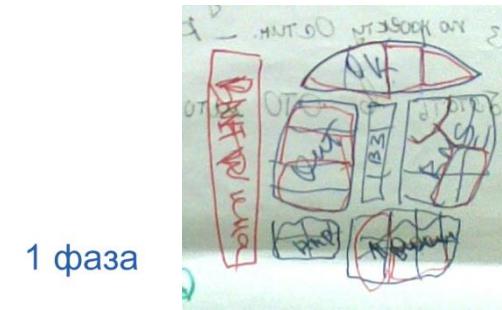
Пример:
деление товара

Варианты процессов
в простой нотации,
схемы понимаются на лету

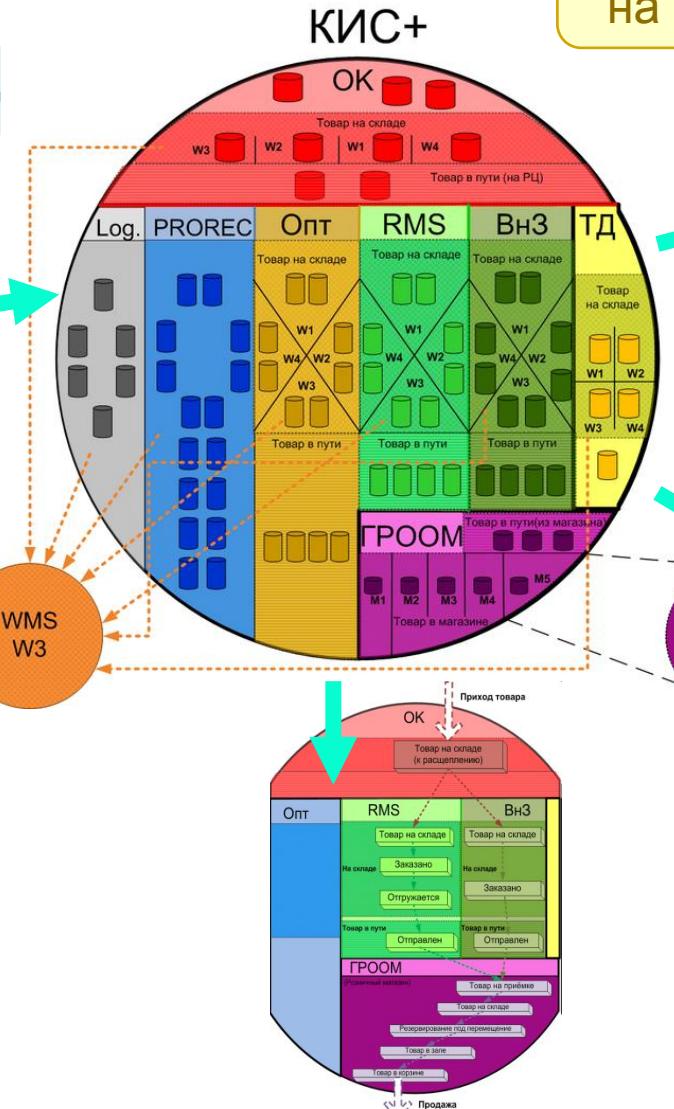
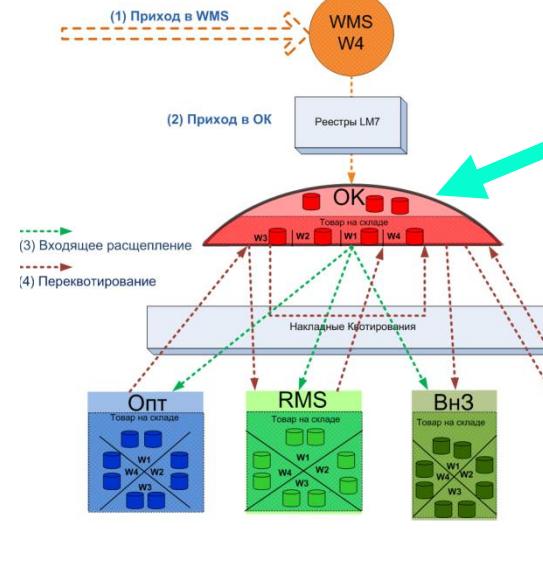


Оригинальный образ для отражения систем

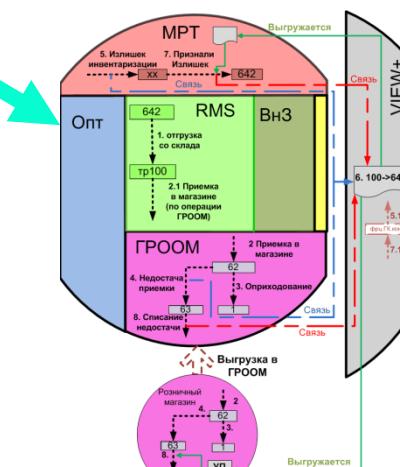
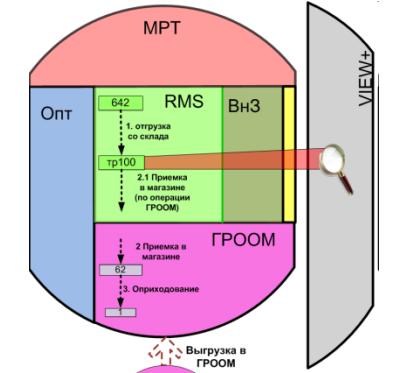
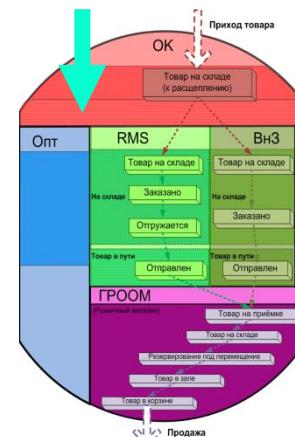
Пример: единая витрина
для многих систем



1 фаза

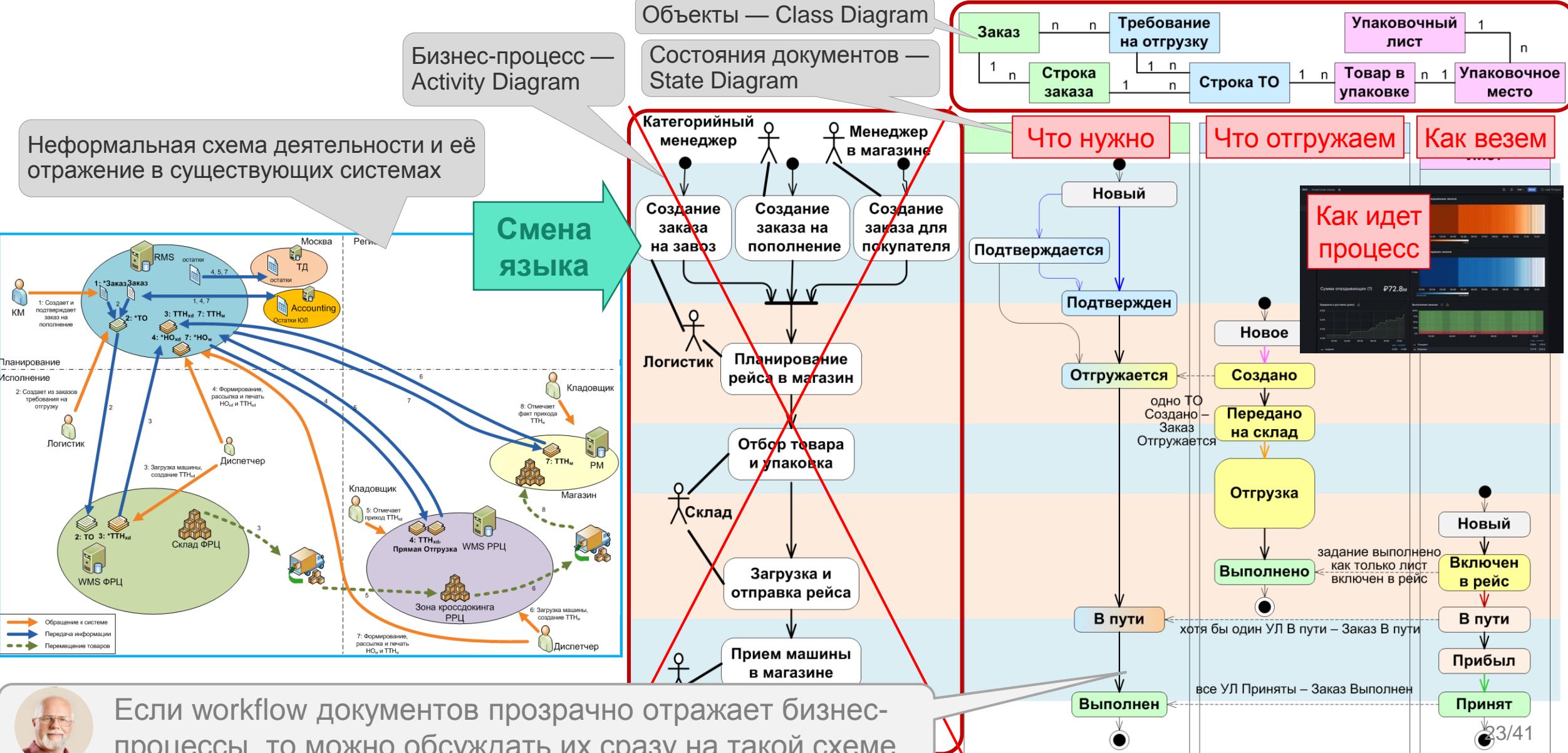


Лупа: смотрим
на сложную структуру



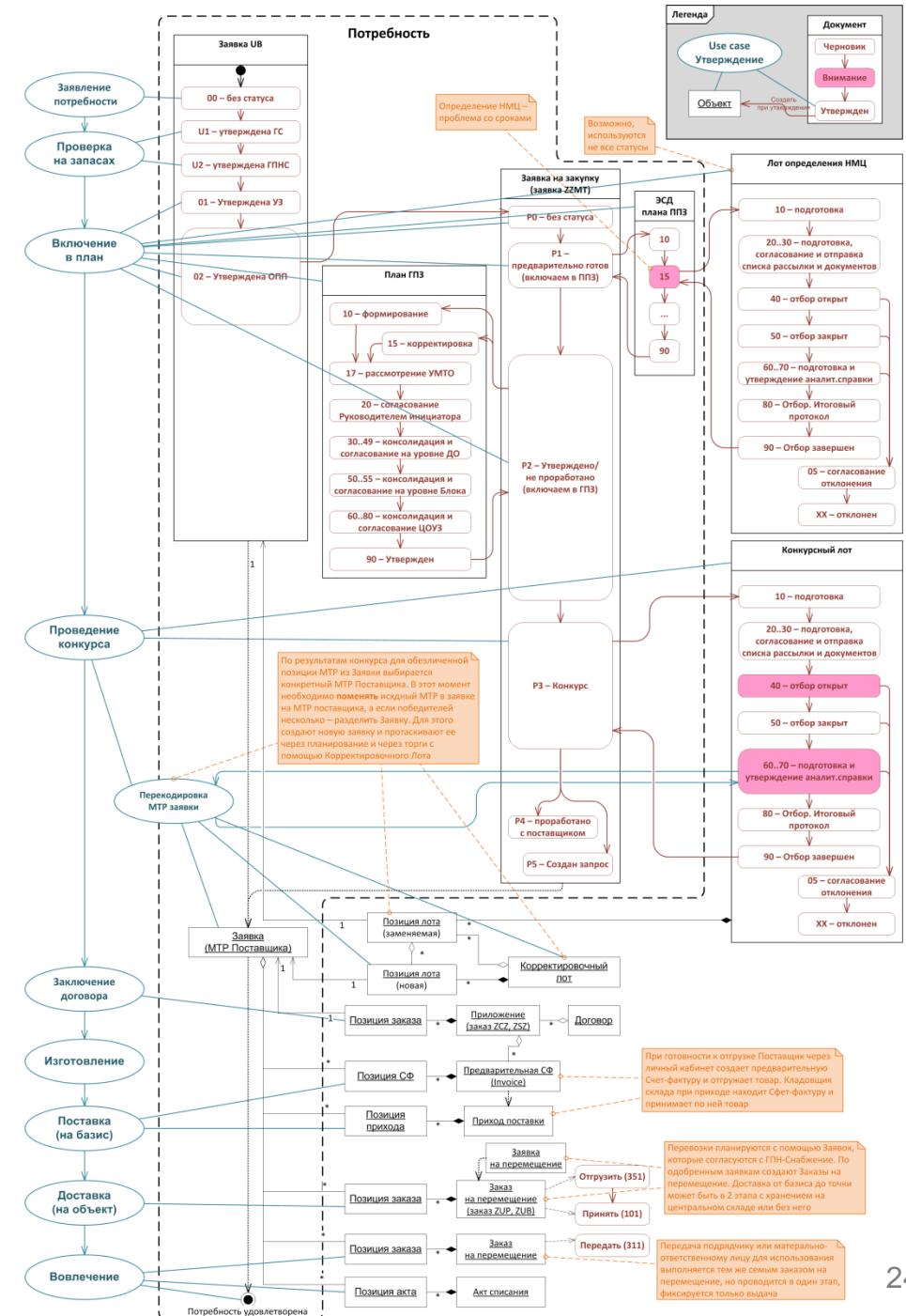
**Workflow документов –
отражение бизнеса**

Формализация снабжения магазинов

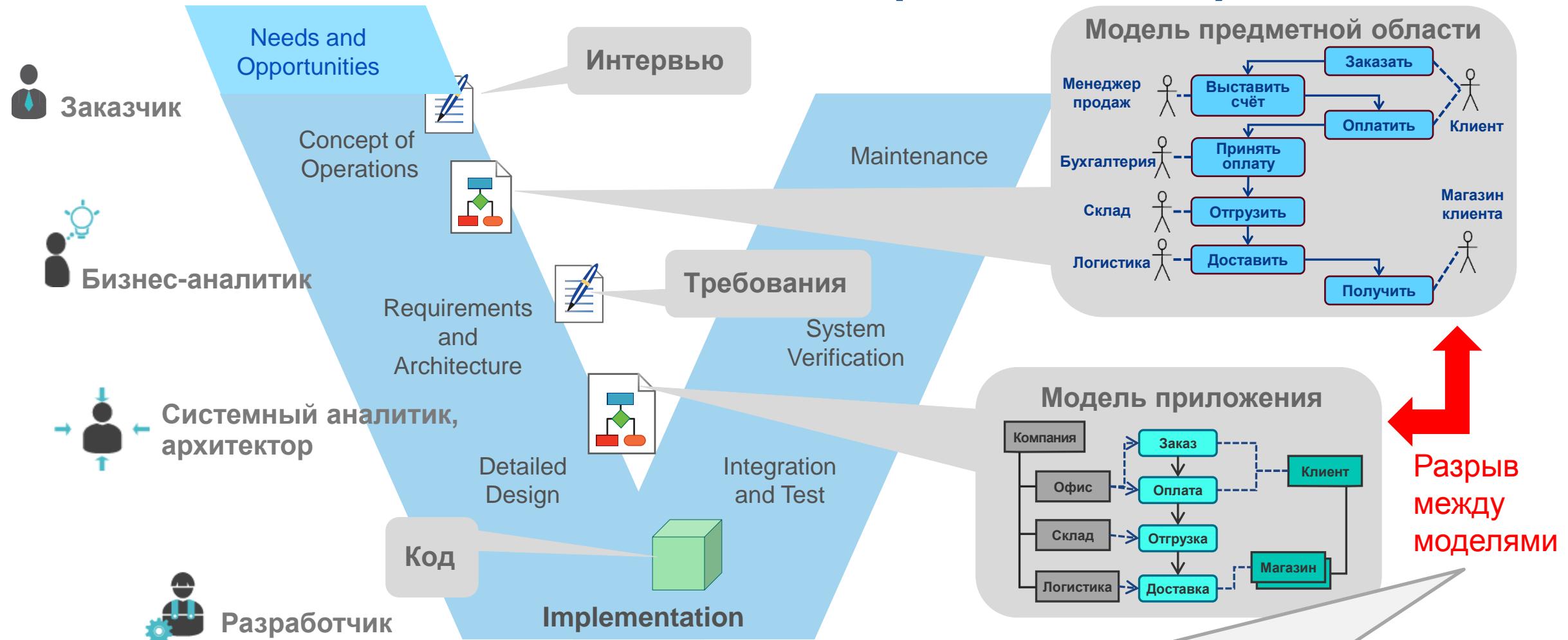


Исторические наследия реального workflow

- Бизнес-процесс усложнялся, для поддержки в системе приспосабливали разные документы
 - Точки внимания бизнеса оказывались глубоко «закопаны» в workflow
 - Часть workflow реализована множеством не связанных документов

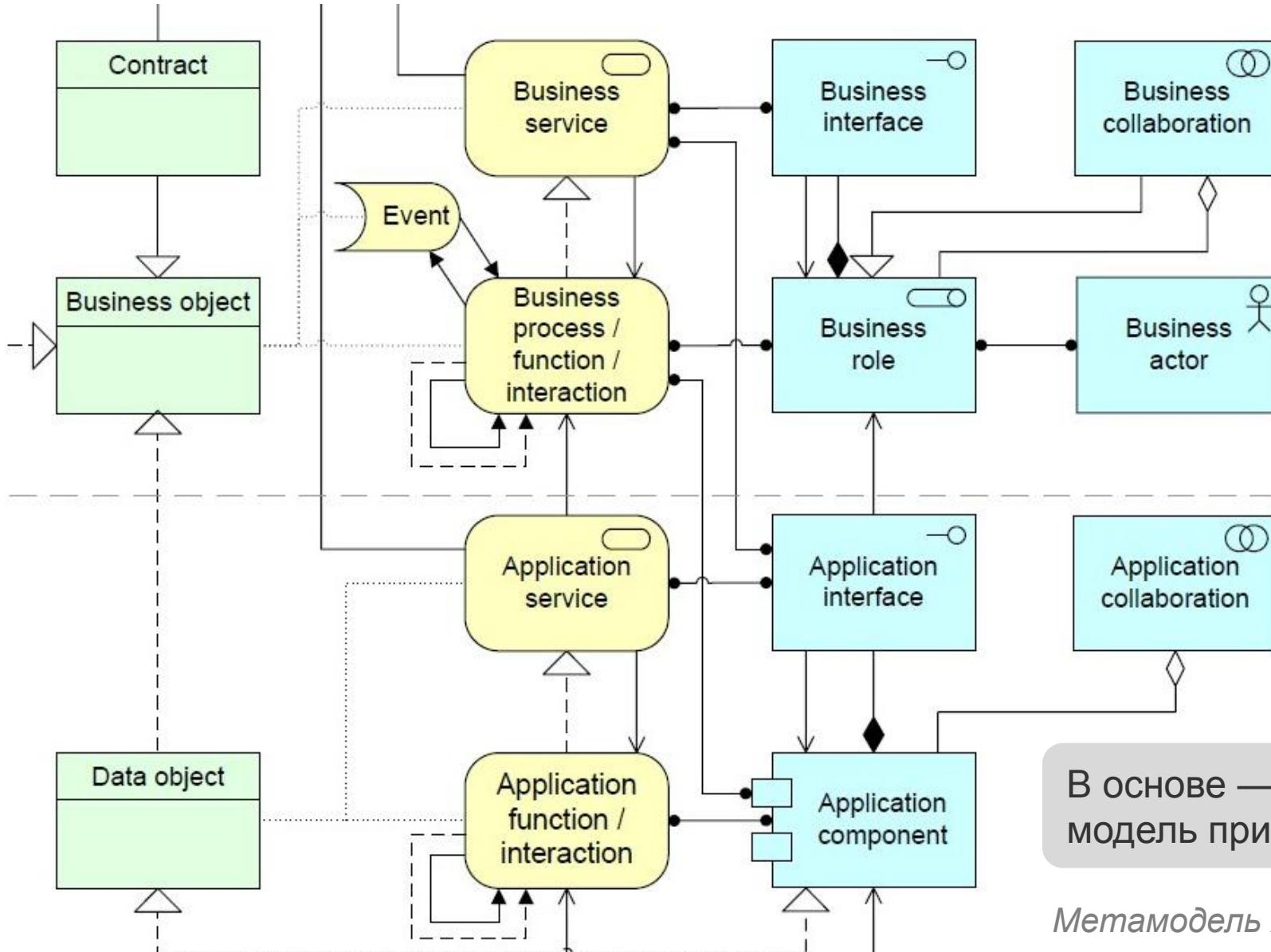


Поддержка артефактами (классика)



Проблема: каждый уровень отдельно –
сложно изменять и поддерживать соответствие

Archimate – гибкая связь бизнеса и софта

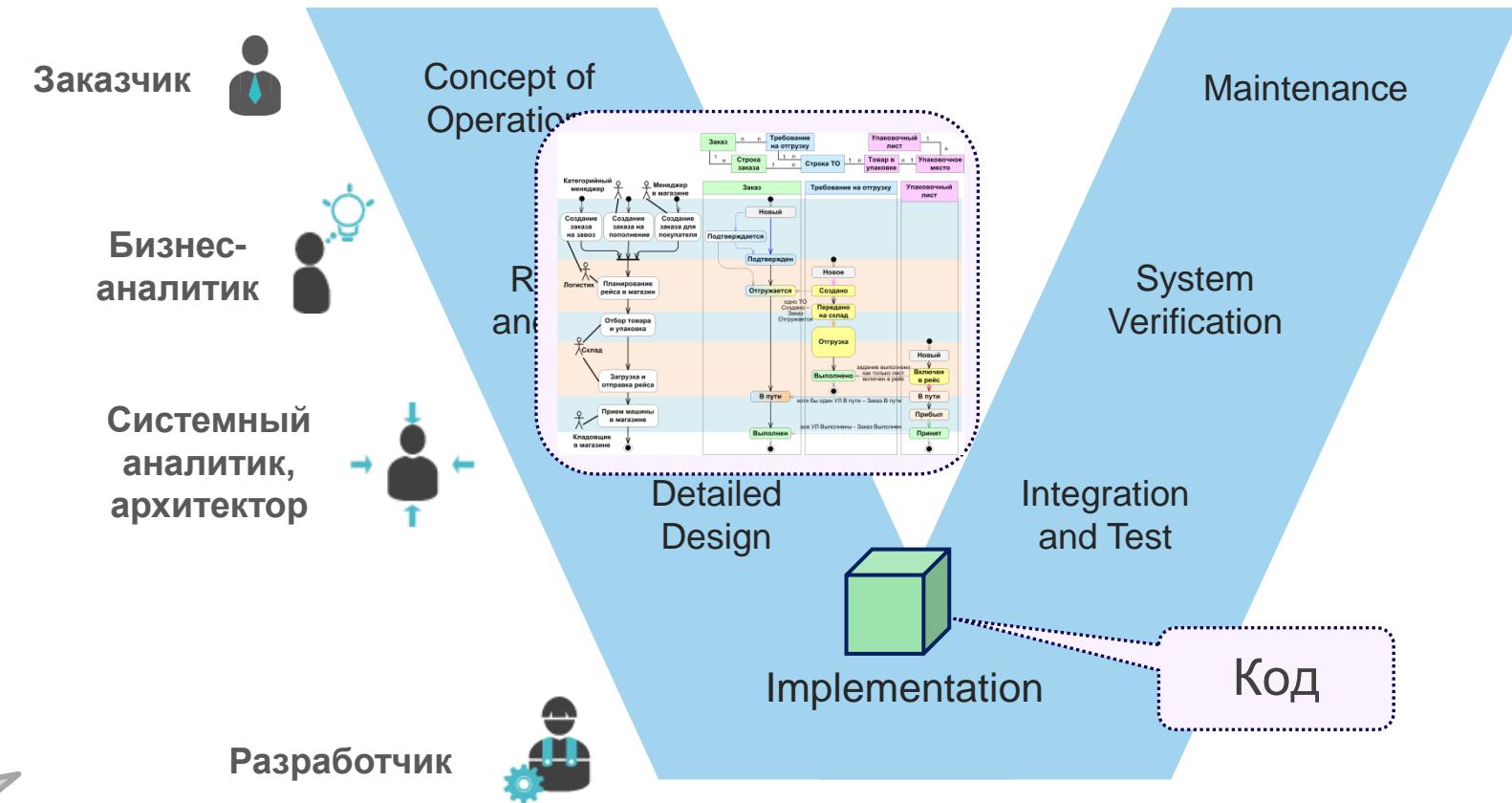


В основе — сервисная модель приложения

Гибко не получается

Domain Driven Design

- Единый язык:
 - на основе терминов предметной области,
 - понятен всем участникам проекта;
- Единая модель, приложения и его встройки в бизнес;
- Прозрачное отражение модели в код.



У меня есть много докладов о DDD, последний —
[«DDD: модели вместо требований 9 лет спустя \(ЛАФ-2023\)»](#)

ТЗ — способ сменить ответственность



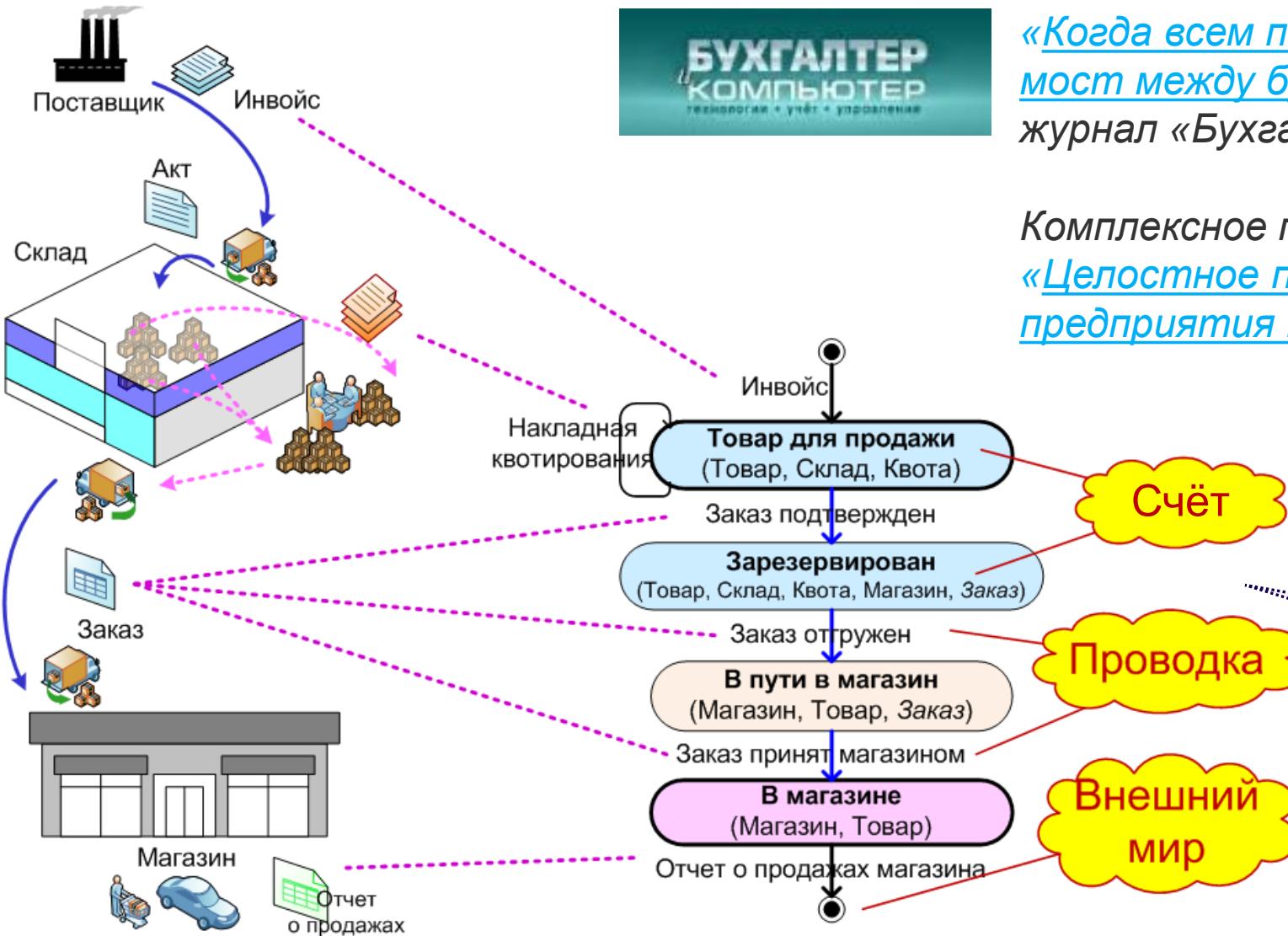
- Ответственность за достижение целей проекта вынесена за пределы ИТ-команды;
- Как следствие, цели часто не достигаются;
- Выделение этапа ТЗ не решает проблему — ТЗ утверждают, а гарантий оно не даёт.



Нужны ли вам требования к системе?
Зависит от границы проекта и контракта с заказчиком.

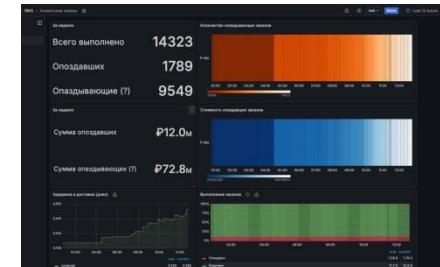
Пара слов об учете

Учёт: бизнес как поток ресурсов



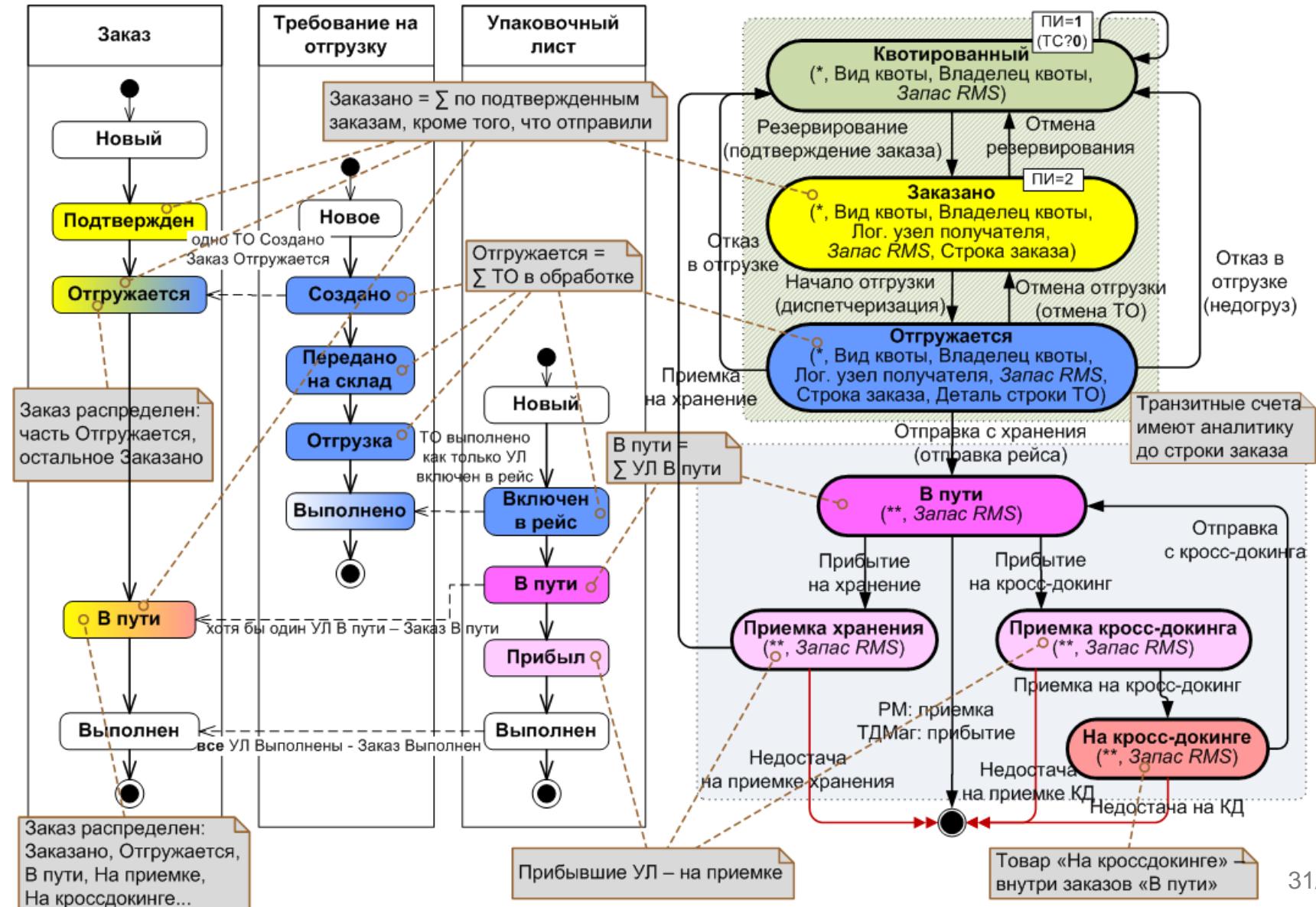
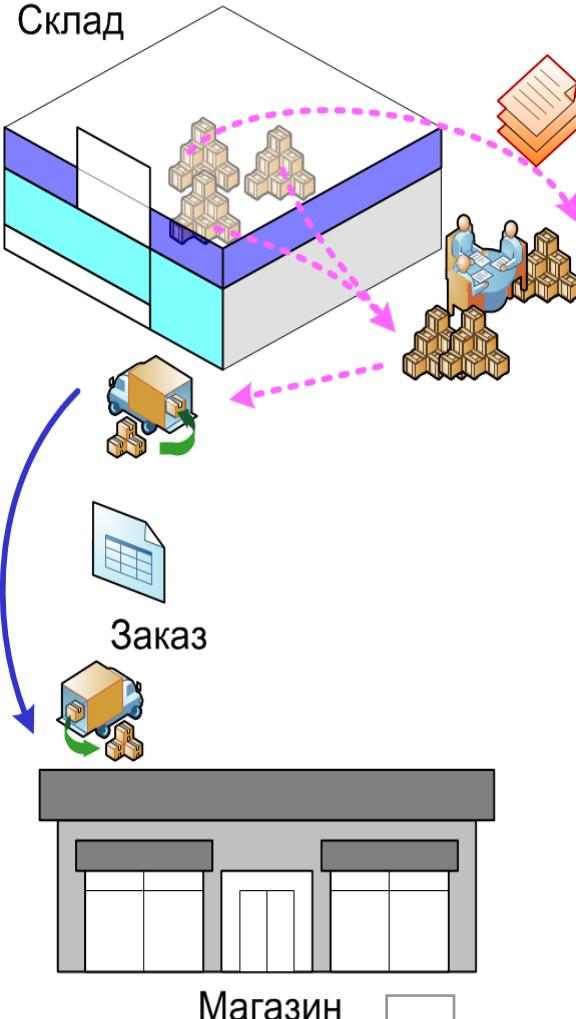
«Когда всем понятно. Диаграммы учёта: мост между бухгалтером и разработчиком» — журнал «Бухгалтер и компьютер», №5-2011.

Комплексное представление — мой доклад
«Целостное представление деятельности предприятия на диаграммах учёта».



Движение товаров и других ресурсов отражается как проводки между счетами

Учет меняется через workflow документов



Разбираем сложный ИТ-ландшафт

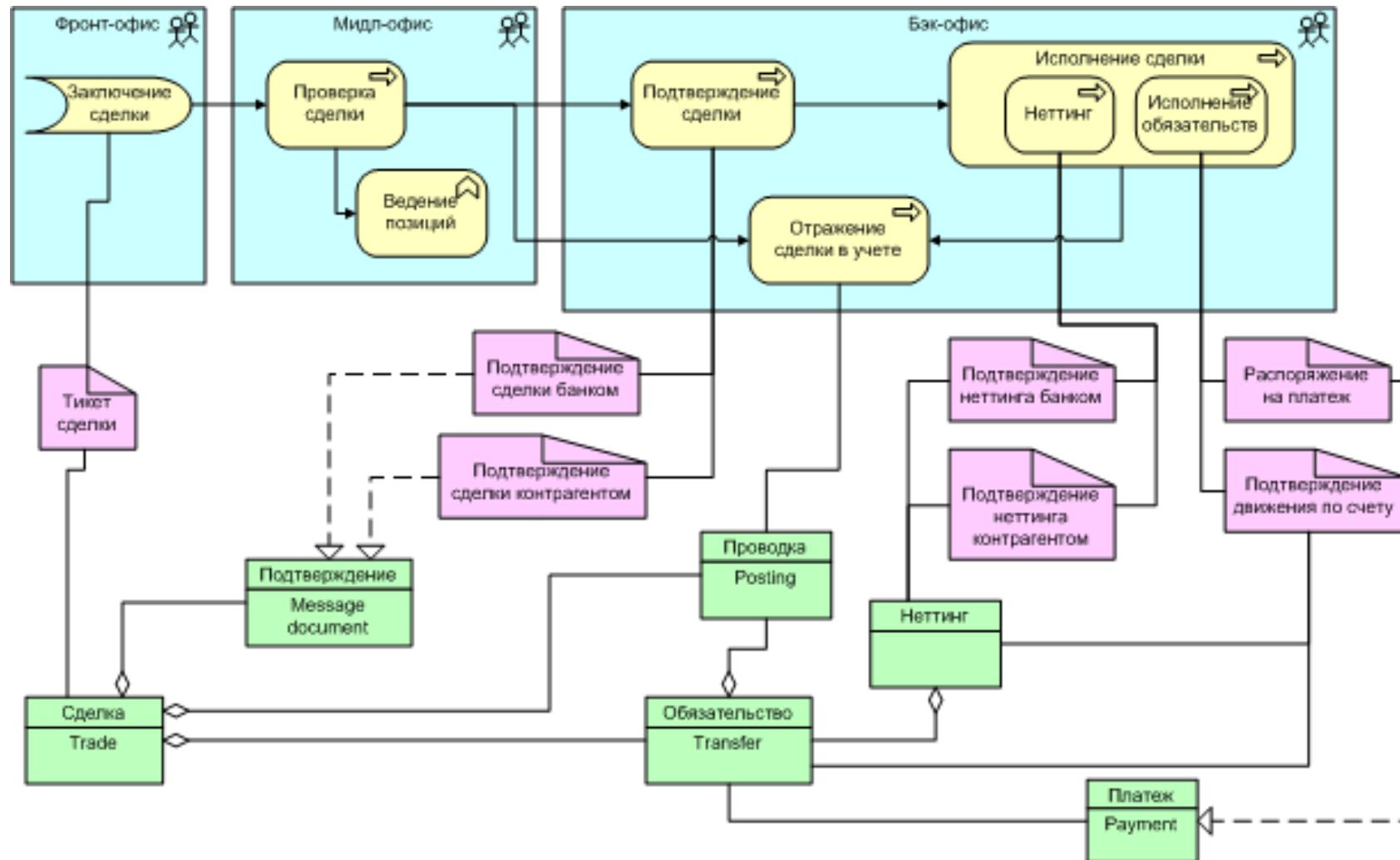


Схемы должны показывать связь
софта и бизнеса в единую систему

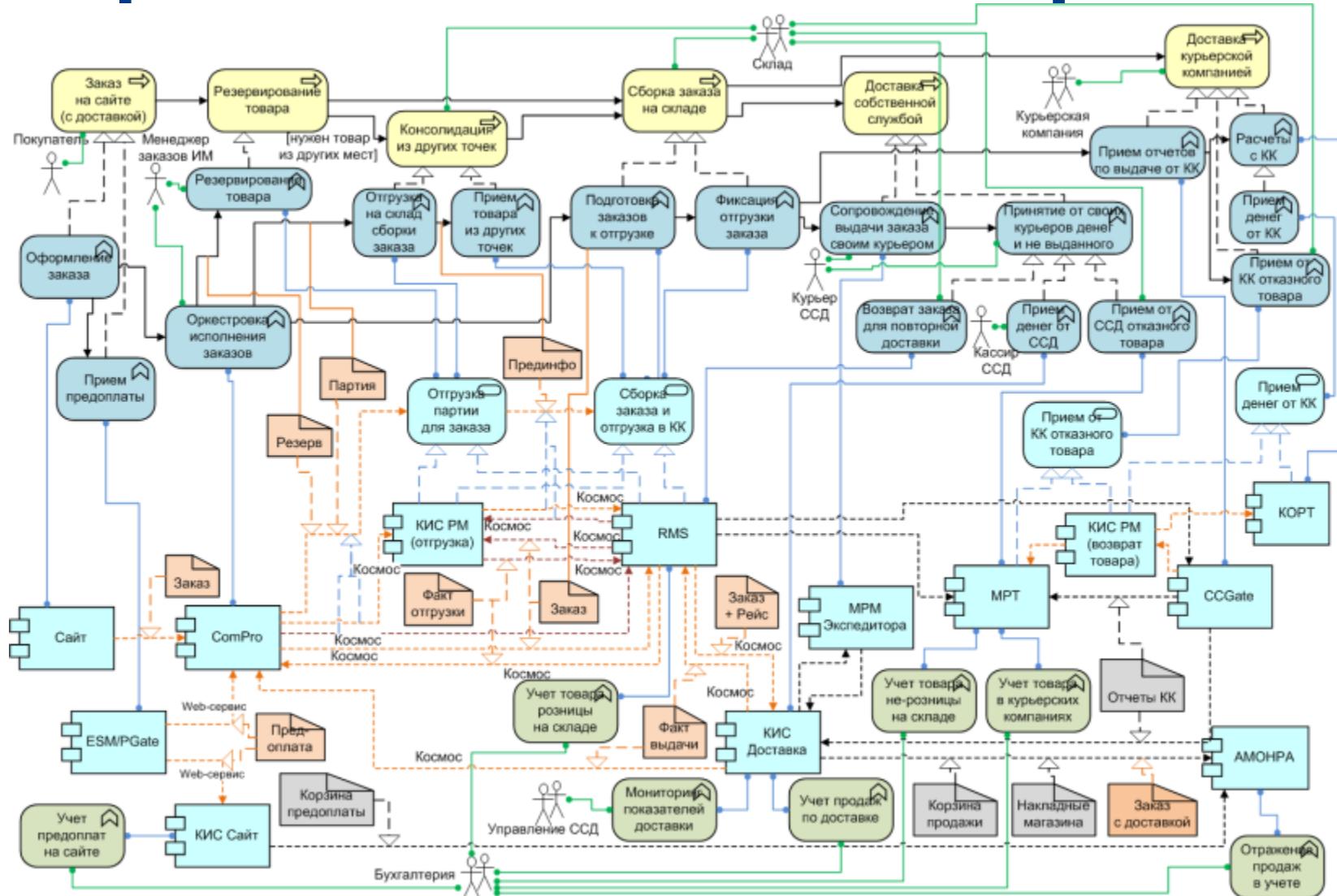
Архитектура снабжения: разбираем на функции



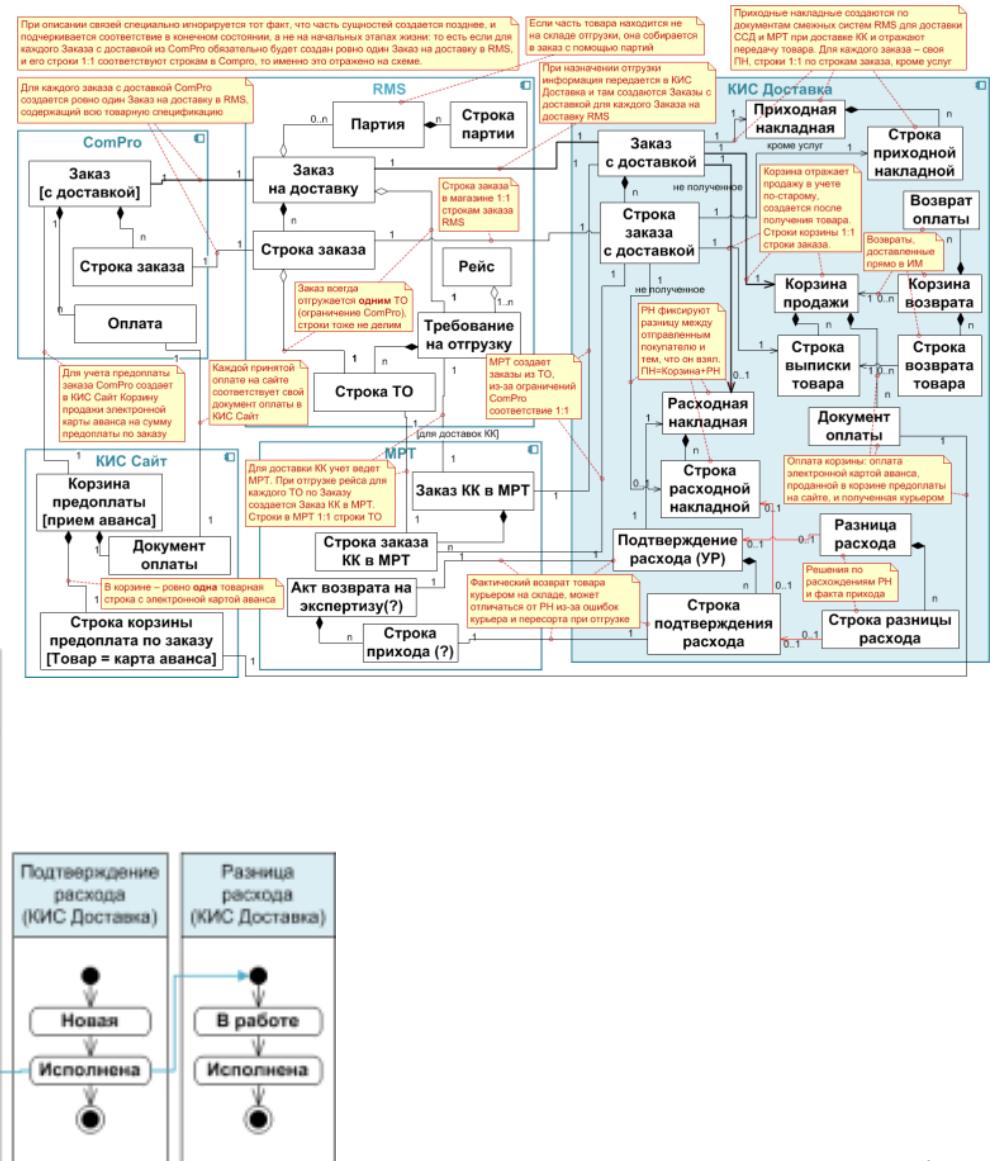
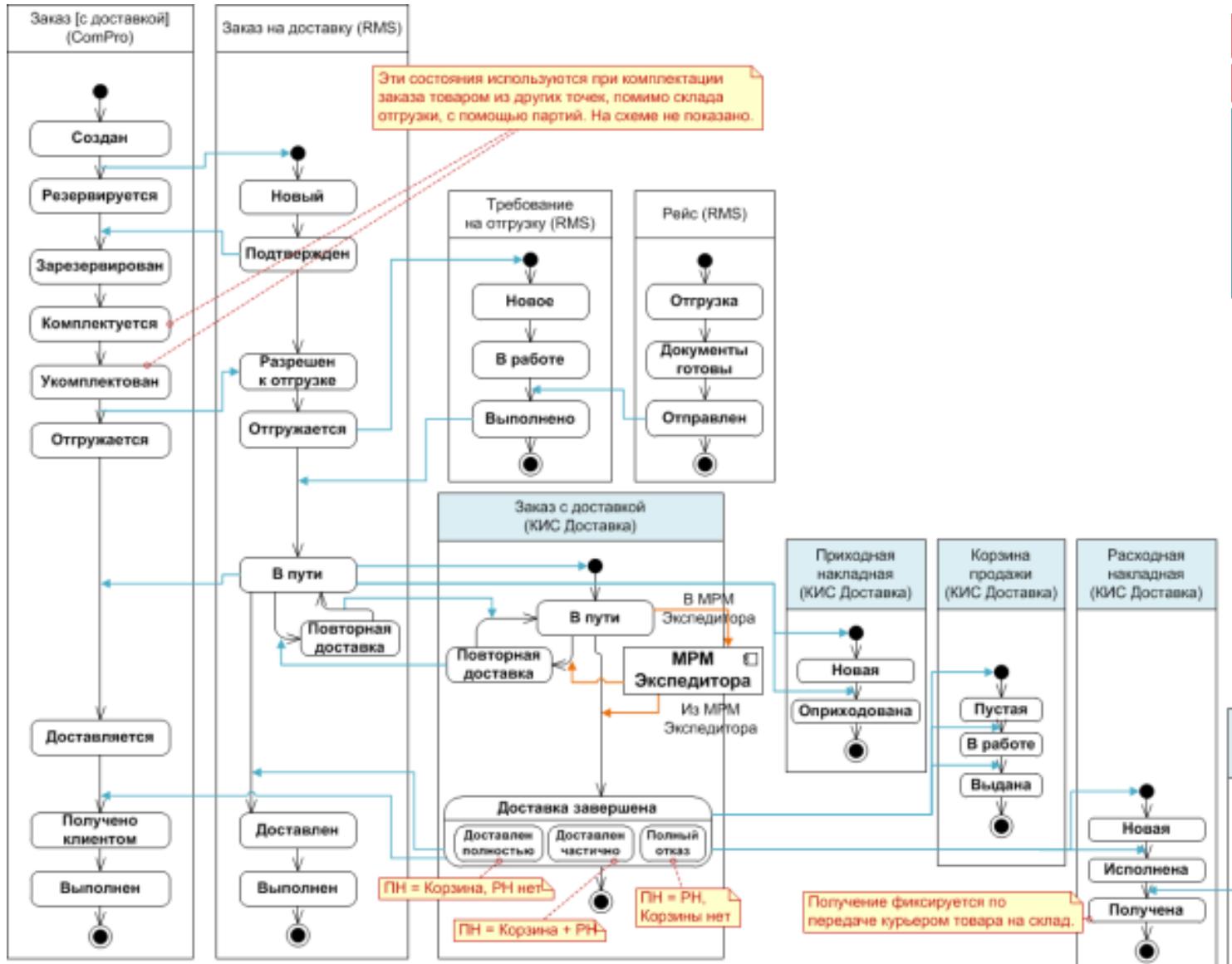
Реализация процесса через объекты



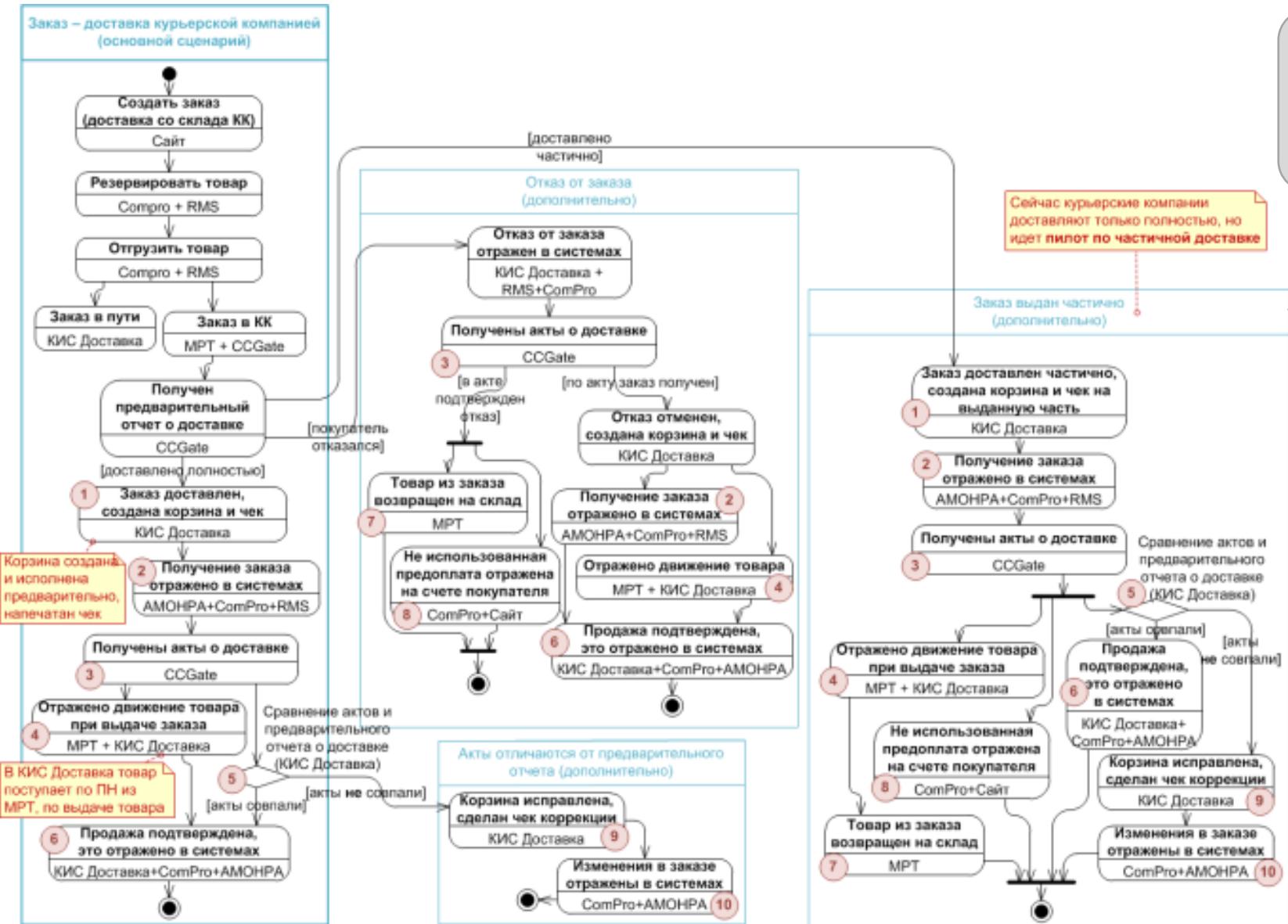
Реализация процесса в компонентах: обработка заказа в интернет-магазине



Реализация процесса через workflow

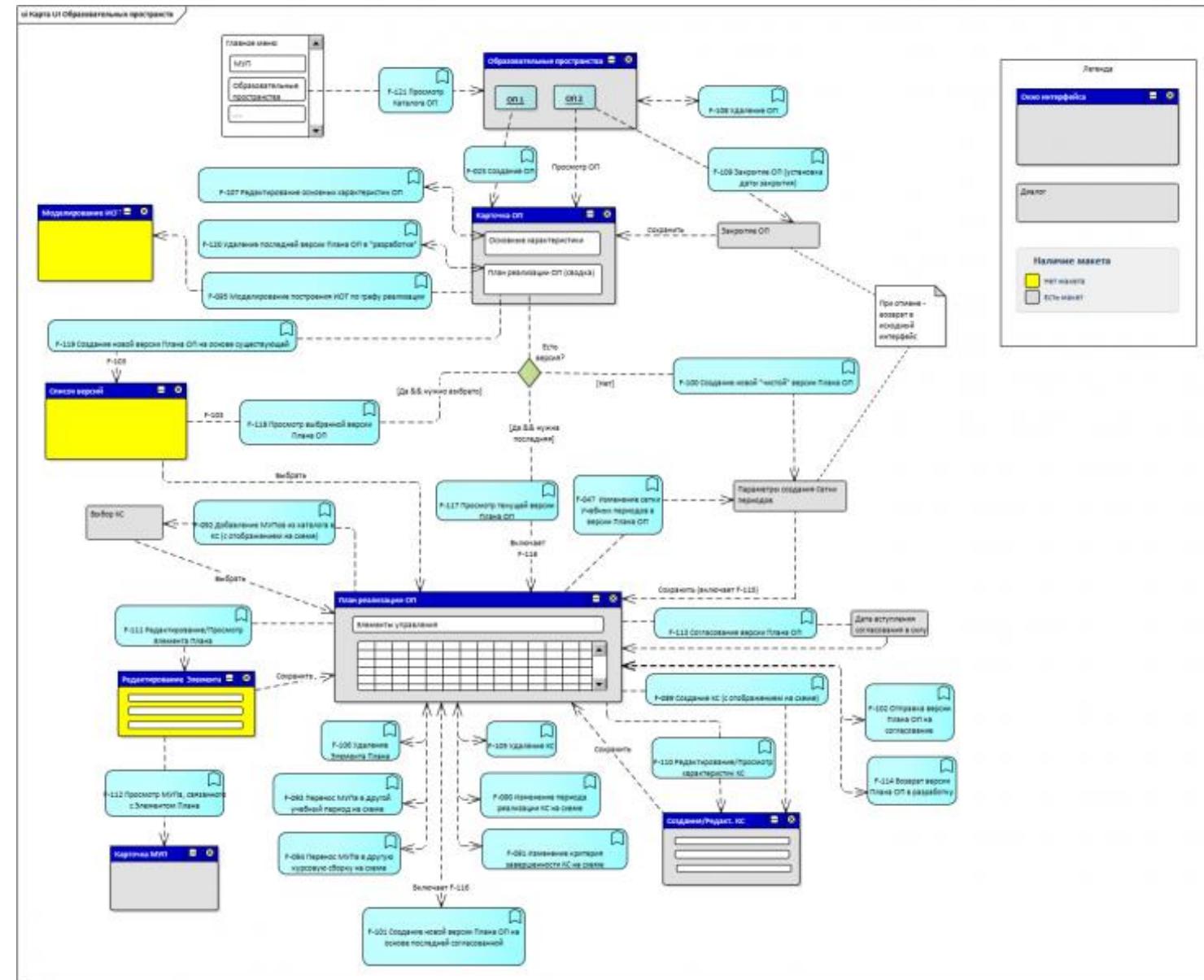


Сценарии демо: делаем сложное понятным

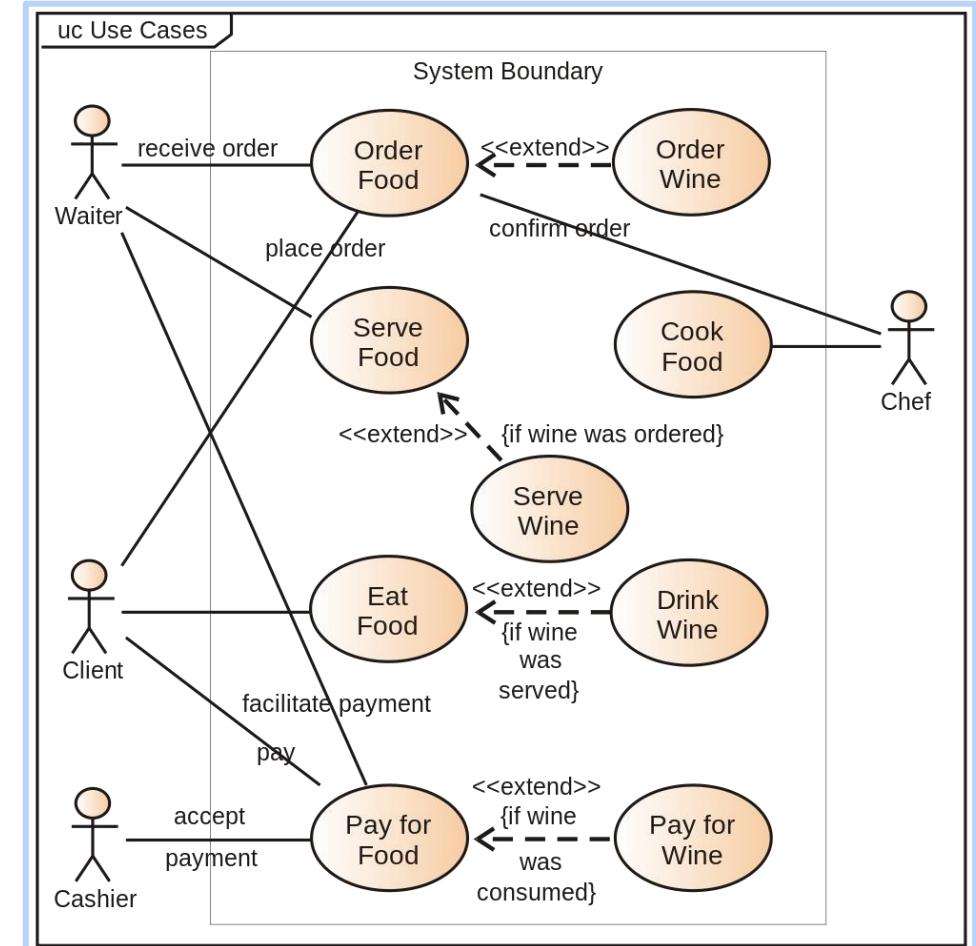


Сценарии демонстраций могут быть эффективной основой планирования разработки

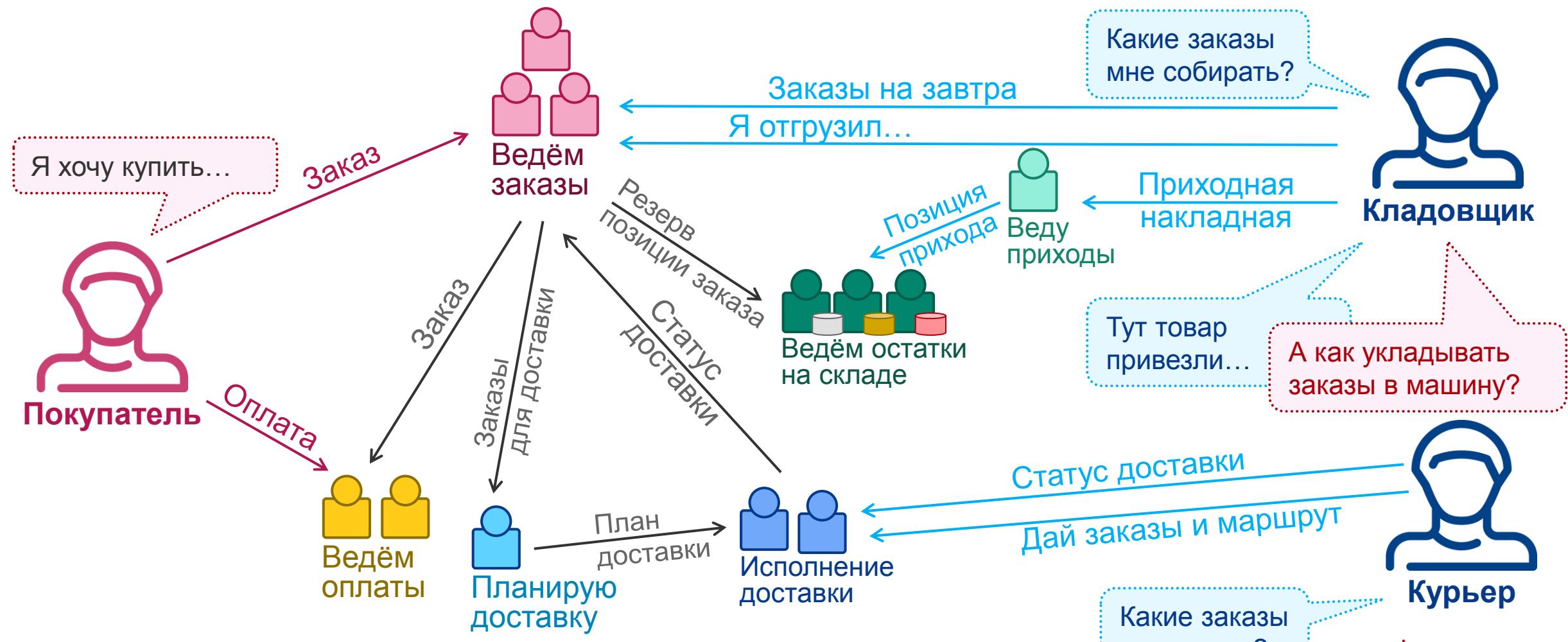
Поддержка бизнес-функций экранами приложения



Контекстная диаграмма C4 Model — система в окружении



Модель для сервисной архитектуры



Из доклада «[Визуальное проектирование масштабируемых приложений](#)

Софт и бизнес – единая система



- Страйте план действий и выбирайте методы под ситуацию
- Изучайте бизнес, а не схемы бизнес-процессов
 - Слушайте пользователей заказчика, они ценный источник знаний, а когда слышите странное – погружайтесь и разбирайтесь
 - Просите показать работу в легаси-системе, **сопоставляйте рассказ** со всеми инструкциями и выгрузками реальных документов
- **Проявляйте связь бизнеса и софта, ищите схемы,** которые хорошо это опишут и покажут важное
- Проектируя новое – убирайте исторические наслоения

Вопросы по докладу
и обратная связь



Максим Цепков



<http://mtsepkov.org>



[@MaximTsepkov](https://t.me/MaximTsepkov)

На сайте много материалов по [анализу](#)
[архитектуре](#), [Agile](#) и [менеджменту](#)
[самоуправления](#), [моделям soft skill](#),
мои [доклады](#), [статьи](#) и [конспекты книг](#)



Вакансии

Пишите на hr@custis.ru,
подходите с вопросами!