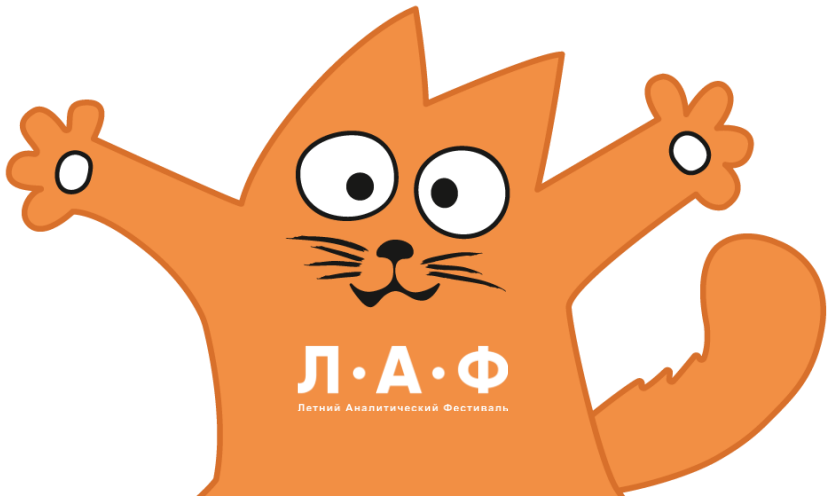
Сначала проект, потом анализ: прошлое возникает из будущего

Максим Цепков
Главный архитектор решений CUSTIS
Навигатор по миру Agile, бирюзовых организаций и спиральной динамики

http://mtsepkov.org



7-8 июня 2025

Кострома

Мой опыт проектирования больших систем

1997 АБС Банка: система была разработана и запущена в эксплуатацию за полгода, в 2000-2001 следующая версия АБС была установлена в других банках

2003 Старт разработки серии систем для группы компаний Спортмастер: каталог товаров (1М+ позиций с ценами), управление товарным запасом и логистика, розничный магазин, оптовая торговля, финансовая отчетность. 1000+ точек установки, развивается до сих пор

2008 Главная книга для Газпромбанка, позднее – еще несколько систем, интеграция с сложный ИТ-ландшафт банка, системы продолжают развиваться

2020 Системы управления снабжением для нескольких крупных заказчиков, интеграция с окружением в сложном корпоративном ландшафте

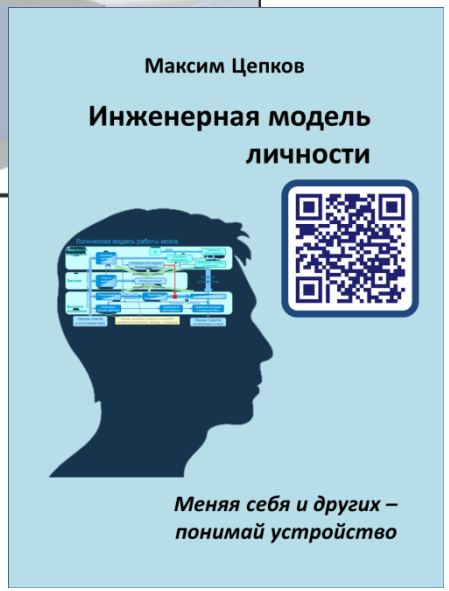
И это не считая множества других проектов

CUSTIS

Л·А·Ф Летний Аналитический Фестиваль

Проект надо воплотить: менеджмент и soft skill

- 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 IT: от инженерного подхода и PMBOK – 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?

Я расскажу свой метод работы с enterprise-проектами

- Обычно проект начинают с описания As Is, существующих бизнес-процессов
- Я действую иначе: после первых интервью строю модель будущего решения
- А план проекта строю, формируя план будущих демонстраций

Метод **эффективнее** классической схемы: система получается нацеленной на решение реальных проблем и получает быструю обратную связь

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

Проект начинается с vision!

Проект будущего фокусирует картину As Is

Настоящее не надо описывать в деталях — надо выделить то, что важно для будущих изменений

СМД-методология. Схема Шага развития (одна из рисовок)

Проект начинается с идеи

1. Начинается с идеи сделать хорошо некоторым людям с помощью софта
2. Далее надо придумать, как они будут действовать, используя софт
3. И сформулировать требования к софту, исходя из способов работы
4. Затем проектируется архитектура — принципиальное устройство софта
5. Затем — дизайн, детальное устройство софта
6. И реализация в коде

Различный уровень заказа.

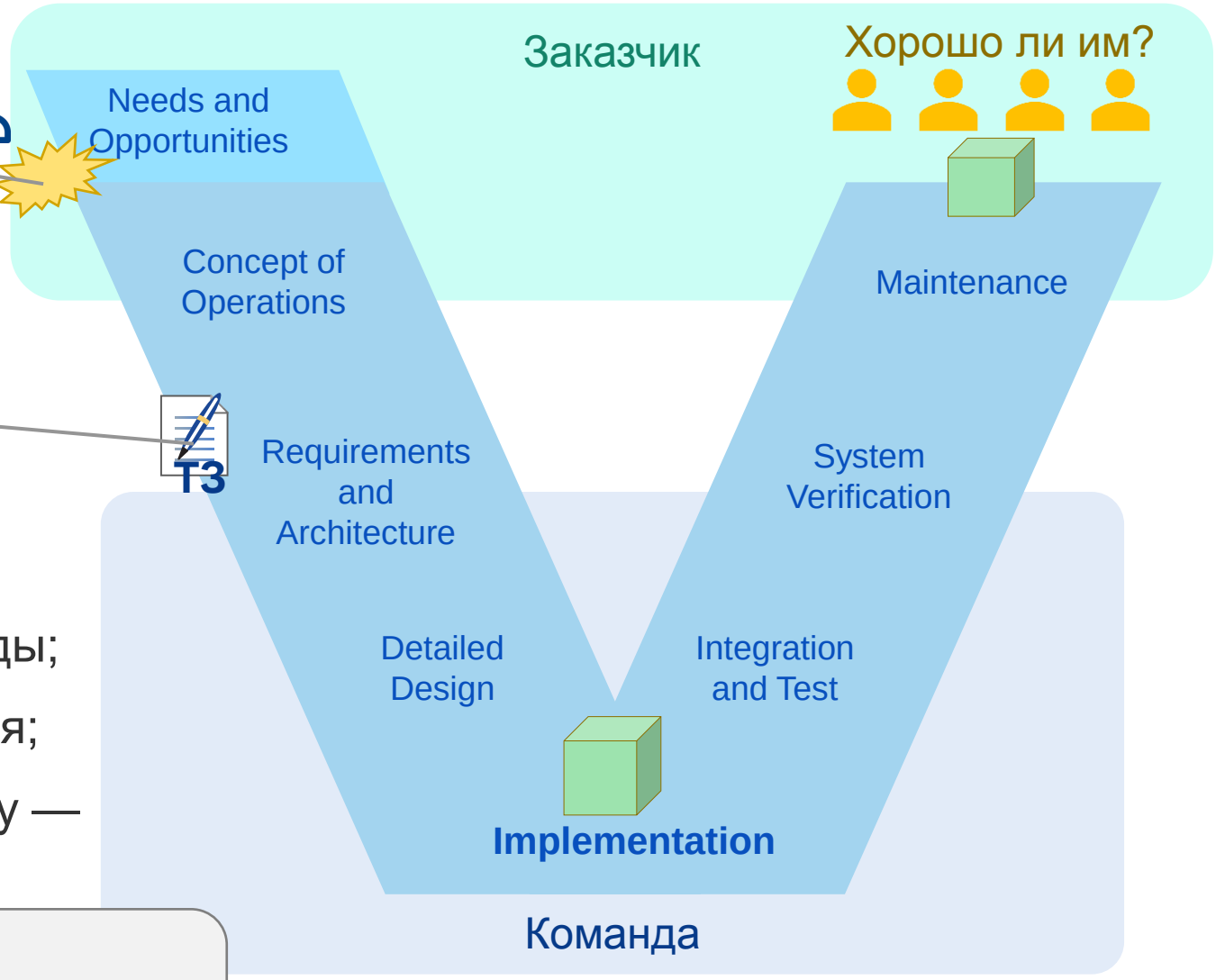
Этот путь не обязательно требует документов, его можно проходить в голове и коммуникации.

[V-Model (Wikipedia)]

ТЗ — способ сменить ответственность

Идея: сделать хорошо людям

Задача: напишите софт, удовлетворяющий требованиям



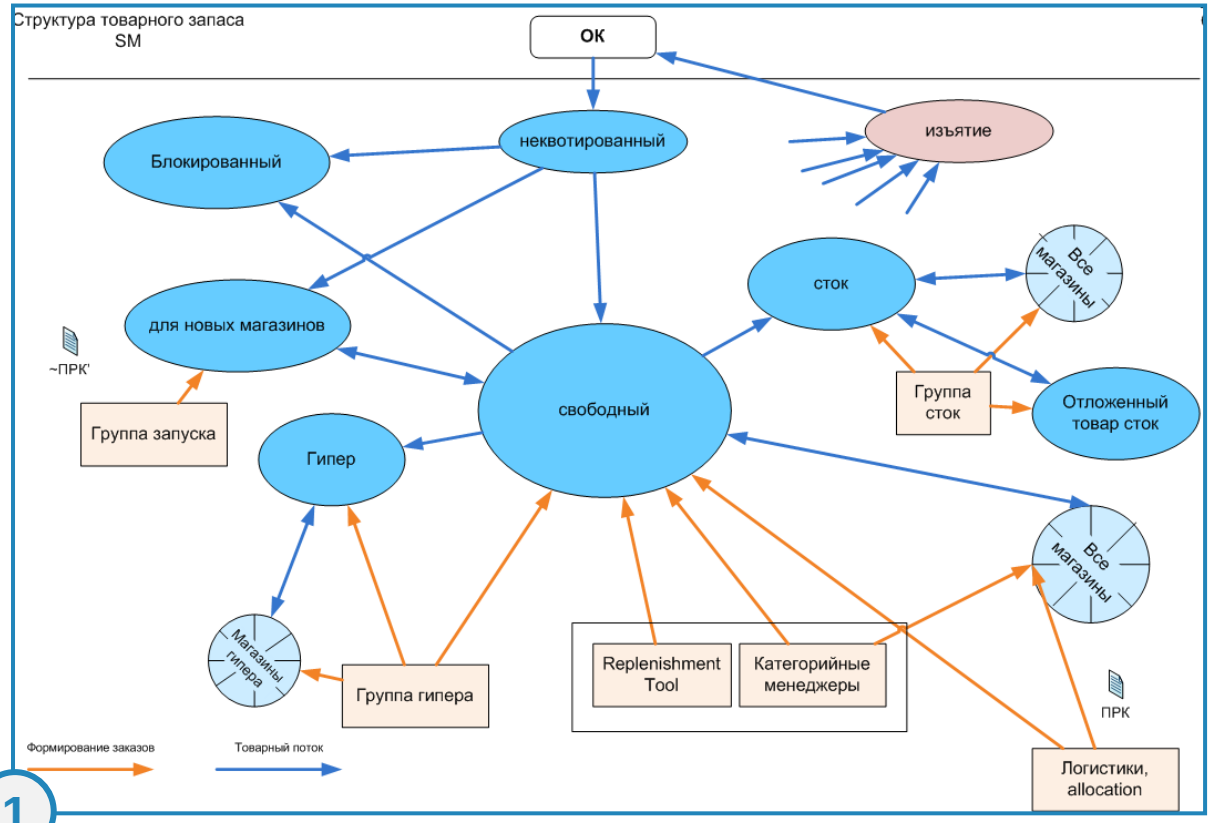
- Ответственность за достижение целей проекта вынесена за пределы ИТ-команды;
- Как следствие, цели часто не достигаются;
- Выделение этапа ТЗ не решает проблему — ТЗ утверждают, а гарантий оно не даёт.

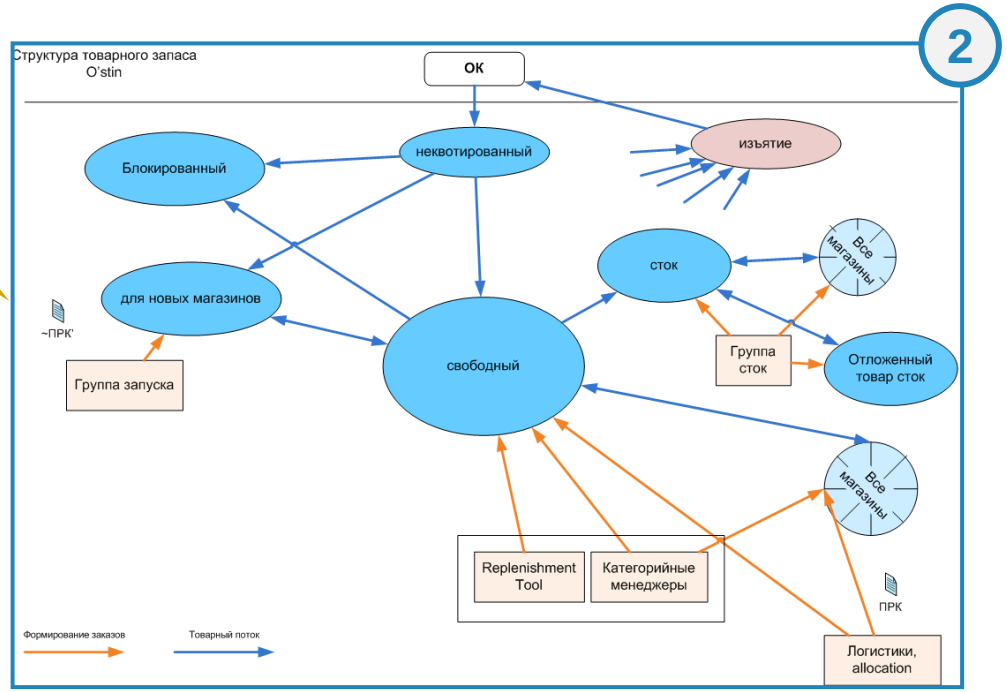
Нужны ли вам требования к системе?
Зависит от границы проекта и контракта с заказчиком.

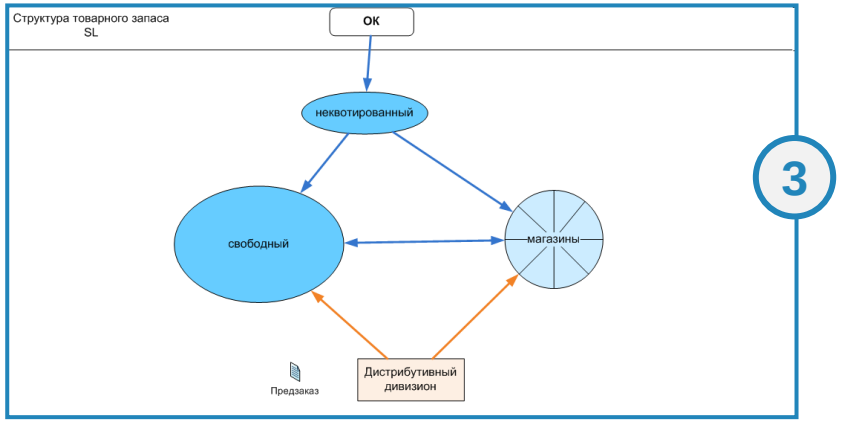
Концепт системы квотирования товара

Пример:
деление товара

Варианты процессов
в простой нотации,
схемы понимаются на лету



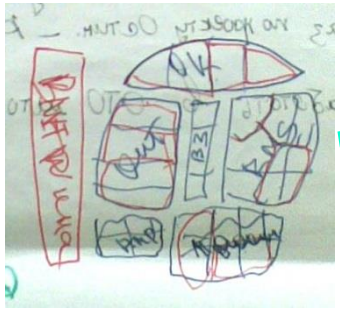




Концепт управления товарным запасом многими системами

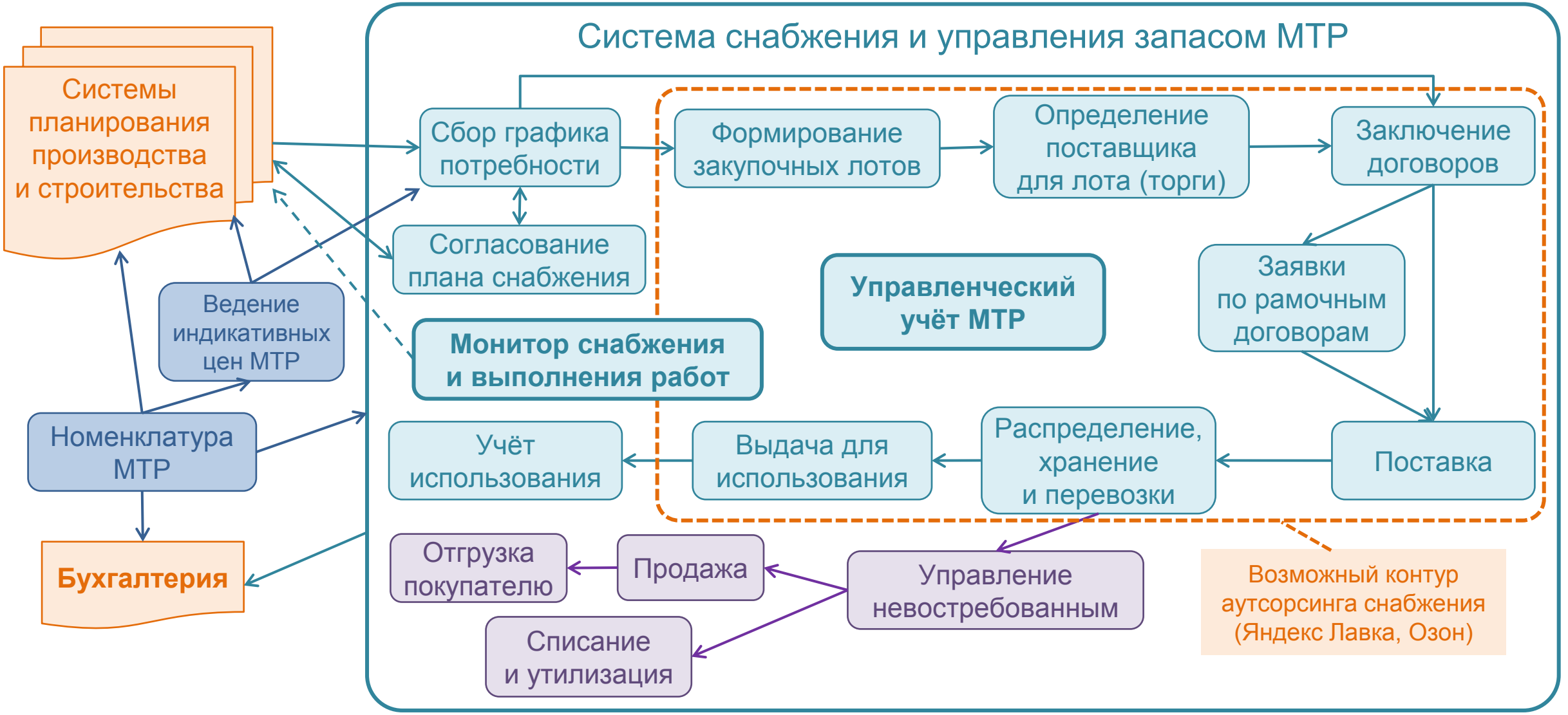
Лупа: смотрим на сложную структуру

1 фаза



Л·А·Ф
Летний Аналитический Фестиваль

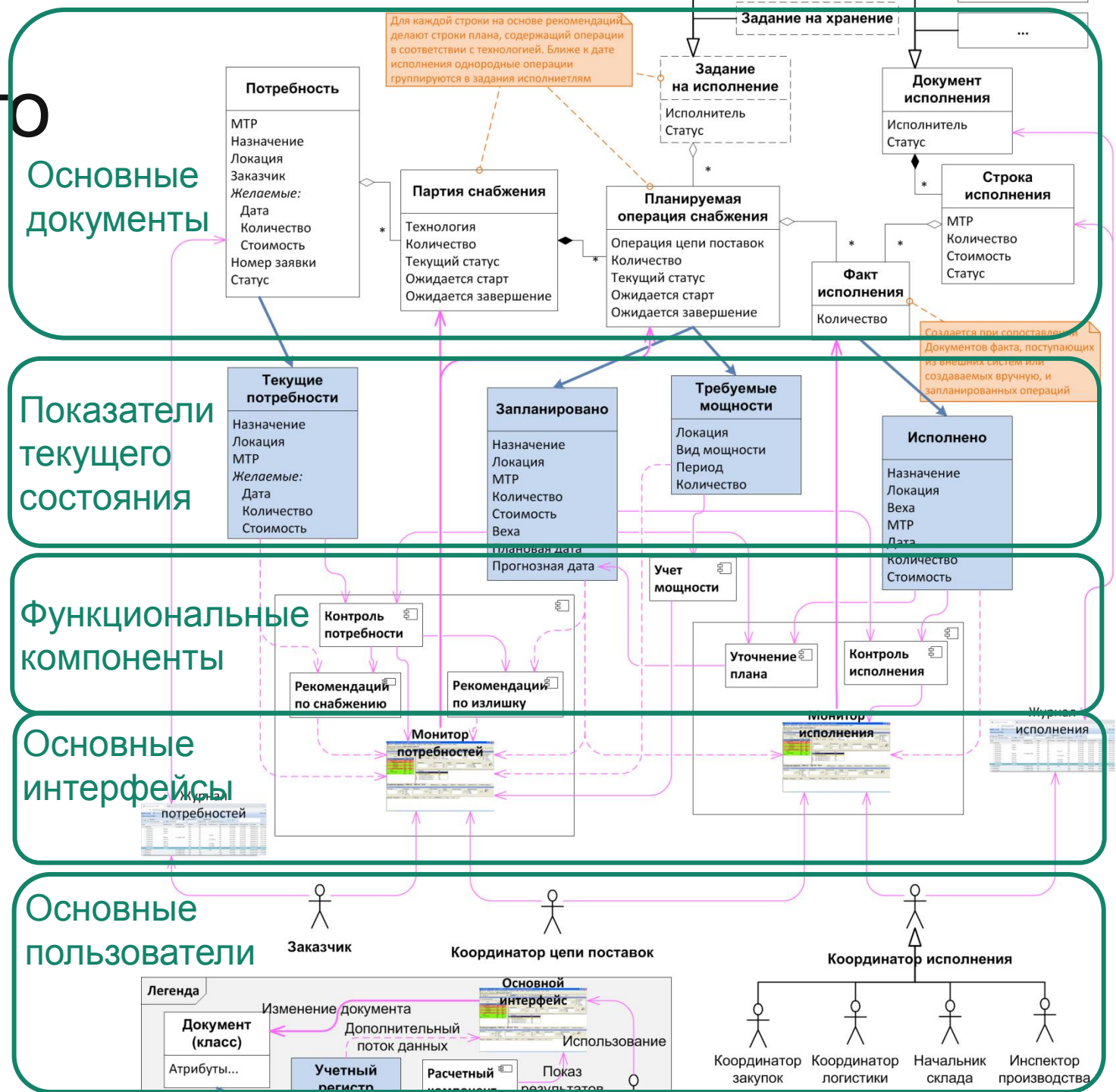
Функциональная архитектура снабжения



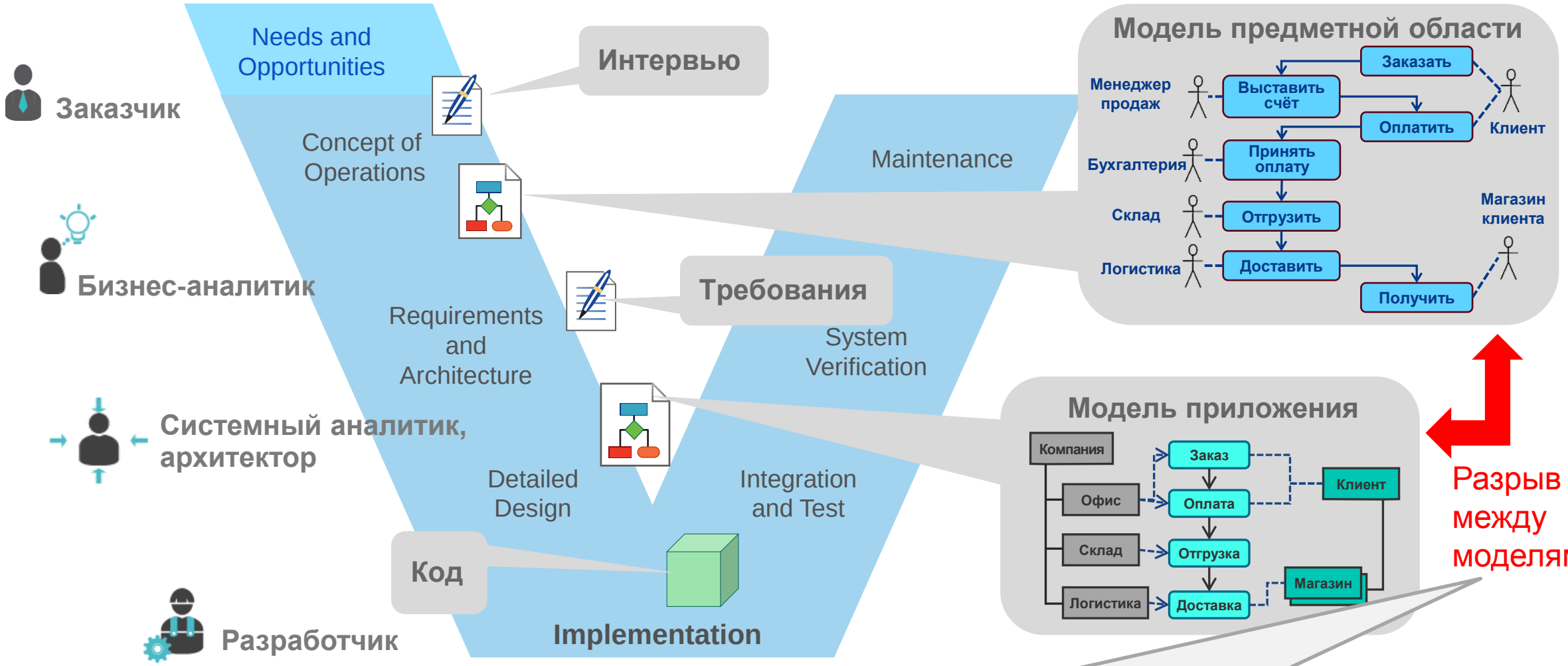
Концепт адаптивного снабжения

Логика работы системы:

- Первичные документы отражают происходящее
- Показатели дают агрегированную картину
- Пользователи:
  - через витрины показателей видят проблемы и рекомендации их решения,
  - и действуют через функциональные компоненты



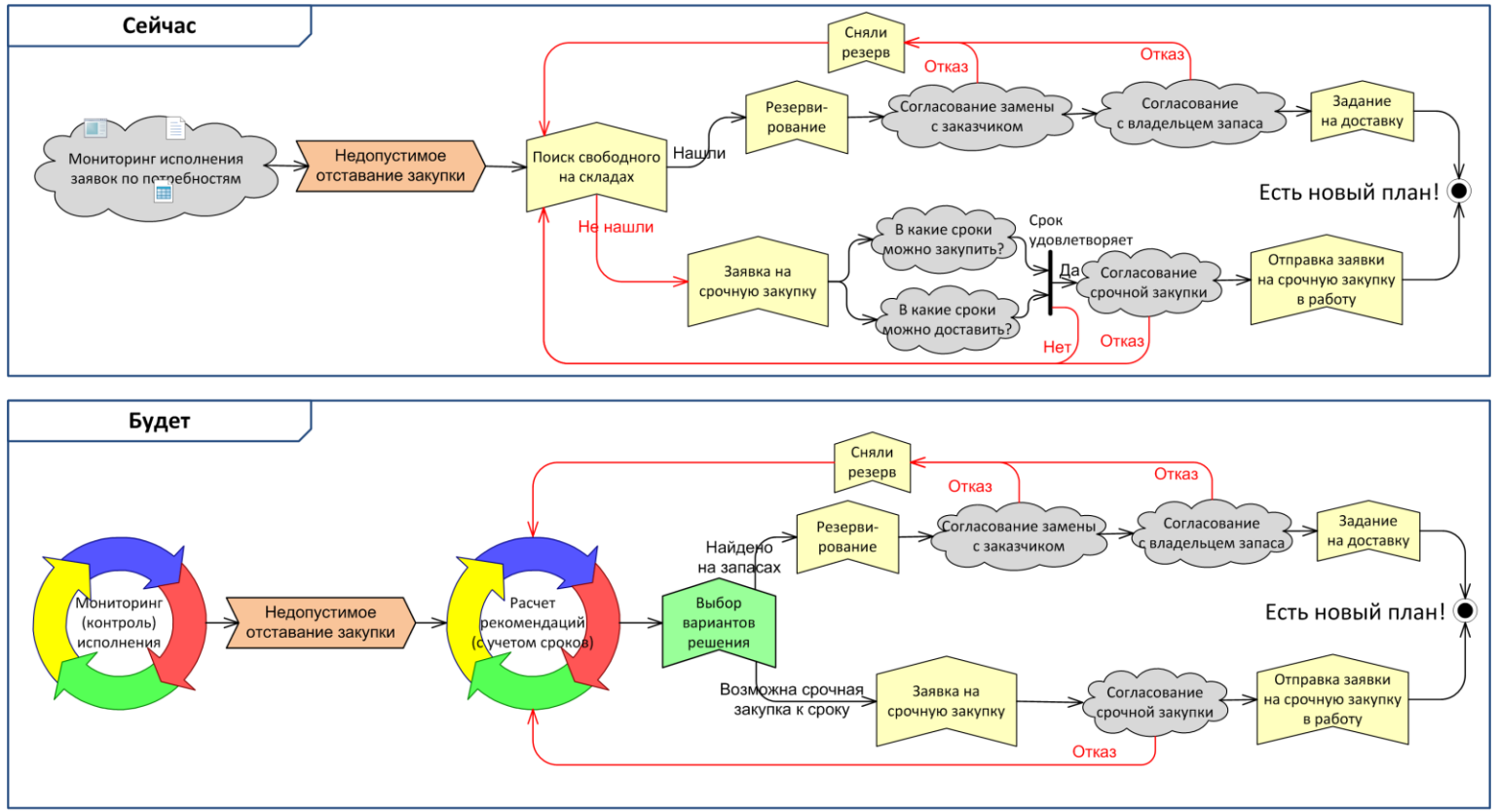
Поддержка артефактами (классика)

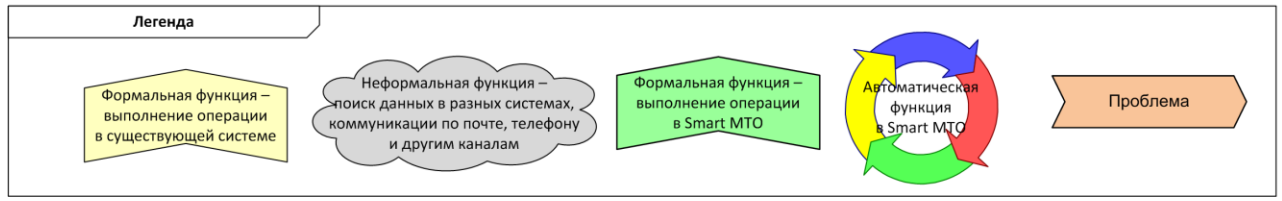


Проблема: каждый уровень отдельно – сложно изменять и поддерживать соответствие

Адаптивное снабжение: что изменится?

Процесс мониторинга исполнения и перепланирования при проблемах

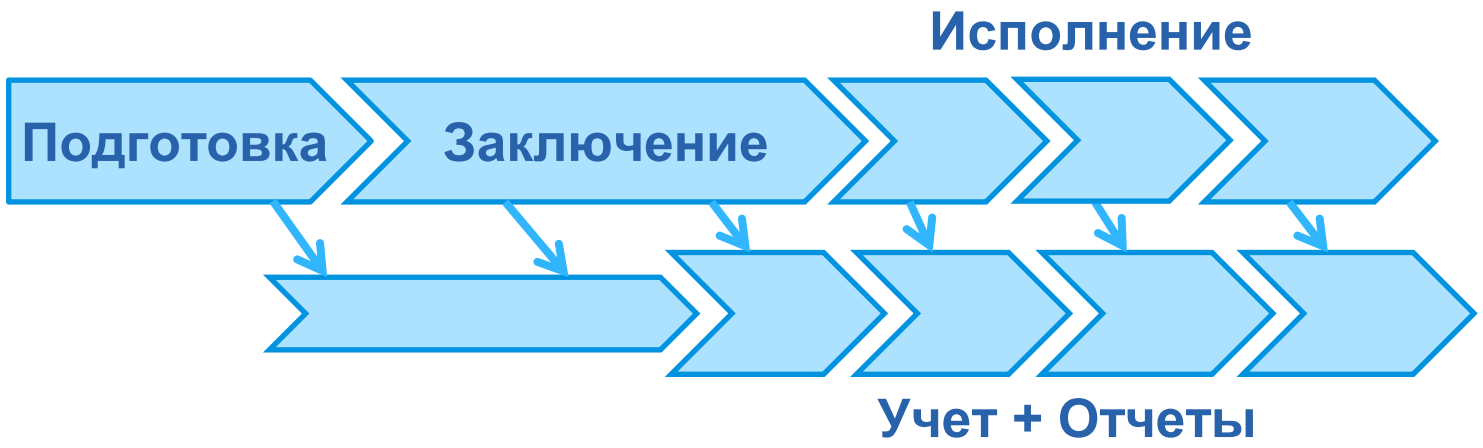




Чтобы схема ясно показала изменения, значки должны различаться. Не бойтесь отступить от формальной нотации, ищите наглядные представления!

Фокус на проблемных точках

Семейство примеров – ведение сделок



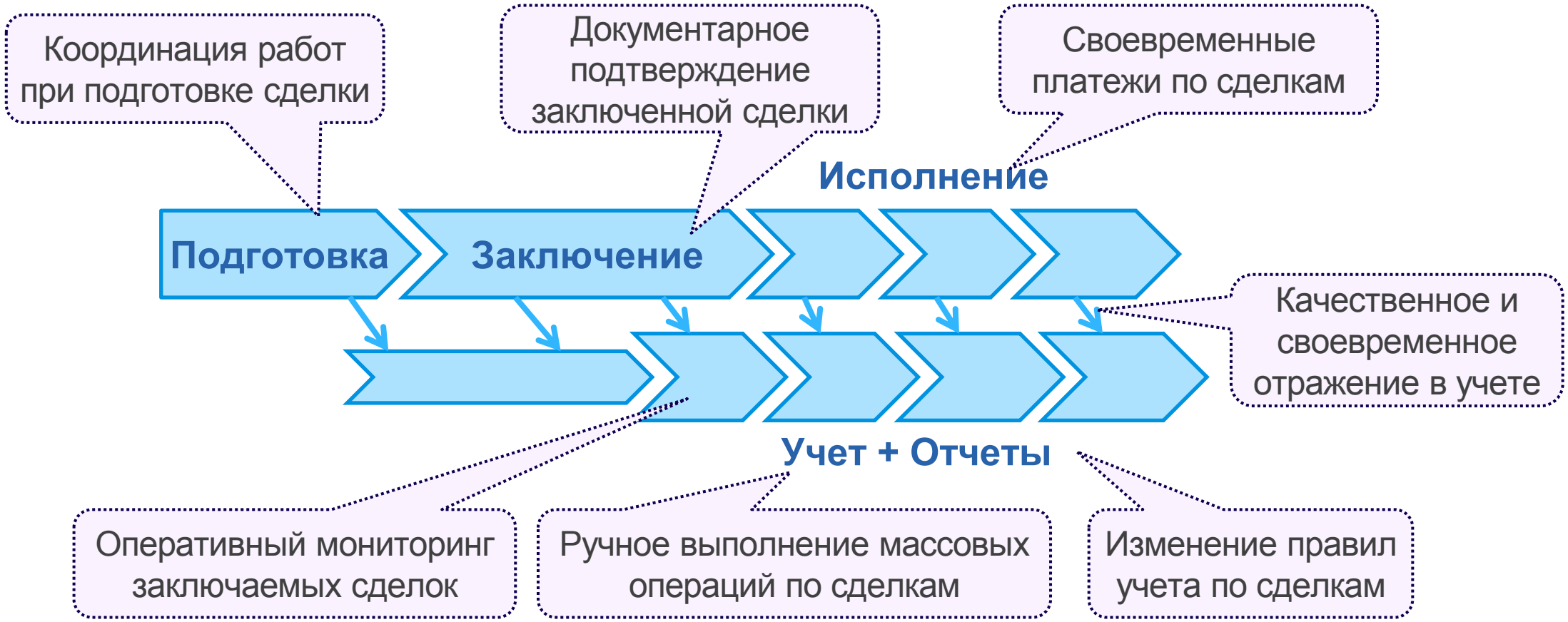
- Ипотечные договора
- Сложные договора кредитования и аккредитивов
- Сделки на финансовых рынках
- Оптовая торговля с долгосрочными контрактами
- Заказы интернет-магазина

и многое другое…

Проблемная точка может быть на разных этапах

- Ипотечные договора
  - **Основная проблема**: на этапе подготовки: много разных специалистов должны проверить договор в разные сроки
  - **Решение**: гибкая система визирования сделки, настройка ее для самых распространенных и запуск в эксплуатацию, на первом этапе – даже без передачи в существующие системы, для них сохранен ручной ввод
  - **Место риска**: достаточная гибкость настроек для всех типов сделок
- Сложные договора кредитования и аккредитивов
  - **Основная проблема**: регулярные массовые операции, которые не может проводить существующая система
  - **Решение**: обобщенная система хранения и выполнение массовых операций, на первом этапе – загрузка данных из существующих систем, а не выгрузка в них,
  - **Место риска**: после первого этапа разработку остановят на кусочном решении

Многообразие проблемных мест

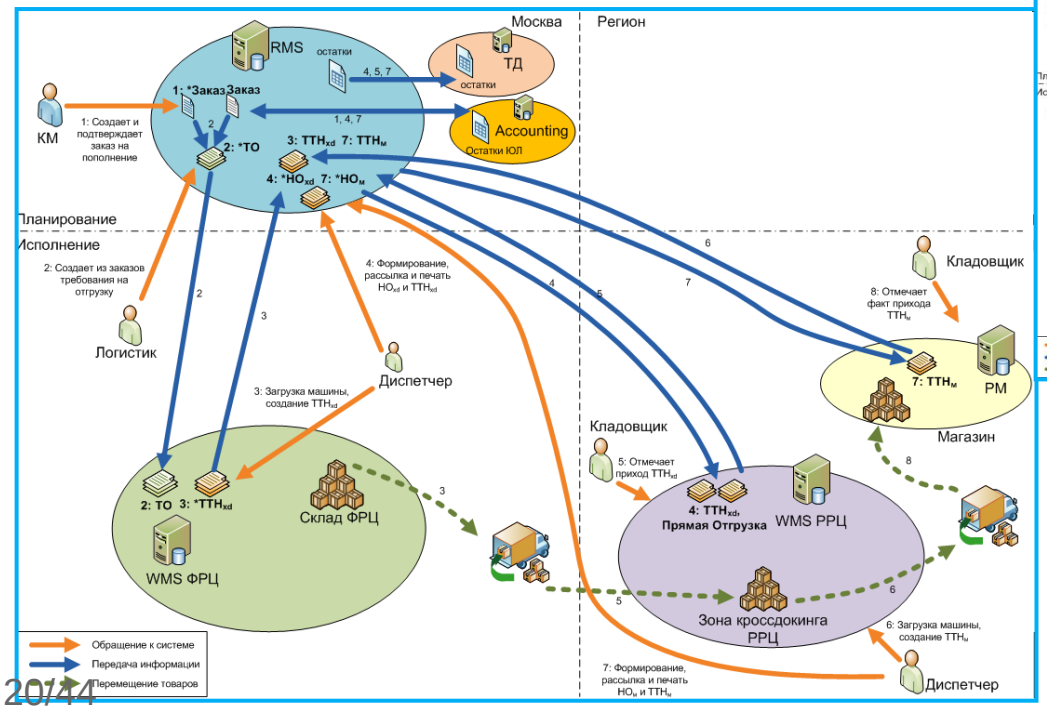


Workflow документов – основной способ реализации процессов

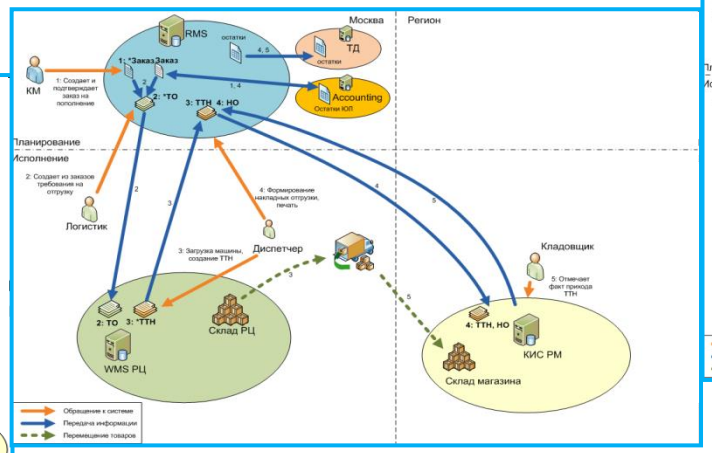
Неформальная модель бизнеса

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.

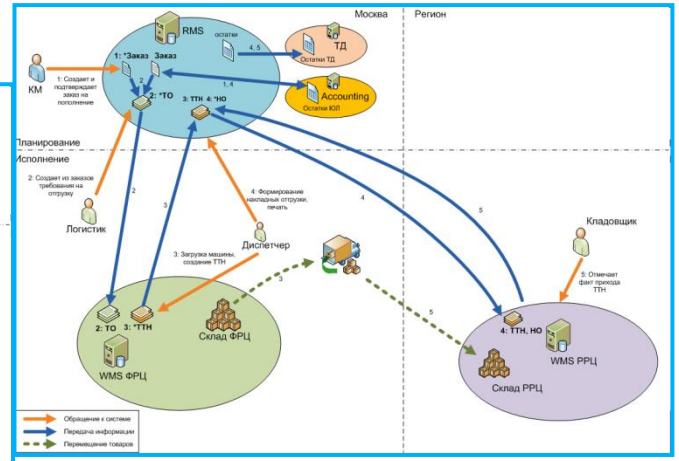
1



2



3



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

Формальная бизнес-модель и реальность

Оптовые продажи магазинам и торговым сетям

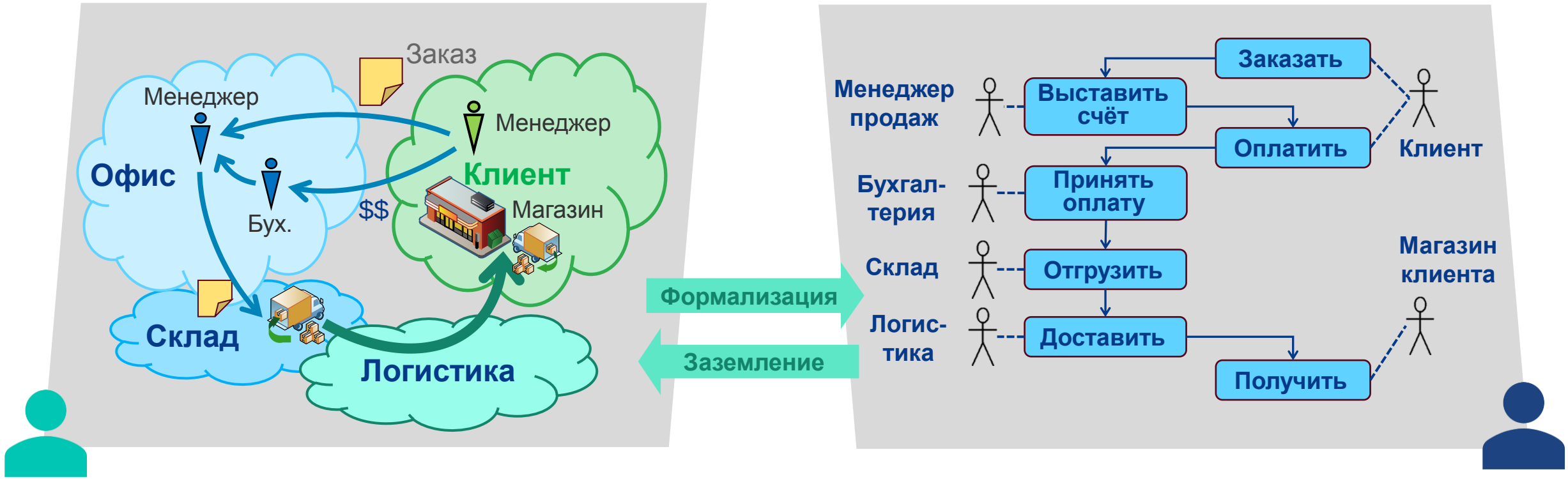
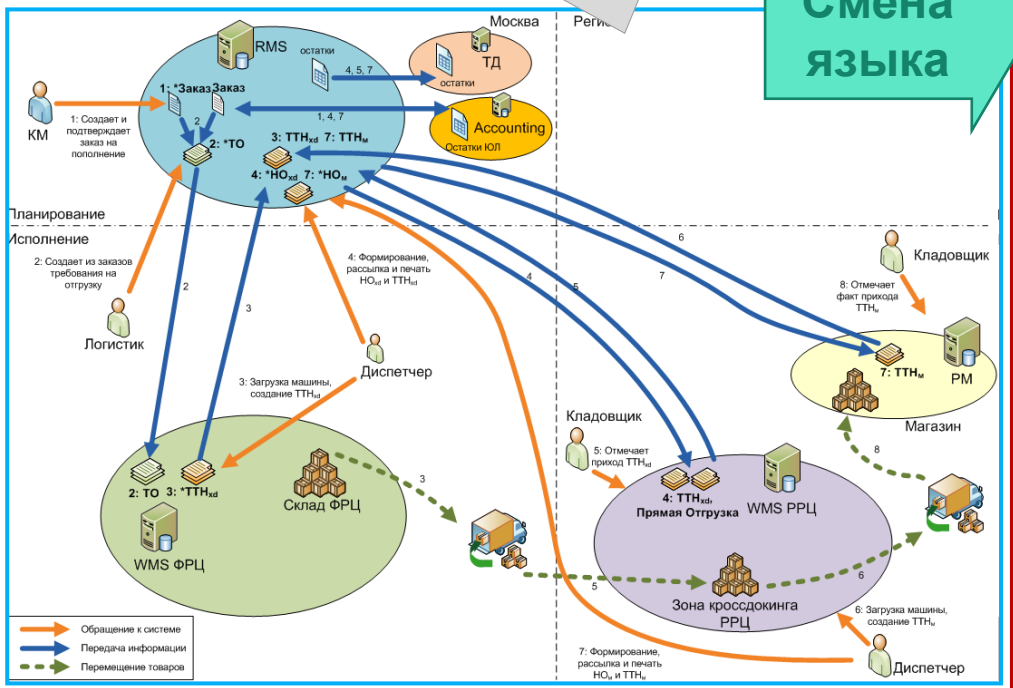


Схема бизнес-процессов – это модель происходящего в реальности.
Видим повседневную деятельность за формальной бизнес-моделью!

Формализация снабжения магазинов

Неформальная схема деятельности и её отражение в существующих системах

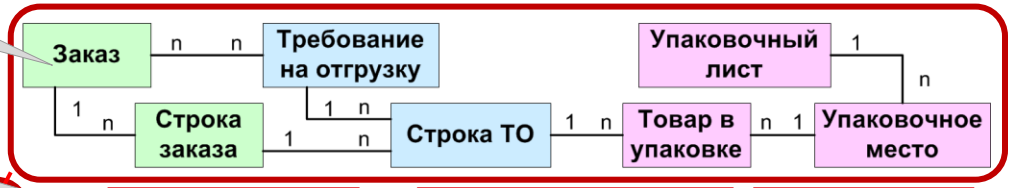


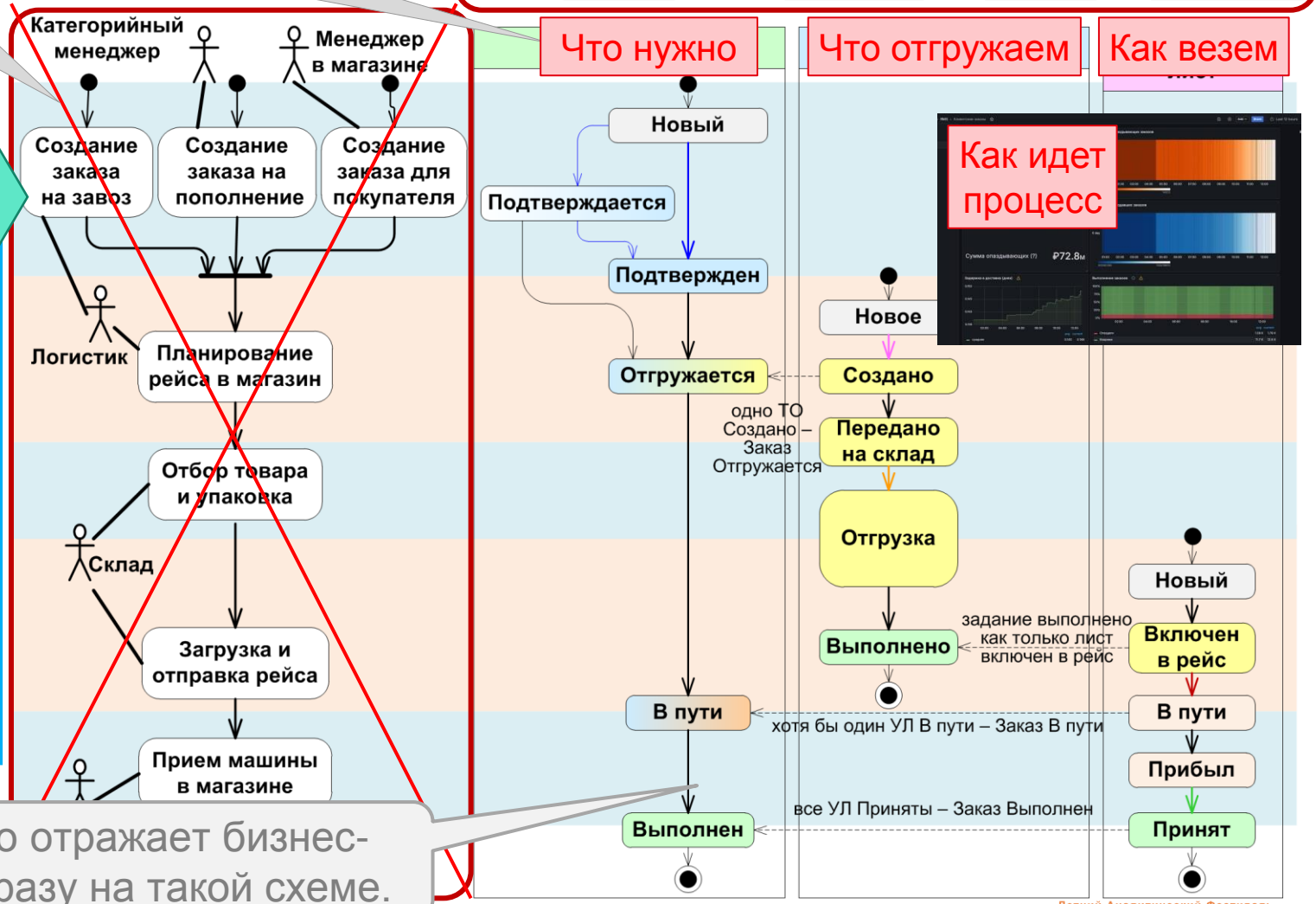
Смена языка

Бизнес-процесс — Activity Diagram

Объекты — Class Diagram

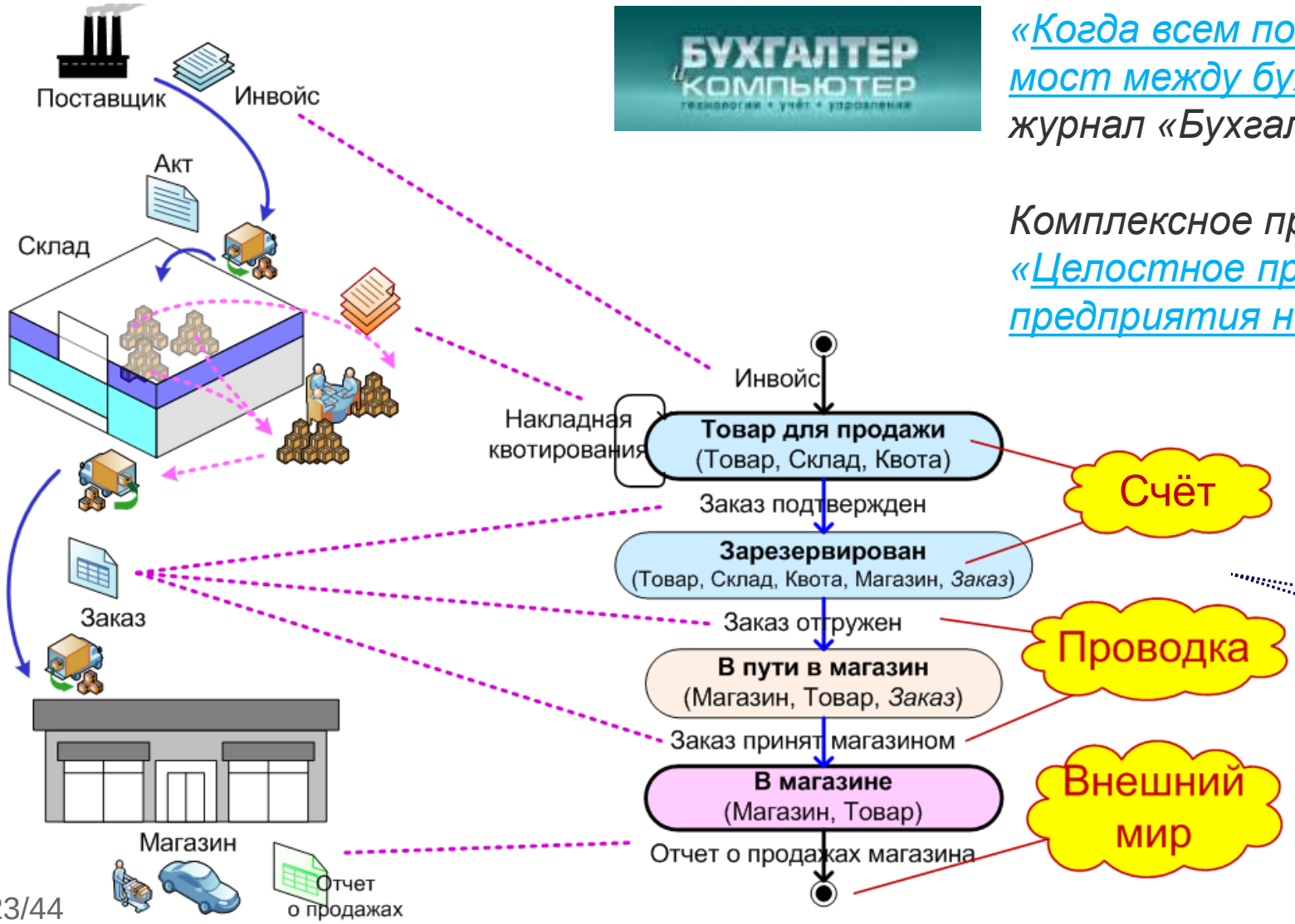
Состояния документов — State Diagram





Если workflow документов прозрачно отражает бизнес-процессы, то можно обсуждать их сразу на такой схеме.

Учёт: бизнес как поток ресурсов



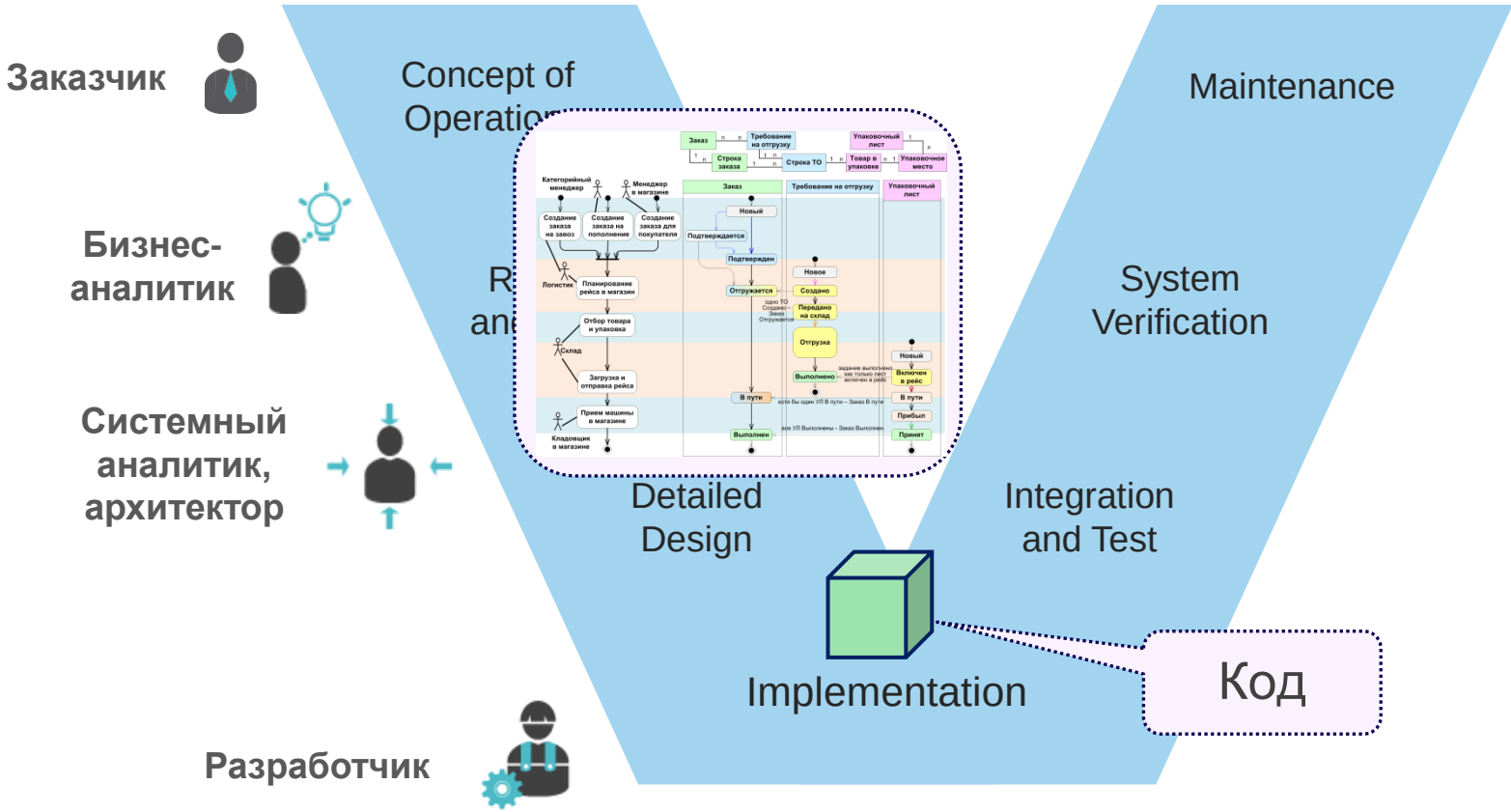
«Когда всем понятно. Диаграммы учёта: мост между бухгалтером и разработчиком» — *журнал «Бухгалтер и компьютер», №5-2011.*

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

Движение товаров
и других ресурсов отражается
как проводки между счетами

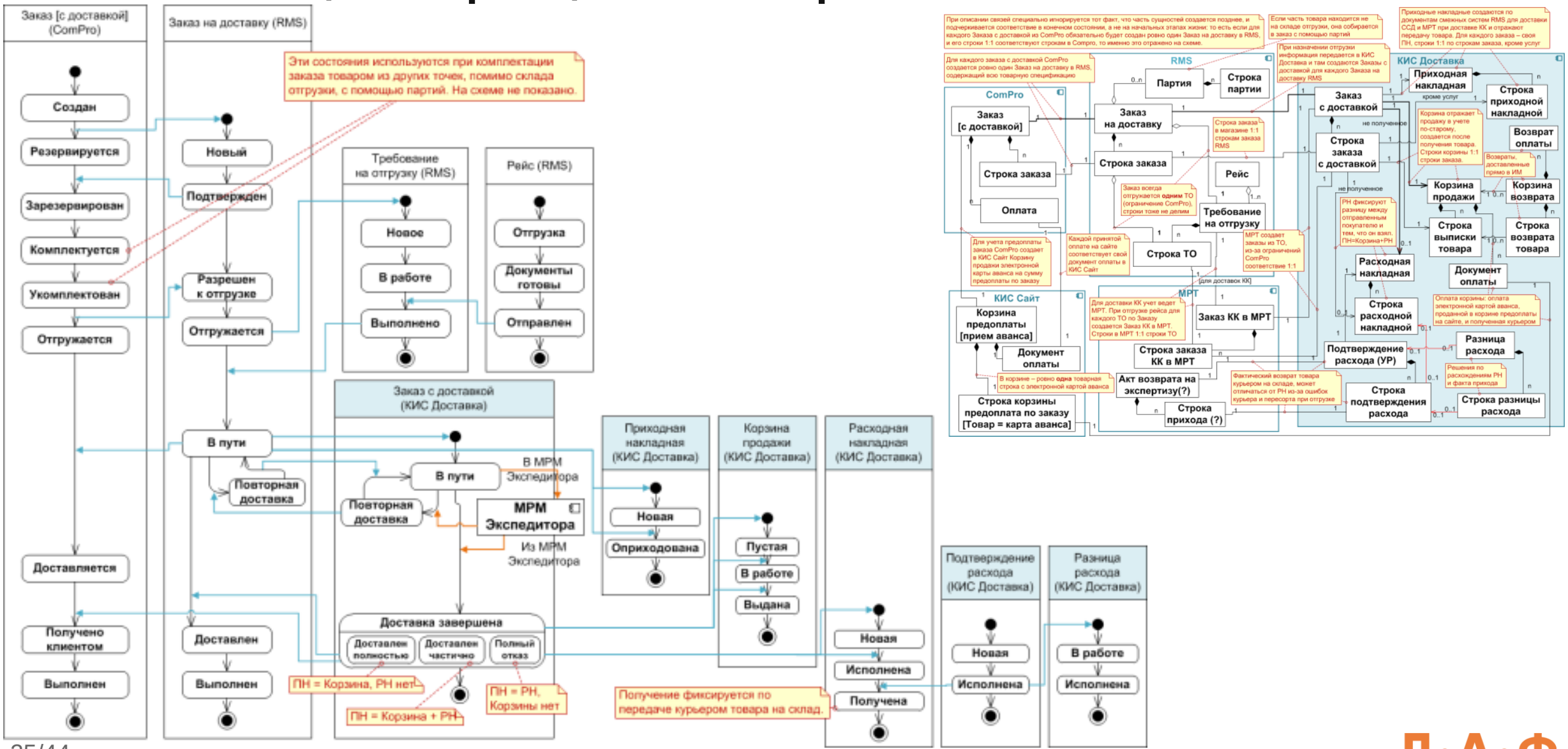
Domain Driven Design

- 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)».

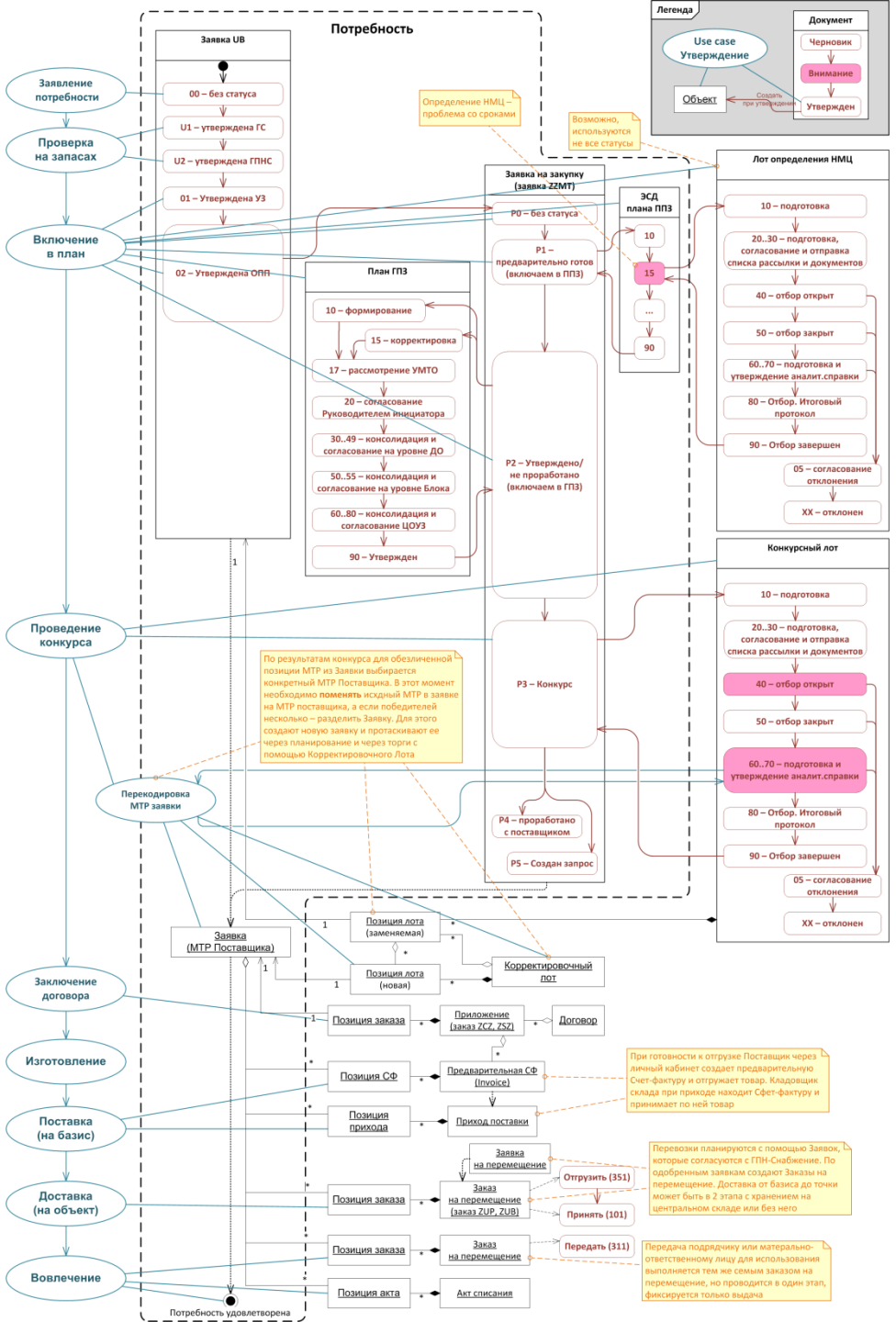
Реализация процесса через workflow



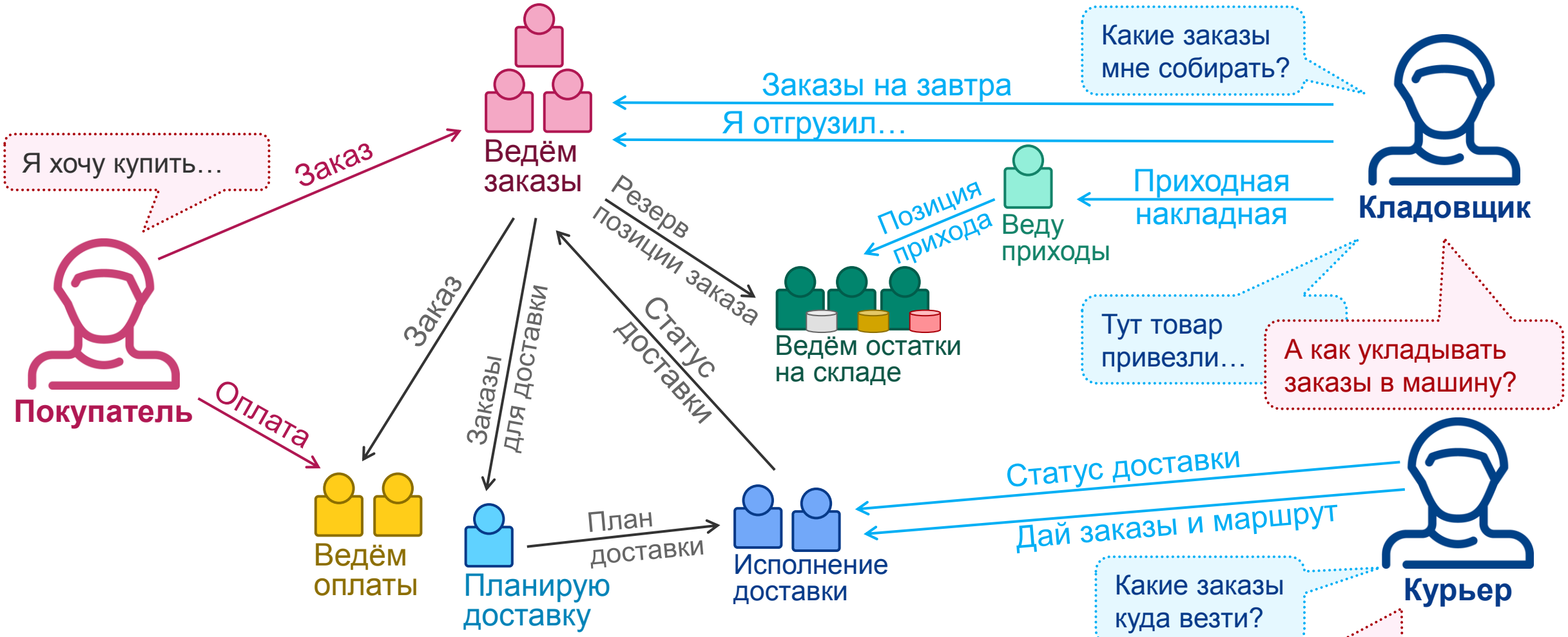
Л·А·Ф
Летний Аналитический Фестиваль

Исторические наслоения реального workflow

- Бизнес-процесс усложнялся, для поддержки в системе приспосабливали разные документы
- Точки внимания бизнеса оказывались глубоко «закопаны» в workflow – их не видно на интерфейсе
- Вторая часть workflow реализована множеством не связанных документов – это порождает проблему трассировки



Модель для сервисной архитектуры

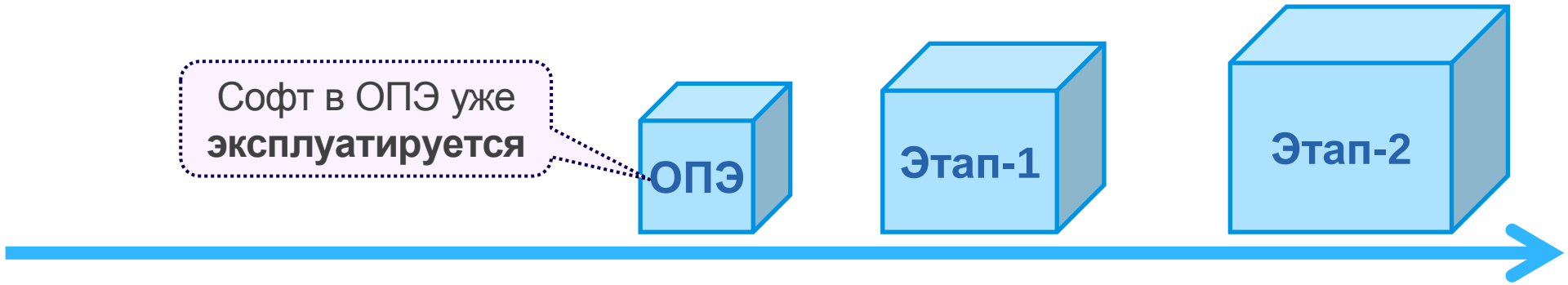


Из доклада «Визуальное проектирование масштабируемых приложений»

Планирование проекта от демо

Сценирование проекта. Этапы

- Общий скоуп проекта определяется бизнес-целями проекта заказчика и не всегда может быть изменен
- Большой проект часто можно разбить на этапы
- Есть опытно-промышленная эксплуатация **(ОПЭ)**, в нее может быть передан ограниченный функционал



Первый этап для ОПЭ – замкнутый функционал, решающий существенную проблему бизнеса.
Принцип MVP (Minimum Viable Product) для ОПЭ и далее

Как выделять первый этап?

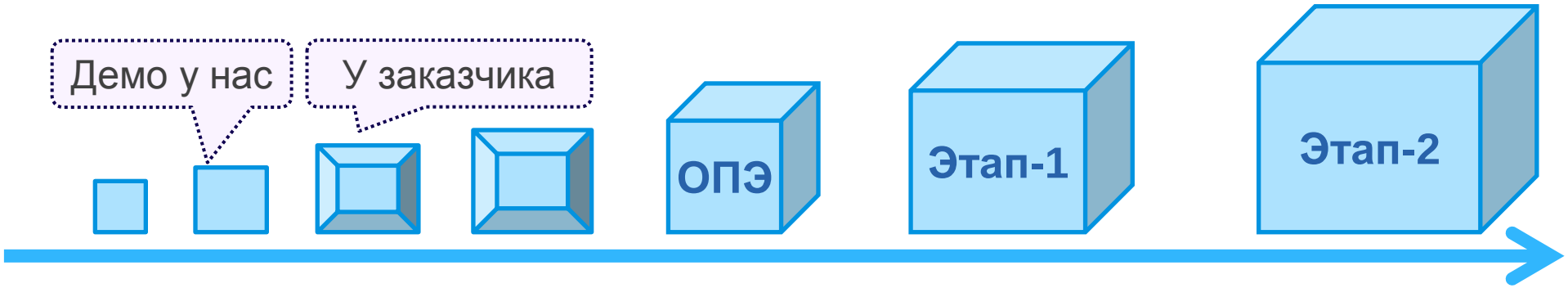
- Понять ключевые проблемы у бизнеса и у ИТ
- Договориться, какие из них будем решать первым релизом
- Понять минимальный объем целостного функционала
  - Объем можно снижать за счет интеграции с существующими системами
- Понять минимальный объем сервиса и бантиков интерфейсов
- Понять минимальный объем многообразия поддерживаемых сделок
  - Часть сделок можно оставить в существующих системах
  - Для части сделок можно сохранить «ручной привод»
  - Но концепт исполнения должен быть ***для всех*** типов сделок

Принцип «**сначала – vision**» работает и при выделении этапов:
мы сначала оконтуриваем будущее, а затем – совмещаем анализ настоящего с проектированием путей достижения будущего

Сценирование проекта. Демо

- Разработка функционала для ОПЭ длится 4–6 месяцев
- Разбиваем его **на 2–4 демо**, представляя **интересный** бизнес-заказчику **целостный** функционал
- Первые демо проводим на нашей территории, так **заказчик знакомится с командой**
- Далее разворачиваем тестовую среду у заказчика, переносим демо в нее, совмещая с испытаниями

Тоже работает **MVP**

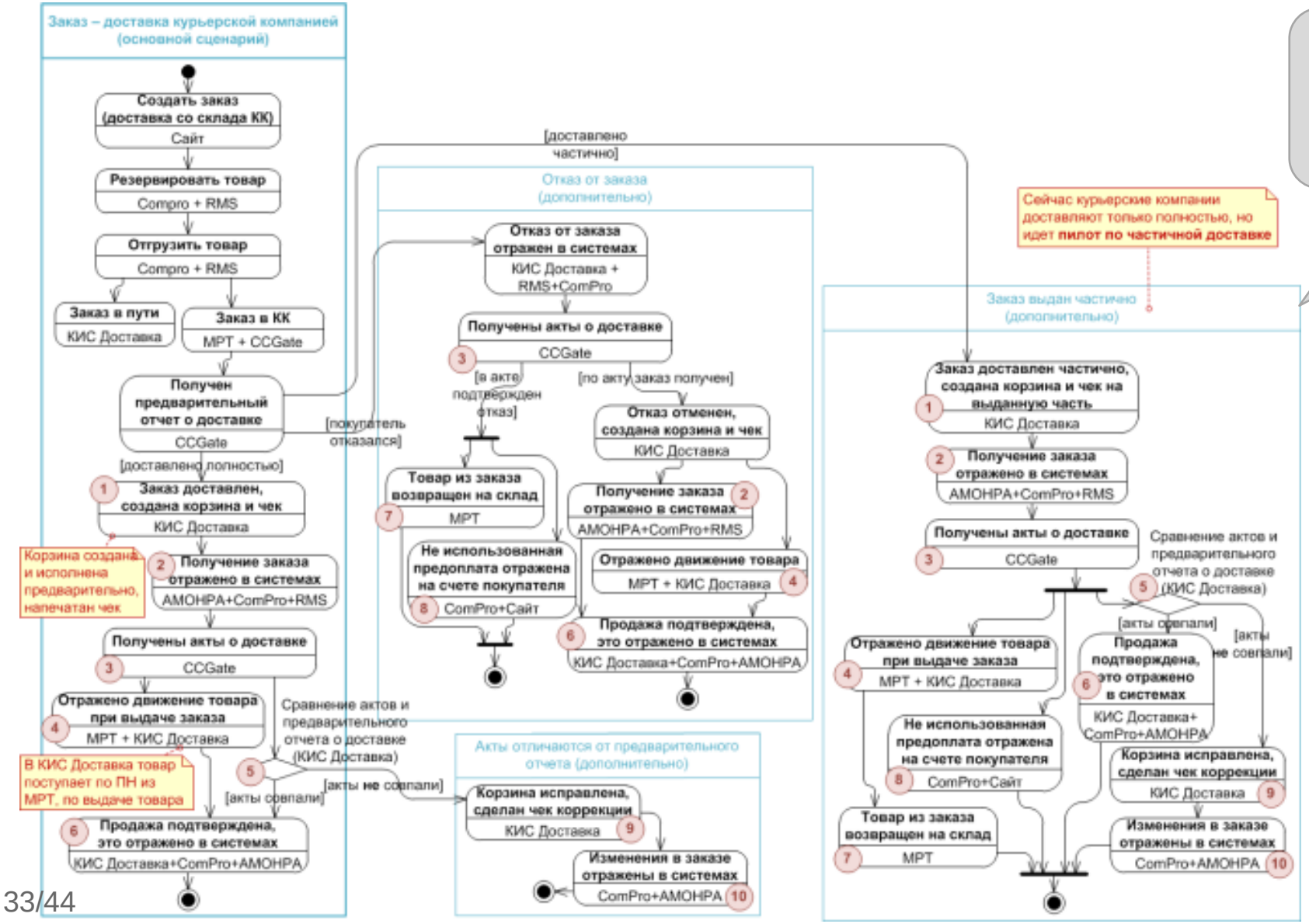


Планирование серии демо

- У каждого демо – своя **целевая группа** и интересный ей целостный функционал: группа должна увидеть **свой** процесс
- Учитываем разрывы с текущей автоматизацией и ожидания стейкхолдеров
  - Нужно показать **ожидаемые улучшения**
  - Включаем в **MVP** не только основной сценарий, но и интересные альтернативы
  - Нужно показать, что «по площади» не станет хуже
- Учитываем логику разработки, но делаем это творчески
  - Если разрабатываем хороший операционный документооборот, полный функционал (документы и справочники) – велик для первого демо
  - Можно показать документооборот, а справочники без интерфейсов
  - Или позвать группу, для которой ценны новшества в справочниках

План демонстраций часто сильно меняет логику разработки, заставляет искать способы создать демонстрируемый функционал быстро, особенно для первых демо, это – тоже проявление принципа «**сначала – vision**»

Сценарии демо: делаем сложное понятным



Сценарии демонстраций могут быть эффективной основой планирования разработки

Подходы к выбору демо

- Если есть ключевая форма – показываем ее и действия на ней, надо получить одобрения тех, кому система принесет ценность
  - Все необходимое (справочники, сделки и др.) – заполняем через БД
  - Заполнять надо реальными данными, которые зрители узнают
  - При показе – разворачивать процесс сначала до ключевой точки
- Если ценность автоматизации лежит в начале процесса
  - Проходим основной сценарий **и** интересные заказчику вариации
  - Показываем, что в целом сервис интерфейсов не стал ниже

Проведение демо

- Демо сценируем и проводим с учетом интересов бизнес-пользователя, на его языке
- Демо обычно проводят аналитики, которые собирали требования – у них уже есть контакт с заказчиком
- Надо вспомнить ожидания заказчиков и учесть при проведении демо
- В демо на нашей площадке участвует вся команда – таким образом заказчик с ней знакомится

Перенос демо к заказчику

- К себе нельзя позвать много пользователей и эксплуататоров
- Для контакта с ними нужно **как можно раньше** получить тестовую среду
- Демо в тестовой среде заказчика – это испытания для ИТ
  и обучение для пользователей, поэтому их полезно разделять
- Надо добиться, чтобы пользователи могли сами посмотреть софт
- Не вся команда участвует – надо доносить до нее обратную связь

Изменения после демо

Участники демо часто просят о дополнительном функционале, что делать?

Договариваемся с драйверами заказчика:

- Они понимают: для успеха бывает необходимо менять требования и скоуп
- Готовы совместно находить решения: обмен скоупа, доп.соглашения и другие варианты
- Иногда (или часто) для поиска решения требуется жесткость позиции, но не конфронтация

Ведите коммуникацию, держа рамку бизнес-целей и совместного успеха

Релизы после ввода в ОПЭ

Определяются **бизнес-потребностями заказчика**

- Релизы к сроку с заданным функционалом
- Релизы заданного функционала к моменту готовности
- Срочные обновления с небольшим функционалом

☹ Это нарушает ритм работы и требует планирования

☹ А также влечет накладные расходы на процесс

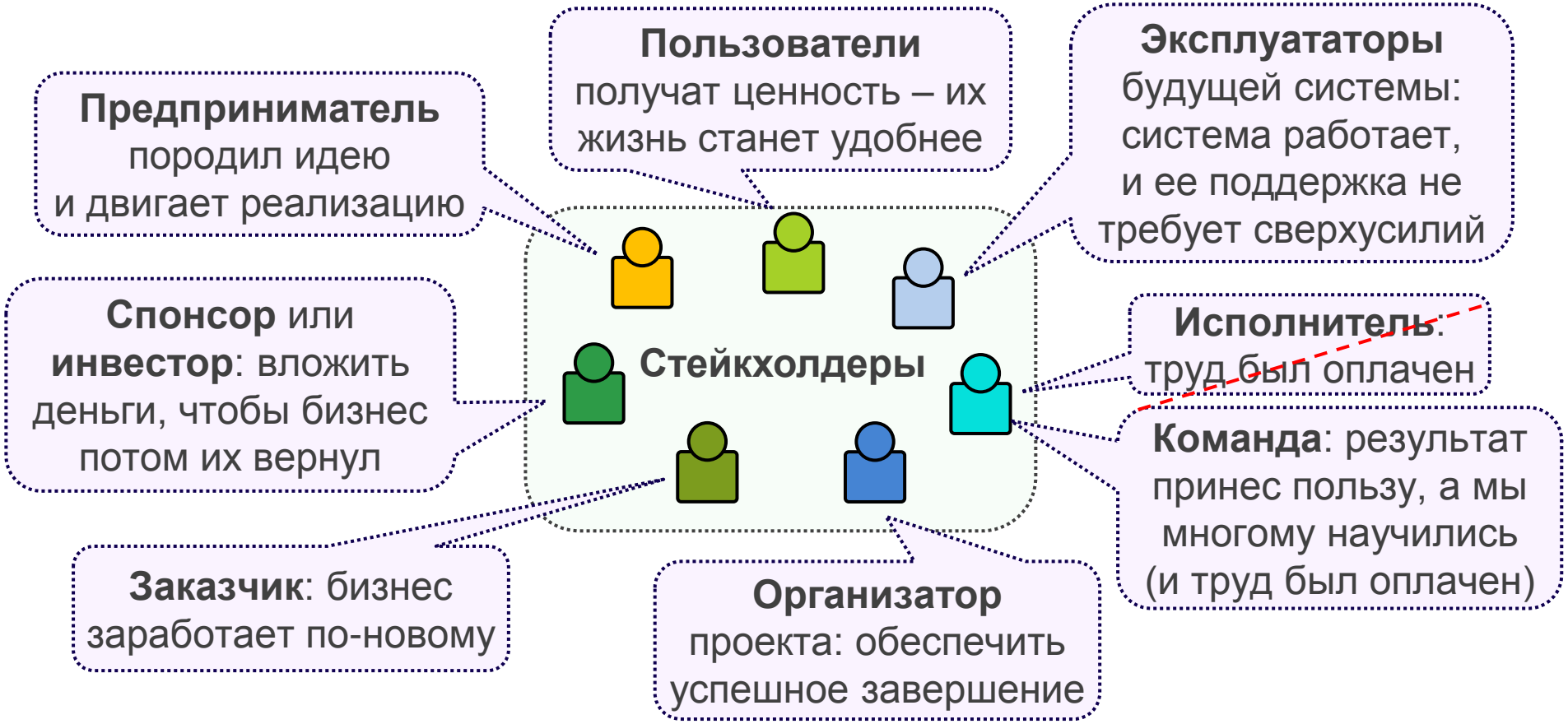
☺ Но обеспечивает решение проблем бизнеса

Continuous Delivery решает эти проблемы, не нарушая ритма,
но требует высокой автоматизации тестирования и обновления ПО

Стейкхолдеры – они разные

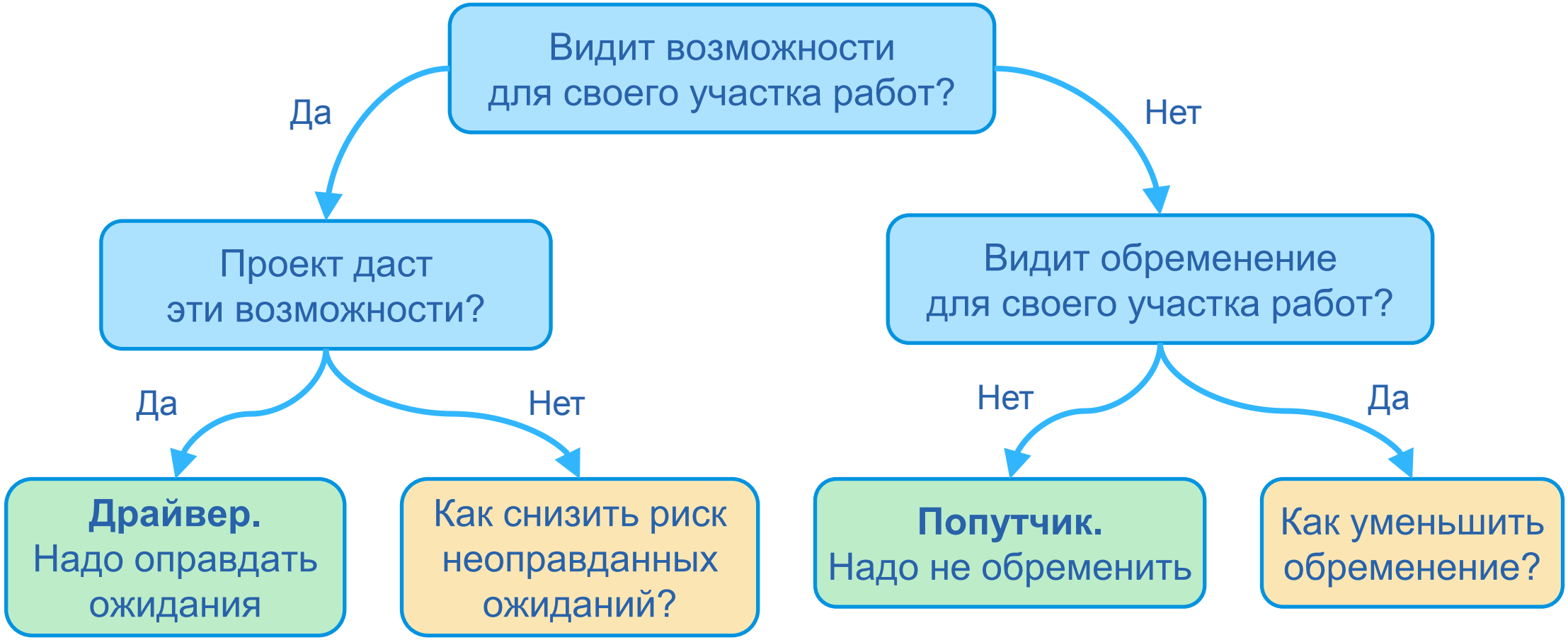
Гармонизация интересов стейкхолдеров – ваша задача, а не заказчика

Виды стейкхолдеров

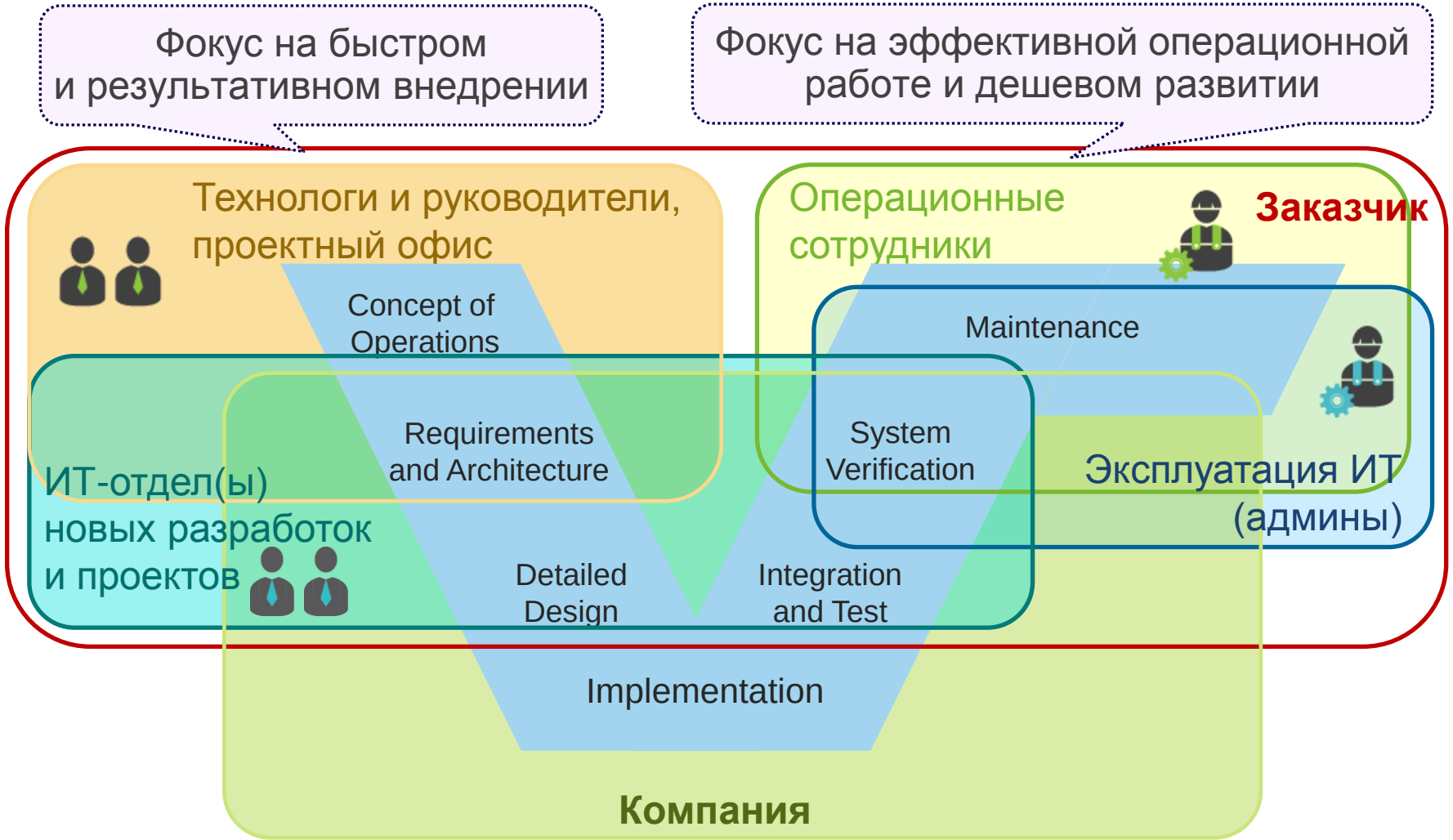


Список не исчерпывающий. У каждого – свои интересы
Стейкхолдеров надо выявить, а их интересы – определить

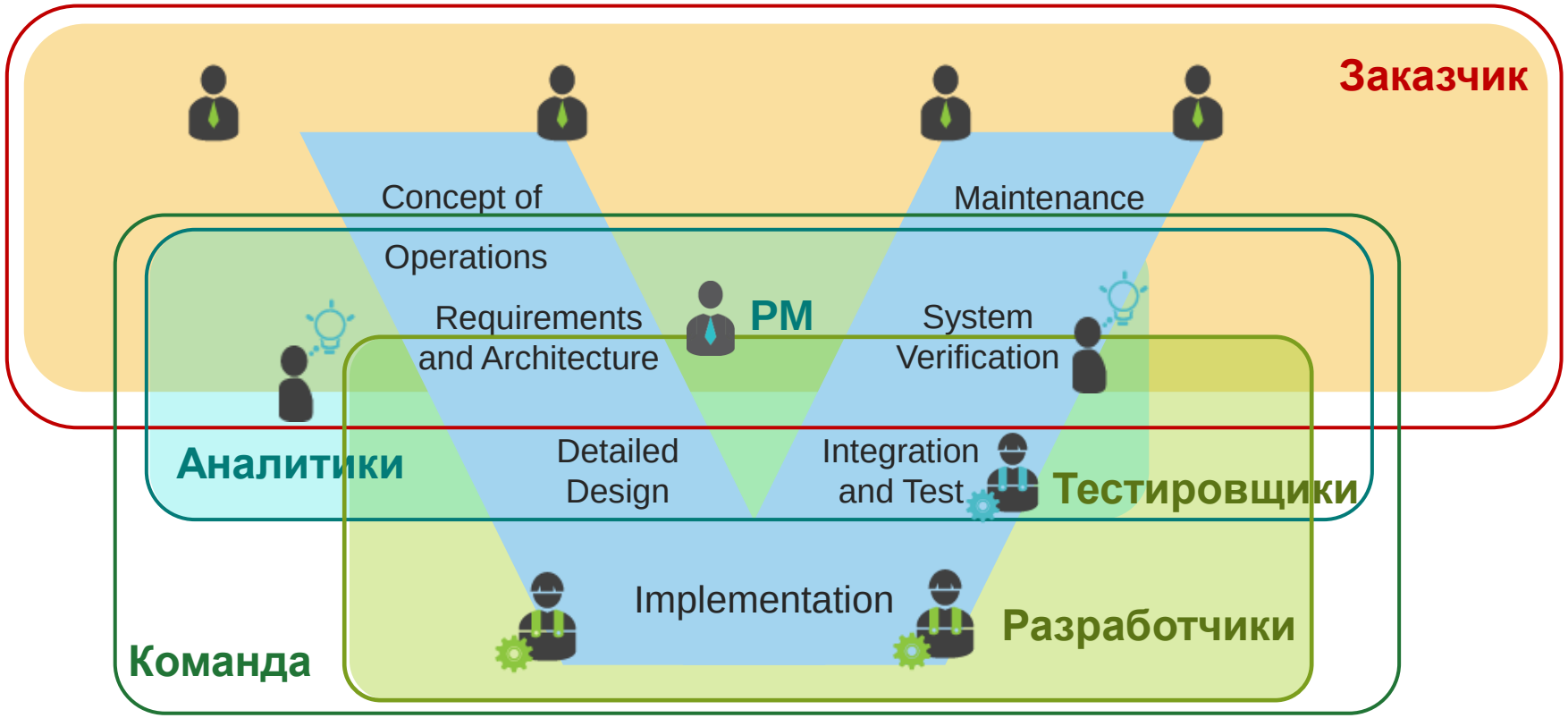
Отношение стейкхолдеров к проекту



Стейкхолдеры заказчика



Разделение ответственности в команде



Я всегда работал в позиции с большим **перекрытием ролевой ответственности**. Аналитик работает с ожиданиями заказчика, и на нем лежит значительная часть ответственности за планирование проекта, проведение демо и внедрение.

Итоги: в чем я вижу ключ к успеху

- Начинайте проект с выяснения реальных проблемных точек
- Сделайте vision, модель системы, которая решит проблемы, покажите ее
- Сценируйте проект, выстраивая взаимодействие через демонстрации, хотя это часто усложняет последовательность разработки проекта
- Используйте неформальные нотации, чтобы подсветить главные смыслы
- Не фокусируйтесь на «правильном» разделении ответственности, действуйте из ситуации для достижения успеха проекта!

Максим Цепков
http://mtsepkov.org
@MaximTsepkov

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

Л·А·Ф
Летний Аналитический Фестиваль