

ЦИФРОВИЗАЦИЯ 2.0: ЛОГИКА ОПИСАНИЯ СИ- СТЕМЫ ЧЕРЕЗ ОБЪЕКТ, РОЛИ И СОБЫТИЯ

Гипотеза описания системы: спецификации как правила связи, фронтир знания и режимы отказа

🕒 30 мин чтения • Цифровизация

Новый подход к описанию системы: логика и что она даёт

Гипотетическая модель. Не руководство к внедрению, а описание логики, которая могла бы лежать в основе системы, спроектированной иначе.

Кому и зачем этот текст

Этот текст — для руководителей. Не для ИТ-директоров, не для архитекторов, не для интеграторов. Для тех, кто принимает решения о цифровизации и потом отвечает за их последствия.

Главный тезис: классический функциональный ландшафт не удерживает единство объекта; текст показывает логику системы, спроектированной не по функциям.

Здесь описан **подход к описанию системы** — не продукт, не вендор, не «купите и внедрите». Подход, который отвечает на вопросы, на которые классический ландшафт ответить не может: где источник правды, кто владелец стыка, как расширяться без перестройки, как планировать честно.

Текст устроен так:

С чем сравниваем — коротко.

Логика нового подхода — семь шагов и следствия из них.

Ключевые понятия — на языке руководителя.

Что это даёт по сравнению с классическим подходом.

Как это работает на объектах из разных бизнес-процессов.
Гибкость, планирование, анализ — как развёртывание логики.
Чем это отличается от REA, event sourcing, DDD и ISA-95.
Ограничения и что делать сейчас.

Часть 1. С чем сравниваем

Классический ландшафт промышленного ИТ устроен по функциям. Есть функция «закупки» — для неё система. Есть функция «производство» — для неё система. Есть функция «качество» — для неё система. Каждая система создаёт свой объект: позицию заказа, код материала, номер партии, инвентарный номер, табельный номер. Эти объекты **разные**, хотя физически это одно и то же. И проблема не в том, что их не синхронизировали. Проблема в том, что они **и не должны быть одинаковыми** — у них разные роли, разные жизненные циклы, разные правила. Но при этом где-то должно быть видно, что это **одно и то же**. И вот этого «где-то» в классическом ландшафте нет. Отсюда — 22 кластера пересечений и три иллюзии (единый источник правды, полная цифровизация, интеграция как решение). Это рабочая рамка текста, результат предыдущего анализа автора, а не внешняя статистика. И отсюда — вопрос: **как могла бы выглядеть система, если проектировать её не по функциям?** Этот текст — ответ на этот вопрос.

Часть 2. Логика нового подхода: семь шагов

Это **гипотеза проектирования**, а не доказанная теория. Подход собран из практической задачи — планирования производственной линии — и проверяется тем, объясняет ли он случаи, на которых классический ландшафт ломается. Семь шагов — каждый ответ на практический вопрос; вместе они образуют связную логику, но не претендуют на то, что из них «выводится всё остальное». Часть следствий — рабочие допущения, которые ещё предстоит проверить на внедрении.

Шаг 1. Ресурсы лежат на оси событий

У нас есть производственная линия. У нас есть смены. У нас есть персонал. Это **ресурсы**, которые существуют во времени. Если нарисовать их на временной оси, получается линия:

ДОРОЖКА	СЛЕВА ОТ «СЕЙЧАС»	СПРАВА ОТ «СЕЙЧАС»
Линия	Зарегистрированные события (●)	Плановые события (○)
Смена А	Зарегистрированные события	Плановые события
Смена В	Зарегистрированные события	Плановые события

● — событие, которое уже произошло и зафиксировано; ○ — событие, которое только планируется. Ресурсы (линия, смены, персонал) — это **дорожки**, по которым идут события. **Что это даёт.** Нет «отдельной системы для линии» и «отдельной системы для смен». Есть **одна ось**, на которой лежат все ресурсы и все события. Всё остальное — заказ, переналадка, простой, ремонт — разворачивается на этой оси.

Шаг 2. Заказ приходит на линию через спецификацию

Заказ не «падает на линию» сам по себе. Он связан с линией через **спецификацию номенклатуры** — правило, по которому номенклатура превращается в продукт на этой линии.

Спецификация — это не атрибут заказа и не атрибут линии. Это **связь** между ними. Она говорит: «эта номенклатура на этой линии производится так: такой-то техрежим, такая-то производительность, такое-то количество в час».

Когда заказ приходит, количество пересчитывается **через спецификацию производительности** — во время события. Не заранее, не в справочнике, а в момент, когда событие заказа встречается с событием линии.

ЭЛЕМЕНТ	ЧТО ДЕЛАЕТ
Заказ на 1000 шт.	Событие-источник
Правило связи «номенклатура → линия»	Определяет, как эта номенклатура производится на этой линии
Правило связи «производительность»	Пересчитывает количество в часы работы
Событие производства	1000 шт. = 8 часов на линии

Что это даёт. Спецификация — не «свойство заказа» и не «свойство линии». Это **ребро**, через которое событие заказа превращается в событие производства. Количество — не хранится, а **вычисляется** в момент встречи.

Шаг 3. Спецификация порождает события

Это самое важное. Спецификация — не только связь. Она **порождает плановые события**. Из спецификации «номенклатура–оборудование» выводятся:

Время переналадки — событие планового простоя на переналадку.
Плановый простой — событие регламентного обслуживания.
Контрольные точки — события проверки качества.

Это не «справочные данные». Это **правила, из которых выводятся будущие события** на линии.

ПРАВИЛО В СПЕЦИФИКАЦИИ	ПОРОЖДАЕМОЕ ПЛАНОВОЕ СОБЫТИЕ
Время переналадки — 2 ч	Переналадка
Период обслуживания — 500 ч	Плановое ТО
Контрольные точки — 10%, 50%, 100%	Проверка качества

Что это даёт. Спецификация — это **генератор событий**. Не «описание объекта», а правило, из которого выводятся события в будущем. Меняется спецификация — меняются порождаемые события. Объект не меняется.

Шаг 4. Параллельные ресурсы и рассинхрон

Линия — не одна. С ней параллельно работают смены, персонал, ремонт. У каждой — своя дорожка на оси событий.

ДОРОЖКА	ЧТО НА НЕЙ	ВОЗМОЖНЫЙ РАССИНХРОН
Линия	Работа, переналадки, простой	Линия работает — персонала в смене нет
Смена А	Окно смены	Смена открыта — заданий нет
Персонал	Доступность людей	Персонал занят — линия простаивает

ДОРОЖКА	ЧТО НА НЕЙ	ВОЗМОЖНЫЙ РАССИНХРОН
Ремонт	Окно ремонта	Линия в ремонте — смена и персонал запланированы

Система **сравнивает** дорожки. Если линия планирует работу, а персонала в смене нет — это **рассинхрон**. Если линия в ремонте, а смена с персоналом запланирована — это тоже рассинхрон.

Система видит рассинхрон и показывает его. Дальше — развилка:

Перестроить график — если это в её власти.

Запросить устранение — если это управленческое решение.

Что это даёт. Анализ — не «отчёт по линии» и не «отчёт по персоналу». Это **сравнение параллельных дорожек** на одной оси. Расхождение видно сразу, и у него есть владелец решения.

Шаг 5. Факт ставит фронтир

Факт снимается **не по времени**. Факт снимается тогда, когда он **зарегистрирован**. И в момент регистрации он **устанавливает фронтир** — границу знания.

ЗОНА	ЧТО В НЕЙ	СТАТУС
Слева от фронта	Зарегистрированные события	Знание (факт)
На фронте	Последний зафиксированный факт	Граница знания
Справа от фронта	План, прогноз, обязательство	Предположение

Фронтир — это не «сейчас по часам». Это **граница между знанием и предположением**. Слева от фронта — зарегистрированные факты. Справа — план, прогноз или обязательство. Числа в примерах ниже — это оценка уверенности, а не математическая вероятность: 1,0 означает «зафиксировано как факт», значения меньше единицы — «ожидание».

Будущее **пересчитывается от фронта** по плановым событиям. Если факт зарегистрировали позже — фронтир сдвинулся позже — будущее пересчиталось от новой границы.

Что это даёт. План — не «статичный документ». План — это **линия будущего**, которая пересчитывается каждый раз, когда факт сдвигает фронтир. И это не «перепланирование»,

а естественное продолжение линии от новой границы знания.

Шаг 6. Эволюция через спецификации

Система расширяется **не перестройкой**, а **добавлением спецификаций и объектов**.

Новое требование — новая спецификация. Она связывает существующие объекты с новым контекстом. Объекты не меняются. Системы-потребители не перестраиваются.

Новое требование.

Новый тип спецификации — связывает существующие объекты.

Новый тип события — порождается спецификацией.

Новая проекция — взгляд на поток событий.

Новый владелец роли.

Что это даёт. Расширение — это **добавление рёбер и узлов**, а не перестройка ландшафта.

Новый регулятор, новый рынок, новое подразделение — всё это новые спецификации, а не новые системы.

Шаг 7. Точность и горизонт — от зрелости спецификации

Спецификация имеет **зрелость**. Четыре уровня:

УРОВЕНЬ	ЧТО ЭТО	ТОЧНОСТЬ	ГОРИЗОНТ
На глазок	Экспертная оценка	Низкая	Широкий, с большим разбросом
Норматив	Справочник, норма	Средняя	Средний
Расчёт	Расчёт по модели	Высокая	Узкий
Статистика	Статистика по фактам	Очень высокая	Очень узкий

По мере внесения факта спецификация **уточняется эволюционно**. Факт — это новое событие. Сравнение факта со спецификацией даёт сигнал. Сигнал уточняет спецификацию. Следующий план точнее.

Спецификация «на глазок» даёт план на широком горизонте с низкой уверенностью. Зарегистрированный факт сравнивается со спецификацией; расхождение даёт сигнал. Сигнал уточняет спецификацию до норматива — горизонт сужается.

Накопленная статистика уточняет спецификацию дальше — горизонт становится ещё уже. Плюс **статистические подсказки**: система может предложить уточнение спецификации на основе накопленных фактов — но решение об уточнении остаётся за владельцем роли.

Что это даёт. В логике модели спецификации уточняются эволюционно: не «переобучение», а уточнение по фактам. Чем больше фактов, тем точнее план. И это видно:

вы знаете, на каком уровне зрелости находится каждая спецификация, и какой горизонт планирования она обеспечивает.

Что из этого следует

- Семь шагов — это один цикл:
- Ресурсы лежат на оси событий.
- Заказ приходит на линию через спецификацию.
- Спецификация порождает плановые события.
- Параллельные ресурсы дают ветки.
- Сравнение между ветками даёт сигнал (рассинхрон).
- Точка решения приводит к управленческому акту.
- Факт ставит фронтир.
- Будущее пересчитывается от фронта.
- Факт уточняет спецификацию — зрелость растёт.
- Следующий цикл точнее.
- В схеме выше: Comparison — сравнение двух точек на линии событий; Signal — отклонение или предпочтение; DecisionPoint — точка решения с владельцем; ManagementAct — зафиксированный управленческий акт; Projection — взгляд на поток событий с заданными параметрами.
- Из этой цепочки следуют пять принципов, которые в классическом подходе звучат как аксиомы. Здесь они — **следствия принятой гипотезы**, а не доказанные законы.

ПРИНЦИП	ИЗ КАКОГО ШАГА СЛЕДУЕТ
От функции к роли	Шаг 2: объект один, роли разные
От атрибута к событию	Шаг 5: факт — событие, атрибуты — проекции
От плана к проекции	Шаг 5: план — линия будущего от фронта
От спецификации как атрибута к спецификации как связи	Шаги 2–3: спецификация связывает и порождает
От «единого источника» к «единой модели»	Шаг 6: эволюция через спецификации

Часть 3. Ключевые понятия — на языке руководителя

Логика опирается на несколько понятий. Они не технические. Они управленческие.

Событие

Всё, что происходит с объектом, — это событие. «Партия получена», «партия проверена», «партия потреблена», «станок отказал», «сотрудник получил квалификацию», «заказ отгружен». У события есть **модальность**: факт (произошло), план (намерение), прогноз (ожидание), обязательство (договорённость), корректировка, отмена.

Что это меняет. Вопрос «где хранится истина» заменяется вопросом «какие события мы фиксируем и какие проекции из них строим». Истина — в событиях. Атрибуты — в проекциях.

Ресурс и буфер

Ресурс — то, что выполняет работу: линия, станок, сотрудник, поставщик, клиент. **Буфер** — то, что накапливается: материал, WIP, деньги, время, энергия, место. Ресурсы и буферы лежат на оси событий и связаны спецификациями.

Что это меняет. В классическом ландшафте клиент — это объект CRM, поставщик — объект SRM, станок — объект EAM. В новом подходе клиент, поставщик и станок — **ресурсы** с разной автономией. Отдельных сущностей не требуется.

Спецификация — три функции

Это ключевое понятие. Спецификация — не атрибут объекта, а **связь**. И у неё три функции.

О термине. Здесь слово «спецификация» конфликтует с привычной «технологической спецификацией» — документом, в котором записан рецепт. Это не одно и то же.

Технологическая спецификация — лишь один из видов описанного ниже правила (например, «трансформация»). Дальше читайте «спецификация» как **правило связи (specification)**: оно отвечает не на вопрос «что записано об объекте», а на вопрос «как объект связан с другим объектом или контекстом и какие события из этой связи следуют».

Функция 1. Связывает. Спецификация — это ребро графа, а не свойство узла.

СПЕЦИФИКАЦИЯ	ЧТО СВЯЗЫВАЕТ	ЧТО ОПРЕДЕЛЯЕТ
Состав	Партия → изделие	Из каких компонентов состоит изделие
Трансформация	Партия → станок	Как партия превращается в деталь
Проверка	Партия → метод контроля	Как проверять: параметры, допуски

СПЕЦИФИКАЦИЯ	ЧТО СВЯЗЫВАЕТ	ЧТО ОПРЕДЕЛЯЕТ
Коммерческая	Партия → договор	Цена, условия оплаты
Логистическая	Партия → склад	Как хранить: температура, влажность
Правило	Партия → выдача	FIFO/FEFO, правила отбора
Классификация	Партия → отчётность	К какой группе относится
Квалификация	Партия → оператор	Кто имеет допуск к работе

Функция 2. Порождает события. Из спецификации выводятся плановые события: переналадка, простой, контрольные точки. Это не «справочные данные», а **правила, из которых выводятся будущие события.**

Функция 3. Уточняется. У спецификации есть зрелость. По мере накопления фактов она уточняется: «на глазок» → норматив → расчёт → статистика. Чем выше зрелость, тем точнее план и шире горизонт.

Что это меняет. Вы можете спросить «в каких ролях выступает этот объект, какие правила к нему применяются, какие события эти правила порождают и на каком уровне зрелости эти правила?» — и получить полную картину. Сегодня эта картина разбросана по системам, документам и головам.

Фронтир

Граница знания. Не «сейчас по часам», а момент, когда факт был зарегистрирован. Слева от фронта — знание (зафиксированные факты). Справа — предположение (план, прогноз, обязательство). Фронтир сдвигается, когда регистрируется новый факт. Будущее пересчитывается от новой границы.

Что это меняет. План — не «документ», а **линия будущего от фронта.** И это честно: вы видите, где кончается знание и начинается предположение.

Проекция

Взгляд на поток событий с определёнными параметрами: горизонт, модальность, ветка. План в ERP и план в APS — не две копии одного плана. Это **две проекции** одной линии событий. У них разные горизонты, разные допущения.

Что это меняет. Вопрос «где источник правды по плану» исчезает, потому что нет «плана» — есть проекции. Расхождение между проекциями — не ошибка, а **разные вопросы к одной линии событий.**

Сравнение

Единственная фундаментальная операция. Все сигналы и решения — результат сравнения двух точек на линии событий. Факт vs план, факт vs норма, факт vs модель, сценарий A vs B, факт vs факт. Результат — **сигнал** (отклонение) или **предпочтение** (ранжирование сценариев).

Что это меняет. Анализ — не «отчёт», а **сравнение** с явно указанными сторонами: левая точка, правая точка, оператор, порог, владелец решения. Сегодня анализ — это отчёт, который кто-то построил по какой-то методологии. В новом подходе — сравнение, у которого всё явно.

Решение и управленческий акт

Каждый сигнал создаёт **точку решения** с владельцем. Владелец сравнивает сценарии, получает предпочтение, фиксирует **управленческий акт**. Акт — это тоже событие. Оно попадает в поток. Все проекции, которые его учитывают, автоматически отражают его. Различаю владельца роли и владельца решения: владелец роли отвечает за спецификацию и правила роли; владелец решения — за выбор сценария и фиксацию управленческого акта. Это могут быть разные люди.

Что это меняет. У каждого решения есть владелец, причина и история. Сегодня решение принимается на совещании и исчезает. В новом подходе — остаётся.

Часть 4. Что даёт новый подход по сравнению с классическим

АСПЕКТ	КЛАССИЧЕСКИЙ ПОДХОД	НОВЫЙ ПОДХОД
Объект	«Размазан» по системам, у каждой свой	Один объект, роли описаны явно
Источник правды	Не определён, каждая система претендует	В событиях; проекции — производные
Расхождение	Выглядит как ошибка	Разные вопросы к одной линии событий
Владелец стыка	Не назначен	Владелец роли и владелец решения назначены
Решение	Принимается вслепую, на совещании	На основе сравнения сценариев, с причинной связью

АСПЕКТ	КЛАССИЧЕСКИЙ ПОДХОД	НОВЫЙ ПОДХОД
История решения	Исчезает после совещания	Остаётся в потоке событий
Новое требование	Новая система и интеграционный проект	Новая спецификация или проекция
Расширение	Рост числа систем и стыков	Рост числа ролей, не объектов
План	Документ в системе	Событие в будущем с модальностью
Факт	Перезаписывает план	Новое событие; план сохраняется
Анализ	Отчёт по методологии	Сравнение двух точек на линии
Сценарии	Файлы в Excel	Ветки на линии событий
Точность плана	«Как получилось»	От зрелости спецификации
Горизонт	Фиксированный	От зрелости спецификации
Рассинхрон	Выясняется по факту	Виден заранее
Интеграция	Маскирует расхождение	Модель делает расхождение видимым
Эволюция	Покупка новой системы	Уточнение спецификаций по фактам

Главное: модель позволяет задавать вопросы, которые сегодня не имеют ответа.

Покажите все роли этого объекта.

Кто владелец каждой роли?

В каких спецификациях участвует объект и какие правила к нему применяются?

Какие события порождает каждая спецификация?

На каком уровне зрелости каждая спецификация и какой горизонт она обеспечивает?

Где рассинхрон между параллельными ресурсами?

Что было планом, что фактом, что корректировкой?

Какие сценарии сравнивались перед решением?

Кто принял решение и на основании чего?

Что произойдёт, если добавить новое требование?

Сегодня эти вопросы либо не имеют ответа, либо ответ собирается вручную из пяти систем. В логике модели такой ответ выводится из описания ролей и событий.

Часть 5. Как это работает на объектах из разных бизнес-процессов

Логика едина. Но проявляется по-разному на разных объектах. Ниже — пять сценариев из разных бизнес-процессов. В каждом показано: какие сущности задействованы, какие шаги логики работают, что становится видимым.

В таблицах ниже колонка «Спецификация» указывает тип спецификации; конкретная спецификация описывается отдельно.

5.1. Партия сырья с отклонением

Ситуация. Партия стали приходит с отклонением по прочности. Отклонение в пределах допуска, но близко к границе. Нужно решить: использовать, вернуть или переработать.

Роли партии:

роль	тип спецификации	владелец
Закупаемая позиция	commercial	Закупки
Объект контроля	verification	ОТК
Вход процесса	transformation	Технолог
Единица хранения	logistic	Склад

Как работает логика:

шаг	что происходит	шаг логики
1	Событие «партия получена» фиксируется в общем потоке	1
2	Событие «входной контроль» фиксирует отклонение	2–3
3	Система сравнивает факт с допуском → сигнал	4
4	Сигнал создаёт точку решения с владельцем — технологом	4
5	Технолог сравнивает три сценария: использовать, вернуть, переработать	сравнение сценариев (часть 7.5)

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
6	Система ранжирует по стоимости, сроку, качеству → предпочтение	сравнение сценариев (часть 7.5)
7	Технолог фиксирует управленческий акт	4
8	Все проекции отражают решение	5
9	Факт уточняет спецификацию контроля	7

Что становится видимым. Не только «партия с отклонением», но и «кто владелец решения», «какие сценарии сравнивались», «почему выбрали этот», «как уточнилась спецификация после факта». Сегодня это знание исчезает после принятия решения.

5.2. Заказ клиента с изменением срока

Ситуация. Клиент просит перенести срок поставки на две недели раньше. Нужно решить: принять, отклонить, предложить компромисс.

Роли заказа:

РОЛЬ	ТИП СПЕЦИФИКАЦИИ	ВЛАДЕЛЕЦ
Обещание клиенту	commitment	Директор по продажам
Производственное задание	sequence	Директор по производству
Элемент расписания	transformation	Планировщик
Груз к отгрузке	logistic	Логистика
Финансовая единица	commercial	Финансы

Как работает логика:

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
1	Событие «запрос на изменение срока» фиксируется	1
2	Система показывает все роли заказа	2

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
3	Система сравнивает параллельные ресурсы: линия, смены, персонал, логистика → сигнал рассинхрона	4
4	Система сравнивает сценарии: принять, отклонить, компромисс	сравнение сценариев (часть 7.5)
5	Система ранжирует по марже, загрузке, риску срыва	сравнение сценариев (часть 7.5)
6	Владелец решения фиксирует управленческий акт	4
7	Все проекции отражают решение	5

Что становится видимым. Заказ — не «строчка в CRM», а объект с пятью ролями и пятью последствиями. Видно, кто принял решение, какие сценарии сравнивались и где был рассинхрон между линией, сменами и логистикой — и как его устранили.

5.3. Отказ оборудования

Ситуация. Станок выходит из строя. Нужно решить: ремонт, замена узла, замена станка.

Роли станка:

РОЛЬ	ТИП СПЕЦИФИКАЦИИ	ВЛАДЕЛЕЦ
Актив с регламентом	process	ТОиР
Ресурс с производительностью	transformation	Производство
Основное средство	depreciation	Финансы
Источник риска	safety	ЕНС

Как работает логика:

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
1	Событие «отказ» фиксируется	1

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
2	Система сравнивает фактическое состояние с моделью надёжности → сигнал	4
3	Сигнал создаёт точку решения с владельцем — ТОиР	4
4	ТОиР сравнивает сценарии: ремонт, замена узла, замена станка	сравнение сценариев (часть 7.5)
5	Система ранжирует по стоимости владения, сроку, риску	сравнение сценариев (часть 7.5)
6	Решение фиксируется как управленческий акт	4
7	Все проекции отражают решение	5
8	Факт уточняет модель надёжности	7

Что становится видимым. Отказ — не «событие в ТОиР», а событие с последствиями для производства, финансов и безопасности. Видно, кто принял решение, на основании чего, и как факт уточнил модель надёжности — следующий прогноз будет точнее.

5.4. Недельное планирование производства

Ситуация. Планируем производство на неделю. Пять дней, пять заданий, параллельные ресурсы.

Как работает логика:

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
1	Ресурсы (линия, смены, персонал) лежат на оси событий	1
2	Заказ приходит на линию через спецификацию	2
3	Спецификация порождает плановые события: переналадка, простой	3
4	Система сравнивает параллельные ресурсы → сигнал рассинхрона	4
5	Система перестраивает график или запрашивает устранение	4
6	Факт фиксируется и ставит фронт	5

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
7	Будущее пересчитывается от фронта	5
8	Факт уточняет спецификацию	7

Что становится видимым. Не «план» и «факт» как две статические цифры, а **история**: что планировали, что скорректировали, что отменили, что ещё впереди. Видно, где был рассинхрон между линией и персоналом — и как его устранили. Видно, на каком уровне зрелости спецификация — и какой горизонт планирования она обеспечивает.

5.5. Расхождение по марже

Ситуация. Маржа по продукту в ERP, FP&A и BI показывает разные цифры. Нужно понять, откуда расхождение.

Как работает логика:

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
1	Система сравнивает три проекции маржи	сравнение проекций (часть 7.4)
2	Система показывает, какие события и методологии дали расхождение	сравнение с объяснением методологий (часть 7.4)
3	Сигнал «расхождение» создаёт точку решения	4
4	Владелец решения — финансовый директор — выбирает методологию	4
5	Все проекции отражают выбранную методологию	5

Что становится видимым. Не «три цифры», а «одна цифра с тремя методологиями». Видно, кто выбрал методологию и почему. Сегодня это знание исчезает после совещания.

5.6. Новый регуляторный отчёт

Ситуация. Регулятор требует новый отчёт — например, по углеродному следу.

Как работает логика:

ШАГ	ЧТО ПРОИСХОДИТ	ШАГ ЛОГИКИ
1	Добавляется новый тип спецификации — «углеродный след»	6
2	Существующие объекты (партия, станок, маршрут) связываются с новым контекстом	6
3	Спецификация порождает события «измерение выбросов»	3
4	Строится новая проекция «отчёт по выбросам»	6
5	Назначается владелец роли — «эколог»	6
6	По мере накопления фактов спецификация уточняется	7

Что становится видимым. Расширение не требует перестройки ландшафта. Оно требует добавления связей и назначения владельцев. Это управленческое действие, а не интеграционный проект.

Часть 6. Гибкость: как роли расширяются через спецификации

Это развёртывание шага 6. Ответ на вопрос: **что происходит, когда бизнес меняется?** Классический ландшафт отвечает на это покупкой новой системы и новым интеграционным проектом. Новый подход отвечает иначе: **новая спецификация порождает новые события и новые проекции, не трогая объект.**

6.1. Новое требование: углеродный след

ЧТО ДОБАВЛЯЕТСЯ	ЧТО НЕ МЕНЯЕТСЯ
Новый тип связи «углеродный след»	Объект «партия»
Новые события «измерение выбросов»	Объект «станок»
Новая проекция «отчёт по выбросам»	Объект «маршрут»
Новый владелец роли «эколог»	Существующие системы

6.2. Новый рынок: выход в другую страну

Добавляются спецификации «сертификация для рынка X», «налоговая классификация для страны Y». Они связывают существующие объекты с новым контекстом. Проекции для нового рынка строятся из того же потока событий. Выход на новый рынок не создаёт «второй экземпляр» объекта. Он добавляет роли.

6.3. Слияние или поглощение

Завод описывается как набор узлов и связей. Существующие объекты головной компании и завода связываются через спецификации — «общий поставщик», «общая номенклатура», «общий клиент». Где связи нет — объекты остаются отдельными. Где связь есть — роль одна, владелец один. Слияние не требует слияния ландшафтов. Оно требует описания связей.

6.4. Новое регуляторное требование: прослеживаемость

Прослеживаемость — не новая система. Это **новая проекция** существующего потока событий. События «партия получена», «партия потреблена», «партия отгружена» уже есть. Проекция «прослеживаемость» связывает их в цепочку. Ничего не покупается. Строится новый взгляд на существующие события.

6.5. Что это даёт

КЛАССИЧЕСКИЙ ЛАНДШАФТ	НОВЫЙ ПОДХОД
Новое требование → новая система	Новое требование → новая спецификация или проекция
Новый рынок → локализованная ERP	Новый рынок → новые роли существующих объектов
Слияние → интеграция ландшафтов	Слияние → описание связей между объектами
Новый регулятор → новая система	Новый регулятор → новая проекция существующих событий
Расширение = рост сложности	Расширение = рост числа ролей, не числа объектов

Главное: в логике модели гибкость достигается не заменой систем, а **расширением модели ролей и связей**. Объект остаётся один. Ролей становится больше. Владельцы ролей назначаются явно.

Часть 7. Планирование и анализ на линии событий

Это развёртывание шагов 1–5 и 7. Ответ на вопрос: **как в новом подходе работает планирование и как принимаются решения?**

В классическом ландшафте план — это документ в системе, а анализ — это отчёт. В новом подходе и план, и анализ — это **операции над линией событий**.

7.1. Планирование как создание будущих событий

Планирование — это не «заполнение формы». Это **создание событий в будущем** с явно заявленной модальностью.

МОДАЛЬНОСТЬ	ЧТО ОЗНАЧАЕТ	ПРИМЕР
Прогноз	Ожидание; уверенность ниже единицы	«Ожидаем спрос 1000 единиц в марте»
План	Намерение; уверенность ниже единицы	«Планируем произвести 900 единиц в марте»
Обязательство	Договорённость; уверенность ниже единицы, но высокая	«Обещали клиенту 800 единиц к 15 марта»
Факт	Произошло и зафиксировано; уверенность 1,0	«Произвели 850 единиц к 15 марта»
Корректировка	Изменение ранее заявленного	«План скорректирован до 870 единиц»
Отмена	Отказ от ранее заявленного	«Заказ отменён клиентом»

Прогноз, план, обязательство и факт — это не «одна цифра в разных системах», а **разные события на одной линии**, каждое со своей вероятностью и своим владельцем.

7.2. Пример: неделя производства

ДЕНЬ	СОБЫТИЕ	МОДАЛЬНОСТЬ	УВЕРЕННОСТЬ	ВЛАДЕЛЕЦ
Пн	Задание 1	план	0.9	Начальник цеха

ДЕНЬ	СОБЫТИЕ	МОДАЛЬНОСТЬ	УВЕРЕННОСТЬ	ВЛАДЕЛЕЦ
Пн	Переналадка	план	0.9	Начальник цеха
Вт	Задание 2	план	0.9	Начальник цеха
Ср	Задание 3	план	0.8	Начальник цеха
Ср	Плановый простой	план	0.8	Служба ТОиР
Чт	Задание 4	план	0.7	Начальник цеха
Пт	Задание 5	план	0.6	Начальник цеха

В понедельник задание 1 выполняется — создаётся событие «факт» с причинной связью с планом. Во вторник задание 2 корректируется. В среду задание 3 отменяется. К четвергу линия событий выглядит так:

ДЕНЬ	СОБЫТИЕ	ЧТО ПРОИЗОШЛО
Пн	Задание 1	План (0,9) → факт (1,0)
Пн	Переналадка	План (0,9) → факт (1,0)
Вт	Задание 2	План (0,9) → корректировка → факт (1,0)
Ср	Задание 3	План (0,8) → отмена
Ср	Плановый простой	План (0,8) → факт (1,0)
Чт	Задание 4	План (0,7), впереди фронта
Пт	Задание 5	План (0,6), впереди фронта

Вы видите не «план» и «факт» как две статические цифры, а **историю**: что планировали, что скорректировали, что отменили, что ещё впереди. И видите, кто владелец каждого события.

7.3. Параллельные ресурсы и рассинхрон

ДОРОЖКА	ЧТО НА НЕЙ	ВОЗМОЖНЫЙ РАССИНХРОН
Линия	Работа, переналадки, простой	Линия работает — персонала в смене нет
Смена А	Окно смены	Смена открыта — заданий нет
Персонал	Доступность людей	Персонал занят — линия простаивает
Ремонт	Окно ремонта	Линия в ремонте — смена и персонал запланированы

Система сравнивает дорожки. Если линия планирует работу, а персонала в смене нет — это рассинхрон. Система видит его и показывает. Дальше — развилка: перестроить график или запросить устранение. Это сравнение параллельных дорожек; подробнее о сравнении — в части 3.

7.4. Анализ как сравнение на линии событий

ЧТО СРАВНИВАЕМ	ЛЕВАЯ ТОЧКА	ПРАВАЯ ТОЧКА	ЧТО ПОЛУЧАЕМ
Факт vs план	Факт	План	Отклонение от плана
Факт vs норма	Факт	Норма	Отклонение от нормы
Факт vs модель	Факт	Прогноз модели	Точность модели
Сценарий А vs В	Сценарий А	Сценарий В	Ранжирование сценариев
Факт vs факт	Факт А	Факт В	Бенчмарк, аномалия

Анализ — это **сравнение**, у которого явно указаны: левая точка, правая точка, оператор, порог, владелец решения. Результат — **сигнал** (отклонение) или **предпочтение** (выбор между сценариями).

7.5. Сценарии как ветки на линии событий

ВЕТКА	ЧТО ЭТО
Основная	События, которые уже запланированы и исполняются
Сценарий А	Альтернативное продолжение от точки ветвления
Сценарий В	Второе альтернативное продолжение от той же точки

Сценарий А и сценарий В — не «два плана в двух системах». Это **две ветки** одного потока событий. У них разные вероятности, разные допущения, разные владельцы. Сравнение веток даёт **предпочтение** — ранжирование сценариев по критерию.

7.6. Зрелость спецификации и горизонт планирования

Уровни зрелости и связь с горизонтом описаны в шаге 7. Для планирования важно, что точность и горизонт не фиксированы, а зависят от зрелости спецификации. В логике модели спецификации уточняются эволюционно, а не «переобучаются»: факт даёт сигнал, сигнал уточняет спецификацию, следующий план точнее. И это видно: вы знаете, на каком уровне зрелости находится каждая спецификация, и какой горизонт планирования она обеспечивает.

7.7. Что это даёт для планирования и анализа

КЛАССИЧЕСКИЙ ЛАНДШАФТ	НОВЫЙ ПОДХОД
План — документ в системе	План — событие в будущем с модальностью
Факт перезаписывает план	Факт — новое событие, план сохраняется
Анализ — отчёт	Анализ — сравнение двух точек на линии
Сценарии — файлы в Excel	Сценарии — ветки на линии событий
История планов исчезает	История планов сохраняется в потоке событий
Решение — совещание	Решение — управленческий акт с причинной связью
Расхождение — ошибка	Расхождение — разные проекции одного потока
Точность плана — «как получилось»	Точность плана — от зрелости спецификации

Часть 8. Чем это отличается от известных подходов

Логика не уникальна в каждой детали. Она пересекается с несколькими известными подходами — и важно понимать, где именно.

ПОДХОД	ЧТО ОБЩЕГО	ЧЕМ ОТЛИЧАЕТСЯ
REA (Resource-Event-Agent, McCarthy, 1982)	Событие первично; ресурсы и агенты — участники события	REA — онтология учёта, описывающая экономический обмен. Здесь шире: роли, спецификации, зрелость правил и фронтير планирования; управленческий акт — тоже событие.
Event sourcing	Поток событий как источник истины; проекции строятся из журнала	Event sourcing — технический паттерн реализации. Он не отвечает, кто владелец роли, какова зрелость правила и где граница знания. Без этих ответов журнал событий воспроизводит тот же хаос, только в одном месте.
DDD (Domain-Driven Design)	Явные границы контекстов, модель предметной области	DDD допускает разные модели в разных bounded contexts и не требует единого объекта. Здесь — попытка удержать один объект с явными ролями; DDD полезен как дисциплина границ, но не как ответ на стык.
ISA-95	Иерархия уровней управления производством (0–4), интеграция ERP–MES	ISA-95 описывает уровни и интерфейсы между ними; объект и событие в нём вторичны. Здесь объект и событие первичны, а уровни становятся проекциями. ISA-95 остаётся полезным языком для описания существующего контура.

Практический вывод. Если вы уже используете event sourcing или DDD, эта логика не отменяет их. Она добавляет то, чего в них нет: явную зрелость правил, фронтир знания и владельца решения. Если вы используете ISA-95, она не отменяет уровни, а предлагает смотреть на них как на проекции одного потока событий.

Часть 9. Ограничения

Это не замена ERP завтра. Существующие системы остаются. Меняется не ландшафт, а слой над ним.

Это не серебряная пуля. Модель ролей и спецификаций нужно описать. Это сложно. Это требует времени и людей, которые понимают и предметную область, и моделирование.

Это не продукт. Это принцип. Продукты могут быть построены на этом принципе, но сам принцип продуктом не является.

Это не решает организационные вопросы. Кто владелец роли — это организационное решение, а не техническое. Модель может показать, что владельца нет. Но назначить его — не её задача.

Это требует смены мышления, а не смены вендора. Замена SAP на 1С или наоборот ничего не изменит. Изменится инструмент, не подход.

Это может конфликтовать с текущими KPI. Если ИТ-отдел оценивается по числу внедрённых систем, а не по согласованности данных, у него нет стимула менять подход.

Это не отменяет необходимости в данных. Модель работает на событиях. Если события не фиксируются — модель не работает. Точность модели зависит от полноты и качества событий.

Режимы отказа

У логики есть предсказуемые способы сломаться. Их лучше знать до внедрения.

РЕЖИМ ОТКАЗА	КАК ПРОЯВЛЯЕТСЯ	ЧТО ДЕЛАТЬ
Версионирование событий	Старое событие нельзя прочитать новой логикой; проекции расходятся	Хранить версию схемы события; миграция — явная операция с журналом
Идемпотентность	Повторная запись (ретрай, двойной скан) создаёт дубль	Естественный ключ события и проверка на дубль при записи
Дубли	Одно физическое событие зафиксировано дважды	Дедупликация по ключу; разбор расхождения как инцидент качества данных
Часовые пояса	Событие «произошло» и «зарегистрировано» в разное время; переход на летнее время ломает смены	Хранить UTC и событийное время отдельно от времени записи
Качество мастер-данных	Событие ссылается на объект, которого нет или он «переехал»	Справочник как отдельный контур качества; валидация ссылки при записи
Стоимость журнала	Журнал растёт, индексы дорожают, проекции замедляются	Политика хранения, архивирование, бюджеты на

РЕЖИМ ОТКАЗА	КАК ПРОЯВЛЯЕТСЯ	ЧТО ДЕЛАТЬ
		хранение
Совместимость схем	Потребитель проекции ломается при изменении события	Контракт события и обратная совместимость; версия — часть контракта
Приватность	Журнал хранит всё навсегда, включая персональные данные	Классификация данных и правила удаления или анонимизации

Часть 10. Что можно сделать уже сейчас

Даже не строя новую платформу, можно начать с **явного описания ролей, спецификаций и линии событий**.

Выбрать один объект — сырьё, оборудование, сотрудник, заказ, проект, инцидент.

Описать, какие роли он играет в каждой системе.

Описать спецификации — какие правила связывают объект с каждым контекстом.

Описать, какие события порождает каждая спецификация — переналадка, простой, контроль.

Описать, кто владелец каждой роли.

Построить линию событий — какие события происходят с объектом, какие были планом, какие фактом, какие корректировкой.

Найти параллельные ресурсы — где возможен рассинхрон.

Оценить зрелость спецификаций — на глазок, норматив, расчёт, статистика.

Найти, где роли пересекаются и где данные расходятся.

Задать вопрос: что произойдёт, если роли будут описаны явно? Изменится ли что-то в принятии решений?

Это не требует покупки новой системы. Это требует **разговора**, которого обычно не происходит. Классический ландшафт не предполагает такого разговора — он предполагает, что каждая система «лучшая в своём классе», а стыки — «проблема интеграции».

Чек-лист перед выбором подхода

- [] Назван один объект, на котором проверяется логика.
- [] Для него выписаны все роли в существующих системах.
- [] Для каждой роли указан владелец.
- [] Для каждой роли указано правило связи и порождаемые им события.
- [] Для каждого правила оценена зрелость.
- [] Определено, где проходит фронтир знания.

- [] Названы параллельные ресурсы и возможные рассинхроны.
- [] Решение о внедрении принимается как гипотеза с критерием проверки, а не как «единственно верная архитектура».

Заключение

«Цифровизация 2.0» — не продукт и не призыв заменить ERP. Это гипотеза о том, что порядок возникает не из большого числа систем, а из явного описания объекта, его ролей, правил связи и границы знания. Проверить её можно на одном объекте и одном процессе, не покупая платформу: если после описания ролей и владельцев становится видно, кто принимает решение и почему, гипотеза работает. Если нет — вы получили полезную карту стыков и сэкономили бюджет.

Источники

McCarthy, W. E. The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment // The Accounting Review. — 1982. — Vol. 57, No. 3.

Evans, E. Domain-Driven Design: Tackling Complexity in the Heart of Software. — Addison-Wesley, 2003.

Fowler, M. Event Sourcing. — 2005.

ISA-95 / IEC 62264. Enterprise-control system integration.

Читайте также

Цифровизация 1.0 (</articles/digital/cifrovizaciya-1-0/>). — атлас 22 стыков классического промышленного ИТ.

ИИ-зоопарк (</articles/digital/ii-zoopark/>). — теневой ИИ и три модели организации функции.

Нервная система компании (</weapons/nervnaya-sistema/>). — информационные потоки и обратная связь на производстве.

Геном завода (</genome/>). — типизированная библиотека процессов.

УПРАВЛЕНИЕ СПОСОБНОСТЬЮ

Серия о том, как компания подменяет способность её имитацией.

Throughput на час узкого места, TDABC и стоимость сложности — почему средняя себестоимость ошибается

🕒 10 мин чтения

ЭКОНОМИКА

Почему инвестиции в качество управления не проходят инвестиционный комитет — и что с этим делать CEO и CFO

🕒 18 мин чтения

УПРАВЛЕНИЕ

Структурная ловушка, запас устойчивости и онтология понятий «развитие», «оптимизация», «антикризис»

🕒 20 мин чтения

ЦИФРОВИЗАЦИЯ

Атлас 22 стыков промышленного ИТ, три иллюзии и протокол диагностики

🕒 31 мин чтения

ЦИФРОВИЗАЦИЯ

Теневой ИИ, абсорбционная способность и три модели организации функции

🕒 22 мин чтения

ЧАСТЫЕ ВОПРОСЫ

Что такое спецификация в этом подходе?

Это правило связи, а не документ и не атрибут объекта. Оно отвечает на вопрос, как объект связан с другим объектом или контекстом и какие события из этой связи следуют.

Технологическая спецификация — лишь один из видов такого правила.

Что такое фронтир знания?

Это граница между зафиксированным фактом и предположением. Слева от фронта — события, которые уже произошли и зарегистрированы; справа — план, прогноз или обязательство. Когда регистрируется новый факт, фронтир сдвигается, и будущее пересчитывается от новой границы.

Чем подход отличается от event sourcing?

Event sourcing — технический паттерн: журнал событий и проекции. Подход в статье — управленческая логика: он добавляет явную зрелость правил, границу знания и владельца решения. Без этих ответов журнал событий воспроизводит тот же хаос, только в одном месте.

Это готовая архитектура для внедрения?

Нет. Это гипотеза проектирования, а не продукт и не доказанная теория. Её можно проверить на одном объекте и одном процессе, не покупая платформу: описать роли, правила связи и владельцев и посмотреть, становится ли видно, кто принимает решение и почему.

МАТЕРИАЛЫ ПО ТЕМЕ

Статья целиком и рабочие артефакты — для скачивания и печати.



**Статья «Цифровизация 2.0: логика описания системы
через объект, роли и события»**

PDF • полный текст

СКАЧАТЬ

Разобрать вашу ситуацию

Опишите свой контур и ограничения — обсудим, где узкое место и что делать с экономикой, развитием и цифровым контуром.

НАПИСАТЬ В TELEGRAM

