

Владимир Репин

# **Моделирование бизнес- процессов в нотации BPMN**

Пособие для начинающих.  
Часть I

Издательские решения  
По лицензии Ridero  
2019

УДК 33  
ББК 65  
Р41

Шрифты предоставлены компанией «ПараТайп»

- Репин Владимир**  
Р41    Моделирование бизнес-процессов  
в нотации BPMN : Пособие для начинающих. Часть  
I / Владимир Репин. — [б. м.] : Издательские  
решения, 2019. — 84 с.  
ISBN 978-5-4496-6989-6 (т. 1)  
ISBN 978-5-4496-6990-2

В книге рассматривается нотация BPMN, предназначенная для проектирования исполняемых бизнес-процессов. Излагается минимально необходимый объем знаний для сотрудников, которые впервые приступают к описанию процессов. Выполняя практические задания, представленные в книге, вы сможете за один день освоить нотацию BPMN на начальном уровне.

УДК 33  
ББК 65

16+ В соответствии с ФЗ от 29.12.2010 №436-ФЗ

ISBN 978-5-4496-6989-6  
ISBN 978-5-4496-6990-2

© Владимир Репин, 2019

# Оглавление

1. Введение. Что такое нотация BPMN?	5
1.1. Нотация BPMN	5
1.2. Процесс и его контекст	6
1.3. Цель, точка зрения и метод	8
2. Графическая схема процесса. Основы	10
2.1. Пул, дорожки, роли, должности	10
2.2. События. Запуск и остановка процесса	12
2.3. Операции процесса и стрелки	13
3. Логика процесса	16
3.1. Шлюз исключаяющее «ИЛИ»	16
3.2. Шлюз «И»	18
3.3. Типовые примеры. Логические ошибки	22
3.4. Некорректное использование шлюзов	24
3.5. Шлюзы для старта процесса	24
3.6. Головоломная задача	26
4. Движение документов на схеме процесса	28
4.1. Движение документов внутри процесса	28
4.2. Поток документов между процессами	30
4.3. Разница между потоком работ и потоком документов	33
5. Межпроцессное взаимодействие	34
5.1. Понятие экземпляра процесса	34
5.2. Старт процесса по событию получения сообщения	35
5.3. Завершение процесса событием отправки сообщения	36
5.4. Промежуточные события отправки и получения сообщений	37
5.5. Типовые ошибки при использовании событий отправки и получения сообщений	40
6. Использование промежуточных событий-таймеров	42
7. Граничные события. Таймер. Эскалация. Ошибка	44
7.1. Граничные события-таймеры	44
7.2. Граничное событие-эскалация	45
7.3. Граничное событие-системная ошибка	46
8. Сложные шлюзы в процессе	48
9. Декомпозиция процессов	51
10. Описание сквозных процессов	52
11. Анализ качества графических схем процессов	56
11.1. Формальный анализ качества схемы	56
11.2. Содержательный анализ схемы процесса	63

12. Комплексный пример описания группы процессов .....	67
13. Организация работы по описанию процесса .....	71
14. Что почитать по BPMN и процессному подходу? .....	75
15. Правильные ответы на задачки .....	76
16. Приложение №1. Основные элементы нотации BPMN .....	78
16.1. Основные элементы нотации BPMN .....	78
16.2. Примеры использования элементов нотации BPMN .....	79

# 1. Введение. Что такое нотация BPMN?

## 1.1. Нотация BPMN

Вы держите перед собой книгу — краткое методическое пособие по освоению нотации BPMN для целей практического описания и анализа бизнес-процессов компании.

В нем излагается минимально необходимый объем знаний по BPMN для сотрудников, которые *впервые приступают к описанию процессов*. Надо четко понимать, что освоение данного материала не сделает вас профессионалом по BPMN, но позволит сделать первые шаги на пути освоения методики моделирования.

Книга не предназначена для повышения квалификации профессионалов в области моделирования и автоматизации бизнес-процессов. Для практиков, имеющих большой опыт автоматизации процессов, или теоретиков BPMN предлагаемые подходы к изложению темы могут показаться чрезмерно упрощенными. Но хочу еще раз подчеркнуть, что целевая аудитория книги — сотрудники, которые только что приступили к изучению методов моделирования процессов.

Нотация — это набор графических обозначений (значки, стрелки, пиктограммы), которые позволят создать графическую схему — модель процесса, понятную другим людям. Конечно, можно описать процесс текстом, но графическая схема гораздо удобнее для быстрого понимания выполняемого процесса.

BPMN (Business Process Model and Notation — нотация и модель бизнес-процессов) разработана компанией Business Process Management Initiative и поддерживается Object Management Group после слияния организаций в 2005 г. Последняя версия 2.0 вышла в 2012 г. В 20013 году BPMN *утверждена в качестве международного стандарта ISO/IEC 19510*.

В настоящее время большинство компаний, поставляющих системы автоматизации бизнес-процессов — BPMS, используют нотацию BPMN для проектирования исполняемых процессов.

Фактически, BPMN сегодня — это лучшая, признанная на международном уровне и активно используемая многими компаниями нотация. Но она, как любая другая нотация, имеет ограничения применимости. Например, для формирования архитектуры процессов компании (т.н. модели верхнего уровня) лучше использовать нотацию IDEF0.

Применение BPMN в вашей компании даст возможность описать процессы в стандартном виде и использовать полученные модели для целей анализа и улучшения, регламентации и стандартизации процессов. В дальнейшем, совершенствуя свои навыки моделирования, вы сможете использовать BPMN для целей автоматизации бизнес-процессов.

Рекомендую последовательно выполнять все задания без исключения, даже если что-то уже покажется вам знакомым и слишком простым. Для выполнения заданий можно использовать MS Visio, Business Studio, Bizagi BPM Modeler или любую другую программу, которая поддерживает создание графических схем в нотации BPMN.

В конце книги вы сможете найти ссылки на дополнительные материалы для более глубокого освоения темы моделирования и процессного управления в целом.

Схемы процессов для книги подготовлены в среде моделирования Business Studio 4. Вид некоторых графических объектов может слегка отличаться от используемых в других программах, поддерживающих моделирование в нотации BPMN.

## 1.2. Процесс и его контекст

Обратите внимание на определение процесса (бизнес-процесса):

*Процесс — устойчивая, целенаправленная совокупность взаимосвязанных видов деятельности, которая по определенной технологии преобразует входы в выходы, представляющие ценность для потребителя.*

Входы и выходы — это информационные и материальные потоки.

Прежде чем описывать процесс в виде графической схемы, очень важно определить его контекст — см. рис. 1.



Рис. 1. Правильный контекст процесса.

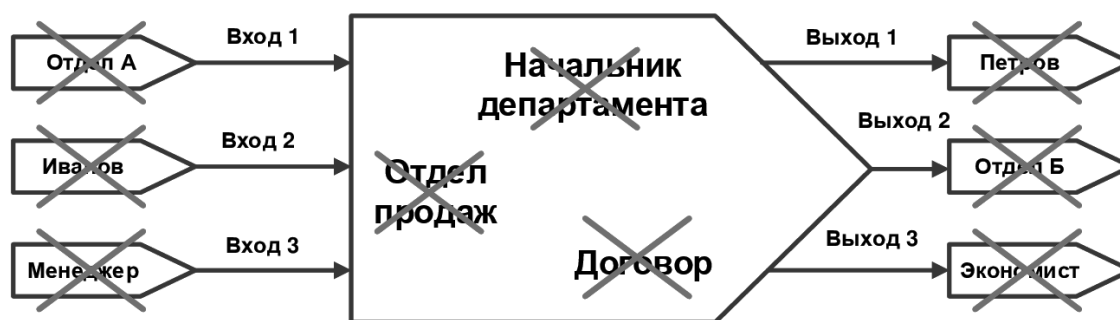


Рис.2. Неправильный контекст процесса.

Запомните принцип: процесс *может*:

- получать входы от других процессов;
- передавать выходы другим процессам.

Процесс *НЕ может*:

- получать входы от отделов, сотрудников, физических лиц и прочих сущностей, кроме процессов;
- передавать выходы другим отделам, сотрудникам, физическим лицам и прочим сущностям, кроме процессов.

Почему это так? Процесс — это не сферический конь в вакууме. Он существует среди других процессов в общей архитектуре процессов организации. Если вы будете рисовать на схеме взаимодействие процесса с «Ивановым» или «Отделом закупок», то вместо системы взаимодействующих процессов получите управленческий винегрет.

Для успешного моделирования *крайне важно представить организацию как систему взаимодействующих бизнес-процессов*. Понятно, что в организации есть процессы, которые вообще не связаны между собой.

Будет здорово, если вы также выявите управляющие воздействия (планы, приказы, распоряжения) и нормативные требования к процессу (ГОСТы, регламенты, положения и т.п.). Это важно для анализа контекста. Но управляющие воздействия не всегда целесообразно показывать на схеме процесса.

Затем нужно указать состав участников процесса, но об этом несколько позже.

Кроме входов и выходов очень важно идентифицировать события, инициирующие процесс, и события, завершающие процесс. Например, стартовое событие — «Получена заявка на выставление счета», завершающее событие — «Выставлен счет на оплату».

События отличаются от входов/выходов. Они определяют условия запуска (продолжения, завершения) процесса.

Если начать описание процесса без определения его контекста, границ и событий, то с вероятностью 99% вы нарисуете не то, что нужно. Работу придется переделывать несколько раз.

### 1.3. Цель, точка зрения и метод

Перед тем как описывать процесс, важно определиться с тремя вопросами:

- цель;
- точка зрения;
- метод.

Определите цель. Для чего вам нужно описание процесса? Если читателями процесса будут только люди, то схема может быть простой, построенной с использованием минимального количества графических элементов.

Если вы планируете описать только функции, которые выполняются в какой-то информационной системе (например, документооборот), — это одно. Если цель — получить, проанализировать и использовать для регламентации схему реального бизнес-процесса со всеми его особенностями — это другое. Можно описать процесс широкими мазками для общего представления, а можно детально, с учетом всех нюансов.

Пример укрупненного представления процесса открытия депозита в банке:

1. прийти в банк;
2. открыть депозит;
3. уйти из банка.

Достаточно ли такого описания для решения практических задач? Смотря каких...

Определите точку зрения (как правило, — это точка зрения бизнес-заказчика модели). Ее выбор существенно повлияет на результат. Приведу пример. Как будет выглядеть процесс открытия банковского депозита в банке?

С точки зрения клиента последовательность такая:

- прийти в банк;
- получить талончик в электронной очереди;
- дождаться своей очереди (увы, операция ожидания — это тоже часть процесса);



- объяснить пожелания сотруднику банка, передать паспорт;
- подписать договор;
- перейти в кассу и внести деньги;
- получить квитанцию о внесении средств;
- покинуть банк.

Тот же процесс с точки зрения сотрудника банка (операциониста):

- выяснить потребность клиента;
- проверить паспорт;
- оформить договор и сберкнижку;
- оформить депозит в системе;
- выдать бирку на внесение денег в кассе, передать документы кассиру.

А каким будет процесс оформления депозита с точки зрения председателя совета директоров банка? Нам придется описать взаимодействие всех участников: клиента, операциониста, кассира и, кроме того, информационной банковской системы (она тоже может рассматриваться в качестве участника процесса). Дело в том, что руководителю важно увидеть картину сквозного процесса в целом, чтобы иметь возможность его целенаправленно улучшать.

Определите метод описания. Можно, например, вместо процесса просто описать функции подразделений. Но тогда получится структура функций, а не процессы (тем более, сквозные).

Можно описать движение документов между операциями (шагами) процесса. Но тогда получится не исполняемый процесс, а схема информационных потоков между операциями процесса. Это, в частности, отличает процесс от документооборота.

Для целей анализа времени выполнения процесса и подготовки к автоматизации необходимо описывать реальный *поток работ (Work Flow)*, причем с использованием понятия «Токен» (можно визуально представить себе зверька, бегущего по стрелкам от одной операции процесса к другой и передающего управление — запускающего операции процесса на выполнение).

Замечу, что если вы приступаете к описанию процессов на самом верхнем уровне, то сначала крайне важно понять цепочку создания ценности организации и построить структурную модель, например, в нотации IDEF0. В данной книге метод построения структурных моделей не рассматривается.

## 2. Графическая схема процесса. Основы

*Пулы. Дорожки. Исполнители (роли, должности). События. Запуск и остановка процесса. Правила именования событий. Описание операций процесса на схеме. Правила именования операций. Пример схемы простого линейного процесса.*

### 2.1. Пул, дорожки, роли, должности

Итак, приступаем к созданию графической схемы процесса. На рис. 3 представлен пул (Pool) — это просто рамка, внутри которой будет описан процесс. Все, что вне пула, — это внешнее окружение. Для изображения взаимодействующих процессов в BPMN принято отображать их на диаграмме в виде «черных ящиков» (свернутых пулов).

Внутри рамки (пула) созданы три дорожки (Lane). Дорожки принято называть в терминах исполнителей процесса. Ими могут быть:

- должности;
- роли.

Например, «Начальник отдела продаж» — это должность. «Инициатор договора» — это роль. «Начальник отдела» — это тоже... роль, т.к. не указано, какого именно отдела начальник.

Роли лучше назвать контекстно. Это означает, что нежелательно называть роль просто «Ответственный», а нужно, например, «Сотрудник, ответственный за подготовку отчета о продажах».

Недопустимо называть дорожки по фамилии исполнителя.

Дорожки на схемах BPMN принято располагать горизонтально, хотя вертикальное расположение также допустимо.

Процесс подготовки Документа			
Начальник отдела подготовки документов	Специалист по проверке документов ОПД	Сотрудник, ответственный за подготовку Документа	

Рис. 3. Пул и дорожки.

## 2.2. События. Запуск и остановка процесса

День рождения — это событие? Еще какое! Вообще, в нашей жизни все начинается с событий. Так и на схемах процессов нужны стартовые события. Первый и самый простой тип стартового события — неопределенное событие (см. рис. 4).



Рис. 4. Стартовое и завершающее события.

При моделировании в Business Studio инициирующие процесс события — зеленые, завершающие процесс события — красные, промежуточные события (будут рассмотрены ниже) — оранжевые. Использование цвета повышает визуальную наглядность схем, но не является обязательным.

Неопределенный тип событий используется, когда мы описываем абстрактный процесс или при декомпозиции конкретного процесса на нижний уровень.

В реальной же ситуации стартовые события могут возникать в следующих случаях:

- наступление определенного времени;
- получение важной информации;
- исполнение некоторого условия.

У процесса может быть несколько стартовых событий. Как быть в этом случае, я расскажу чуть ниже.

Событие, завершающее процесс, также должно быть показано на схеме. У процесса может быть несколько разных завершающих событий. Это нормально.

На рис. 4 показано событие с черным кружком в середине — это завершающее процесс событие типа termination. Если в конце одной из веток процесса возникает такое событие, то все выполняемые в данный

момент ветки процесса будут остановлены. В некоторых случаях это необходимо.

### 2.3. Операции процесса и стрелки

На рис.5 показана схема простого линейного процесса. Процесс начинается со стартового события-таймера «Ежедневно, 10—00». Внутри зеленого кружка показана пиктограмма часов. Такого рода *пиктограммы в BPMN принято называть маркерами*. Наличие маркера может существенно изменить смысл объекта на схеме процесса. В BPMN используется много разных типов маркеров. Хорошо то, что для начала можно обойтись их малым количеством.

С содержательной точки зрения стартовые события-таймеры могут быть абсолютные и относительные. Примеры. «23 февраля 2018 года, в 17—00», «Ежедневно, в 9—00» — абсолютные таймеры. «Через 2 часа после начала термической обработки» — относительный таймер. Некорректно именовать стартовые события-таймеры, например, так: «Наступило утро» или «Не позднее второй половины дня».

Вернемся к рис.5. Четырехугольники со скругленными краями на схеме — это шаги процесса. Их можно называть действиями, функциями, задачами, но лучше — операциями процесса (в нотации BPMN они называются Task — задача).

Принято именовать операции процесса глаголами, например: «Выполнить...», «Подготовить...», «Рассчитать...» и т. п. Категорически нельзя называть операции процесса так: «Заявка», «Договор», «Коммерческий отдел», т.е. использовать названия документов, отделов и т. п.

Нежелательно в названии процесса указывать роли или должности других исполнителей, например: «Передать документ Начальнику Коммерческого отдела». Название должности в модели организационной структуры<sup>1</sup> может измениться, а в названии операции процесса этот факт не отобразится.

Общее количество операций на схеме процесса целесообразно ограничивать — до 12—15. Общий критерий — схема должна нормально смотреться и легко читаться человеком с нормальным зрением при распечатке на листе формата А4. Конечно, если для ознакомления со схемой процесса используется web-страница<sup>2</sup> и 24-дюймовый монитор, то это ограничение становится не таким жестким.

---

<sup>1</sup> При использовании Business Studio.

<sup>2</sup> Просмотр моделей процессов в BS Portal.

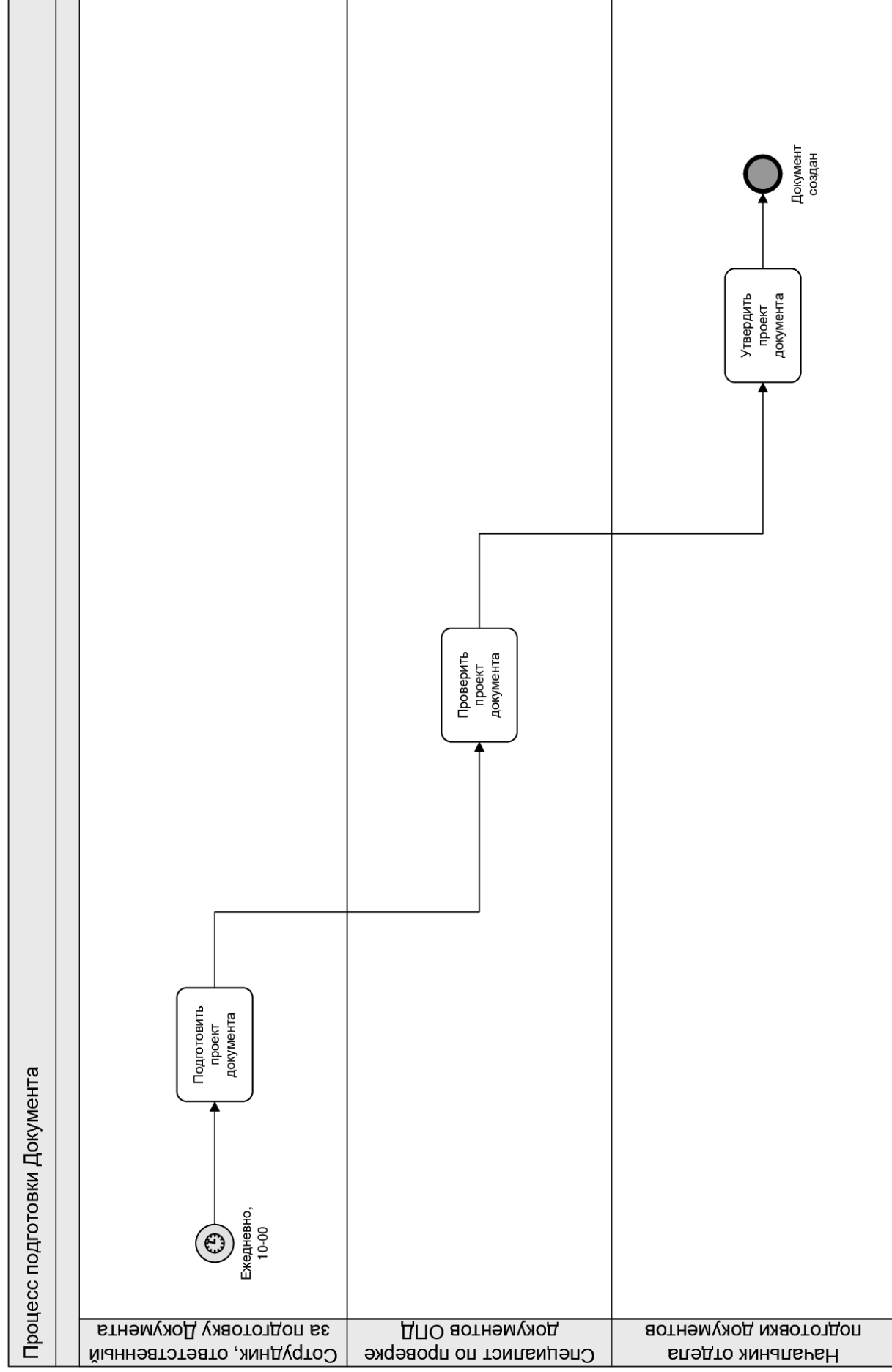


Рис. 5. Схема простого, линейного процесса.

Как быть, если схема не умещается на один лист А4? Агрегировать шаги процесса, а потом создавать модели нескольких подпроцессов, взаимодействующих между собой.

Операции процесса на схеме соединены стрелками. Эти стрелки имеют тип «Sequence flow» — они показывают последовательность выполнения операций во времени. Можно сказать, что они управляют «поток операций» — Work Flow.

Хотя нотация BPMN допускает ситуацию, когда в одну операцию процесса одновременно входит несколько стрелок (выходит несколько стрелок) типа sequence, рисование таких схем может запутать неопытного пользователя. Поэтому я рекомендую взять на вооружение и использовать следующее правило: «У операции может быть только одна стрелка запуска и одна стрелка продолжения». Данный принцип проиллюстрирован на рис. 6:

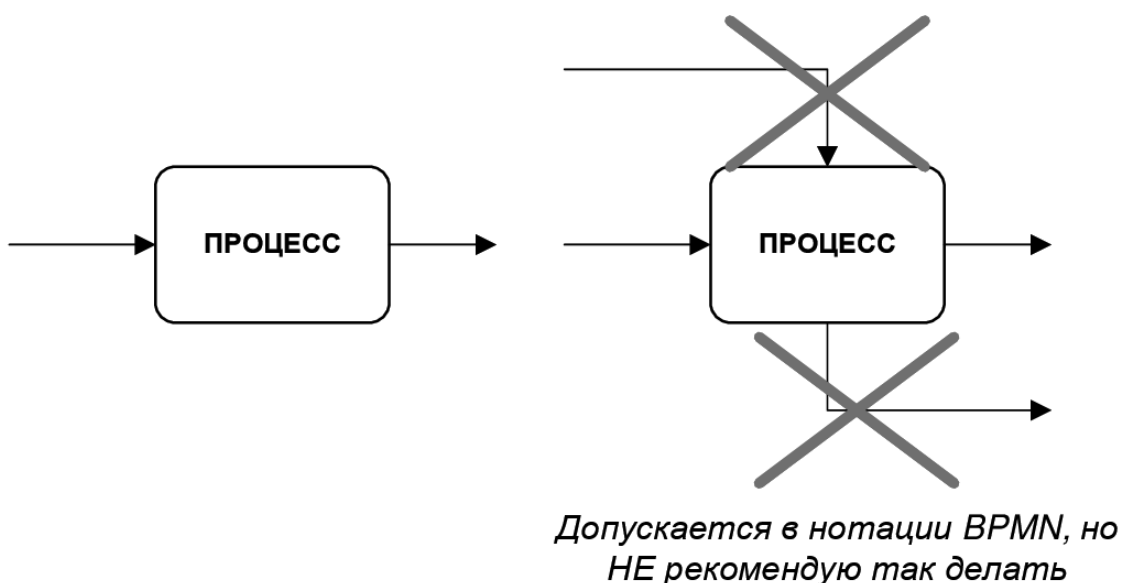


Рис. 6. Правило двух стрелок.

Итак, на рис. 5 представлен простой линейный процесс. На практике процессы редко бывают линейными. По разным причинам возникают возвраты на предыдущие операции процесса. Как быть в этом случае? Об этом — в следующем разделе.

### 3. Логика процесса

*Операторы логики (шлюзы). Шлюз исключающее «ИЛИ». Как правильно показывать возвраты. Шлюз «И». Типовые примеры. Логические ошибки. Шлюзы для старта процесса. Хитрые шлюзы. Головоломная задача.*

#### 3.1. Шлюз исключающее «ИЛИ»

Может ли сотрудник, ответственный за подготовку документа, ошибиться? Да. Может ли Специалист по проверке документов пропустить эту ошибку, а начальник выявить? Тоже да. Конечно, такой процесс эффективным назвать нельзя. Он явно нуждается в оптимизации. Но сначала *нужно корректно отобразить на схеме ситуацию «как есть»*, т.е. показать возвраты и переделки предыдущих операций. На рис. 7 показан такой процесс.

Желтый, поставленный на ребро квадратик с косым крестиком внутри — это так называемый эксклюзивный шлюз (развилка  $\equiv$  маршрутизатор  $\equiv$  элемент логики). В данном случае использован шлюз «Исключающее ИЛИ». Он показывает, что после выполнения операции процесс может пойти по нескольким альтернативным веткам. Количество исходящих потоков из шлюза «ИЛИ» нотацией BPMN не ограничено.

Например, после выполнения операции «Проверить проект документа» может быть две ситуации: 1) «Документ проверен. Ошибок нет» и 2) «Выявлены ошибки в документе». Во втором случае возникает возврат и переделка операции «Подготовить проект документа».

«Почему нельзя рисовать ветвления процесса и возвраты безо всяких там шлюзов?» — вопрос, которым задается обычное рабочее-крестьянское сознание. Да, можно рисовать как попало, как душе угодно. Только это уже будет не BPMN, не инженерный подход к проектированию процессов, а свободный полет фантазии на тему... И да, потом схему понять не сможет никто, кроме автора.



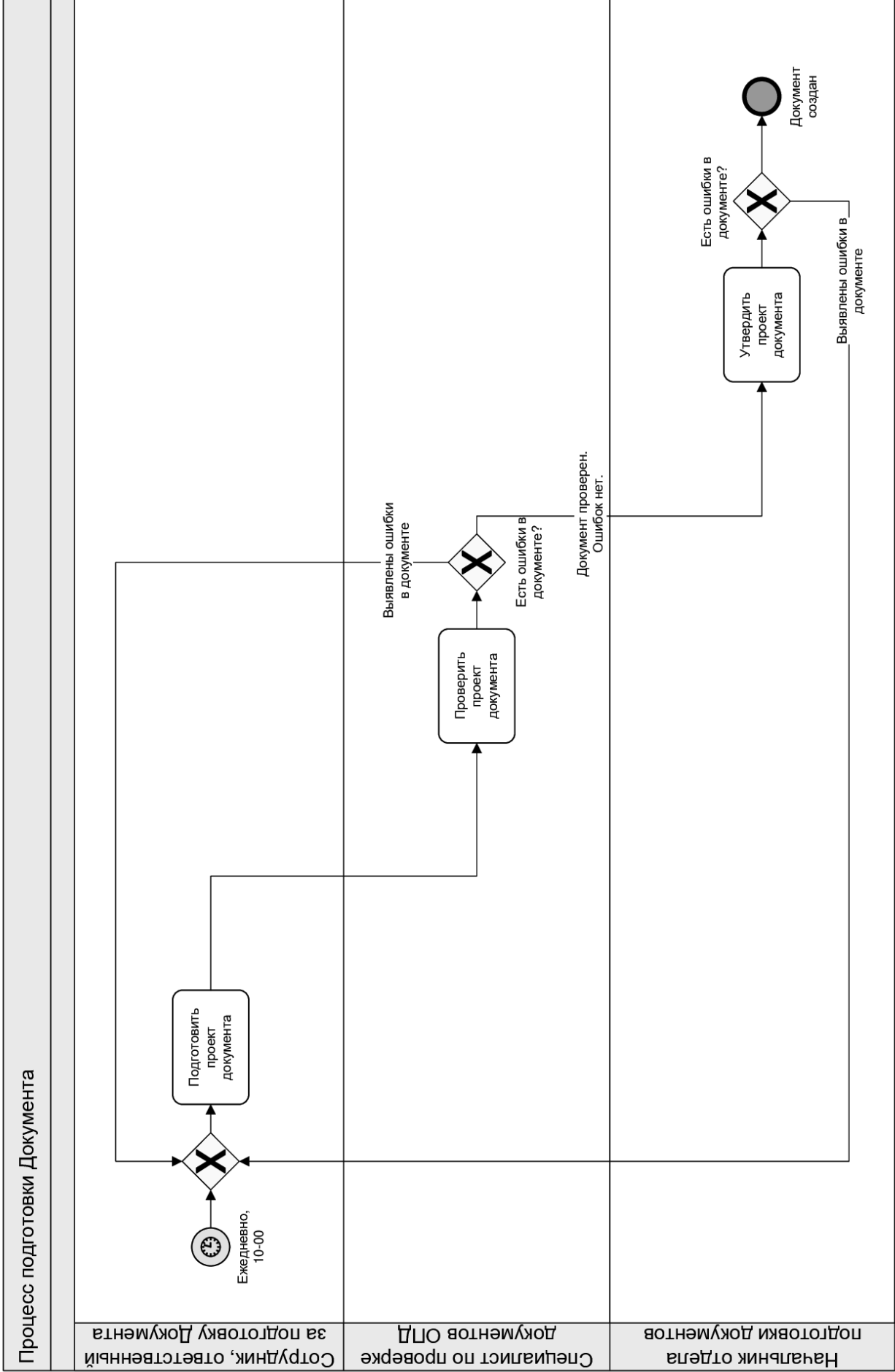


Рис. 7. Использование шлюза «Исключающее ИЛИ».

На рис. 8 показан фрагмент схемы с возвратом. Нотация BPMN допускает такой возврат, но я рекомендую использовать «Правило двух стрелок» (см. выше). С точки зрения этого правила возврат, представленный на рис. 8 является некорректным.

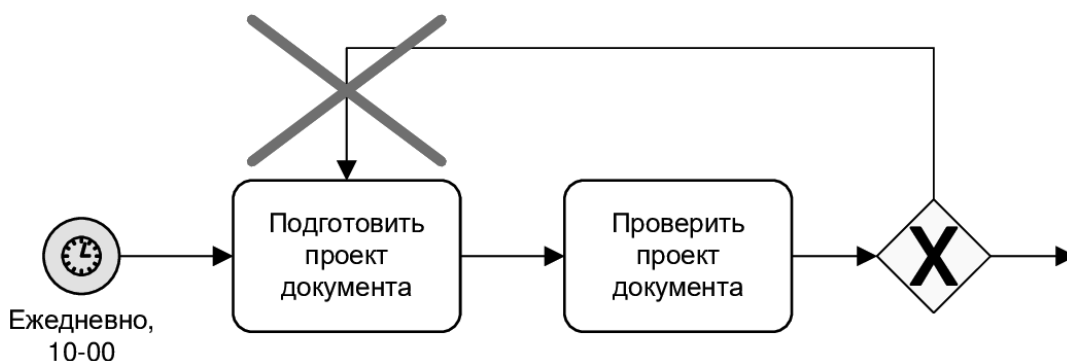


Рис. 8. Нежелательный вариант отображения возврата на схеме процесса.

Шлюзы типа «Исключающее ИЛИ» могут не только разделять потоки работ, но и объединять их. На рис. 7 первый шлюз показывает, что мы можем приступить к выполнению операции «Подготовить проект документа» либо сразу после начала выполнения процесса, либо вернуться после выполнения одной из двух других операций.

Обратите внимание, что шлюзы можно подписывать. Удобно формулировать вопрос, в зависимости от ответа на который возможны различные альтернативные ветки процесса. Кстати, стрелки с этими альтернативными ветками так же желательно подписывать. Для компьютера это всё равно, а вот для человека схема становится существенно более информативной и удобной в работе.

И последнее. Если внутри шлюза нет никакого маркера, то это тоже шлюз «Исключающее ИЛИ».

### 3.2. Шлюз «И»

На рис. 9 представлена более сложная схема.

На ней показано, что два сотрудника одновременно выполняют расчеты для разных разделов документа. Затем первый сотрудник включает расчеты в документ и передает его второму на проверку.

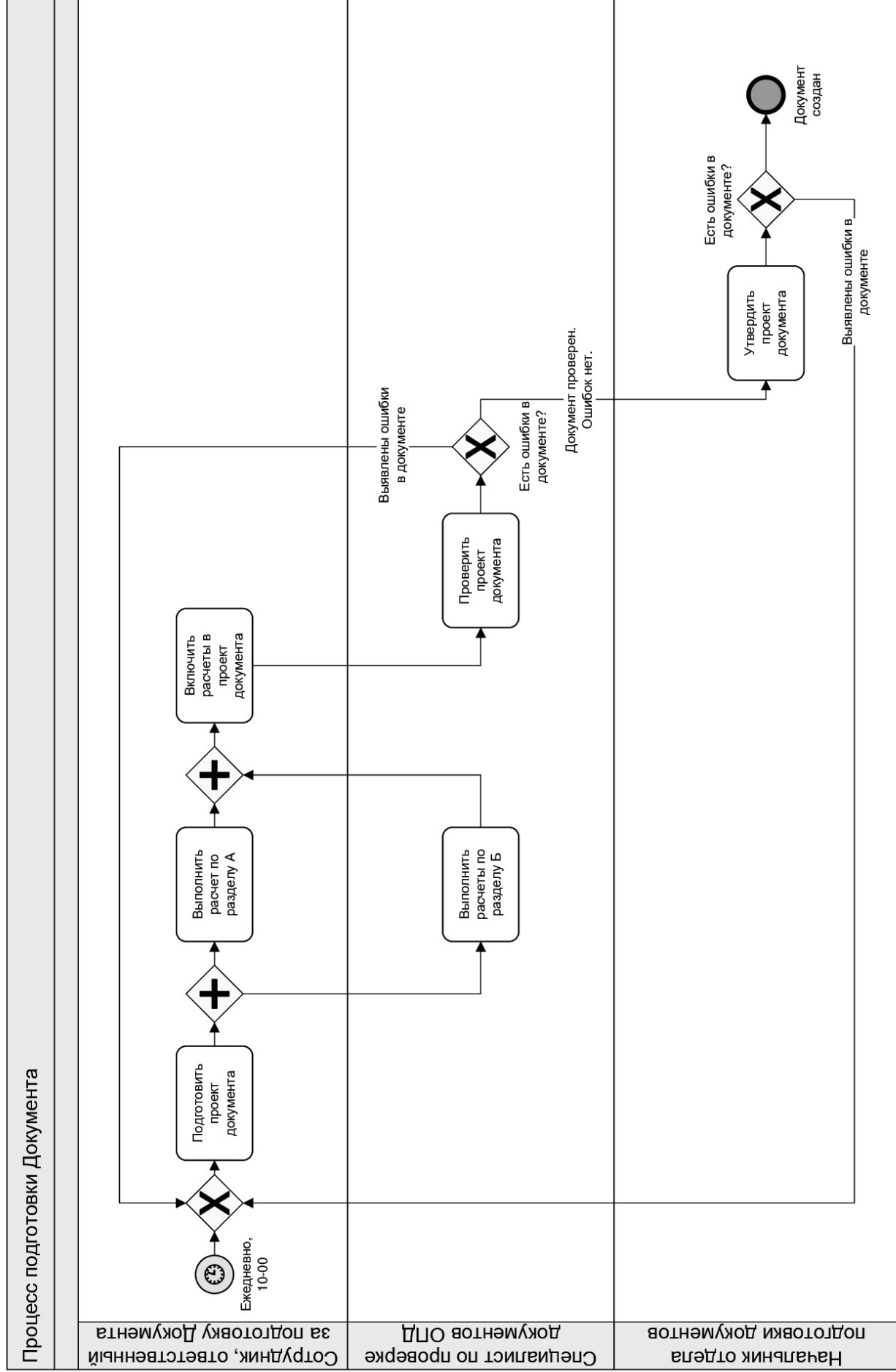


Рис. 9. Использование шлюза «И».

Для того чтобы показать, что две ветки процесса выполняются одновременно используется шлюз «И» — желтый квадратик с маркером в виде знака «плюс».

Первый такой шлюз разветвляет процесс на параллельно выполняющиеся ветки, второй — объединяет процесс.

В примере, изображенном на рисунке 9, это означает, что операция «Включить расчеты в проект документа» не будет запущена, пока не будут выполнены обе операции «Выполнить расчет по разделу А» и «Выполнить расчет по разделу Б».

Кстати, стрелка перехода от операции «Включить расчеты в проект документа» выходит из нижней грани четырехугольника и входит слева в операцию «Проверить проект документ». Можно ли так рисовать? Да, можно.

Слева или справа, сверху или снизу — это решение нужно принять и закрепить в так называемом Соглашении по моделированию (там еще много всего, причем более важного).

На рис. 10 показаны различные варианты использования шлюзов «И».

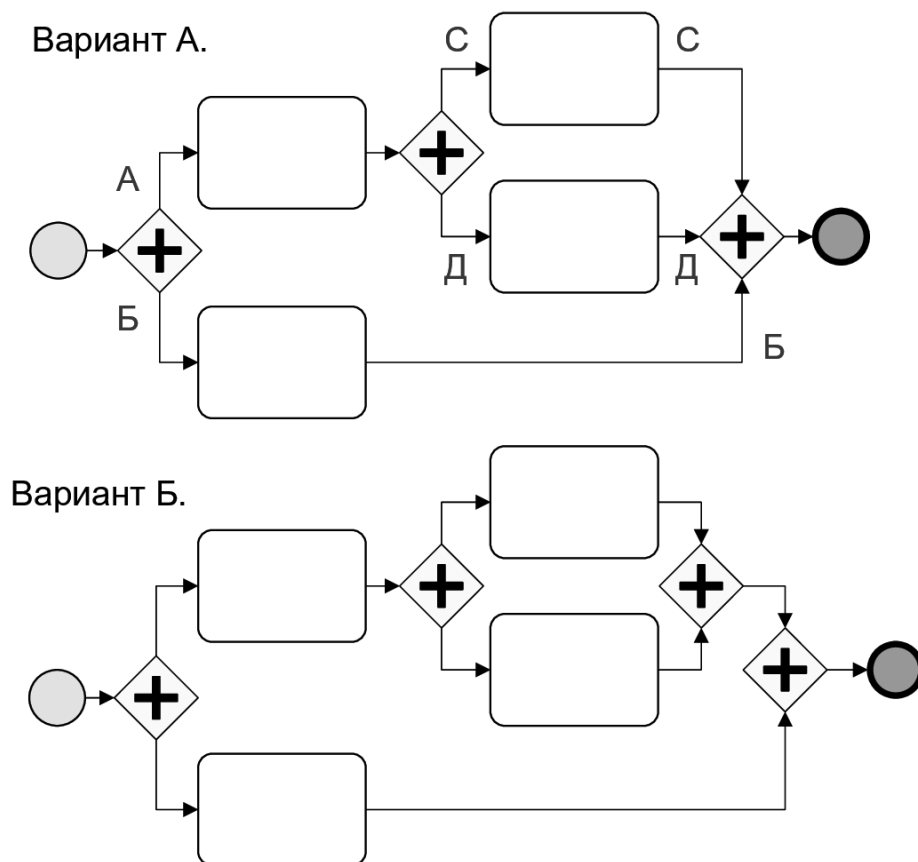


Рис. 10. Расходящиеся и сходящиеся потоки процесса.

Вариант А. После первого шлюза «И» возникает два параллельных потока А и Б. Затем после второго шлюза поток А разделяется на потоки С и Д. Затем все три потока сливаются в последнем шлюзе «И», который служит для объединения трех потоков.

Схема Варианта Б — полностью эквивалентна схеме Варианта А. Возможно, визуально она выглядит нагляднее. Но это вопрос довольно субъективный.

Когда на схеме процесса становится много операций, она может стать плохо читаемой, некрасивой. На рис. 11 приведен пример такой «некрасивой» схемы процесса. Формально, так делать можно. Но в контексте моделирования системы (архитектуры) процессов организации и удобства использования человеком можно выделить следующие недостатки этой схемы:

- слишком много операций на схеме — необходимо укрупнять, объединяя или исключая операции;
- поток работ движется на схеме зигзагом (справа — налево, слева — направо и т.п.), это запутывает пользователя;
- названия событий — «Начало» и «Конец» — плохо потому, что мы моделируем не сферического коня в вакууме, а *процесс в рамках системы взаимосвязанных и взаимодействующих процессов организации.*

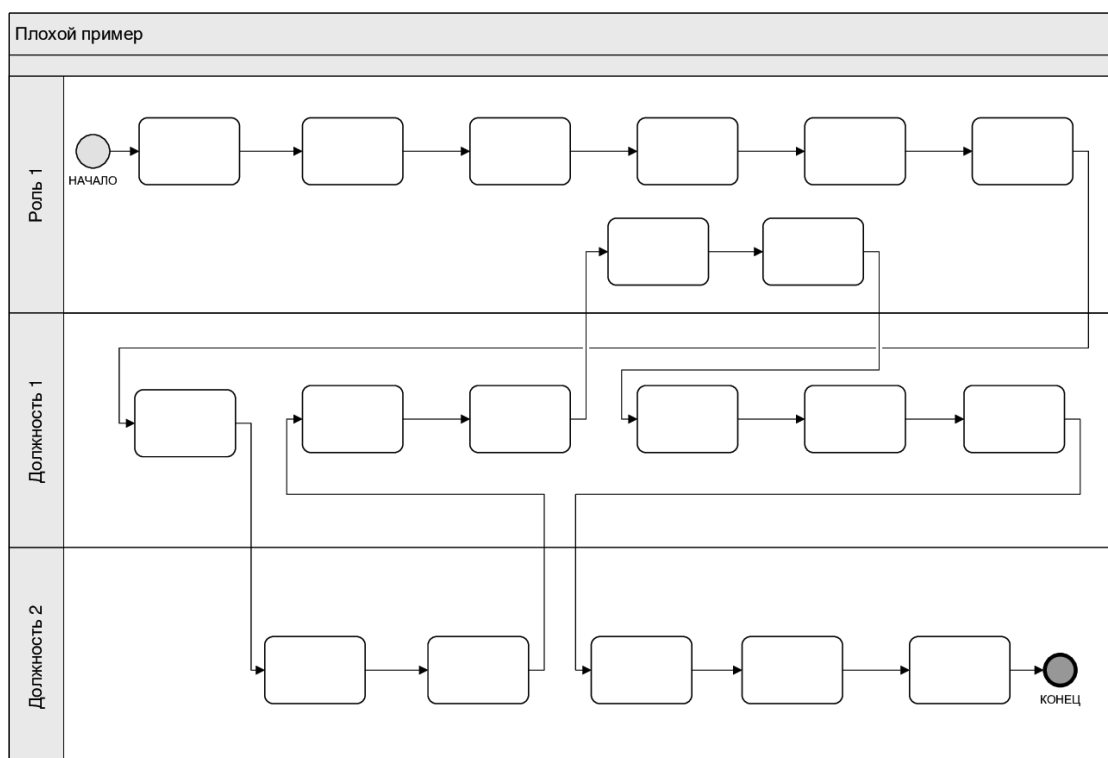


Рис. 11. Пример «некрасивой» схемы процесса.

На рис. 12 представлен еще один основной шлюз, который нужно знать. Это шлюз «Неисключающее ИЛИ». После этого шлюза процесс может пойти либо по одной ветке, либо по нескольким, либо по всем веткам одновременно. Вот такой странный шлюз.



Рис. 12. Шлюз «Неисключающее ИЛИ».

### 3.3. Типовые примеры. Логические ошибки

В этом разделе вы познакомитесь с типовыми примерами использования шлюзов и наиболее часто возникающими при этом логическими ошибками.

На рис. 13 показана самая простая, но часто возникающая логическая ошибка. После выполнения операции 1 выполняются либо операция 2, либо операция 3. Об этом нам говорит шлюз типа «Исключающее ИЛИ». Но далее мы видим шлюз «И» — объединение двух потоков работ. Но это объединение невозможно, и операция 4 никогда не будет выполнена.

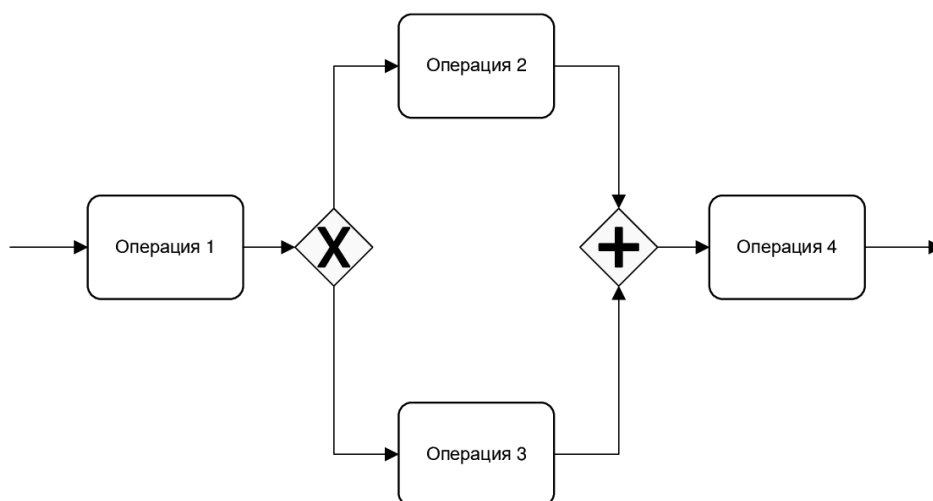


Рис. 13. Логическая ошибка на схеме. Пример 1.

На рис. 14 представлен пример еще одной логической ошибки, похожей на первую. Шлюз «И» разделяет процесс на две параллельно выполняемых ветки. Всё хорошо, но после операции 2 стоит шлюз «Исключающее ИЛИ». Т.е. в одном из случаев процесс никогда не дойдет до второго шлюза «И» и операция 4 никогда не будет выполнена.

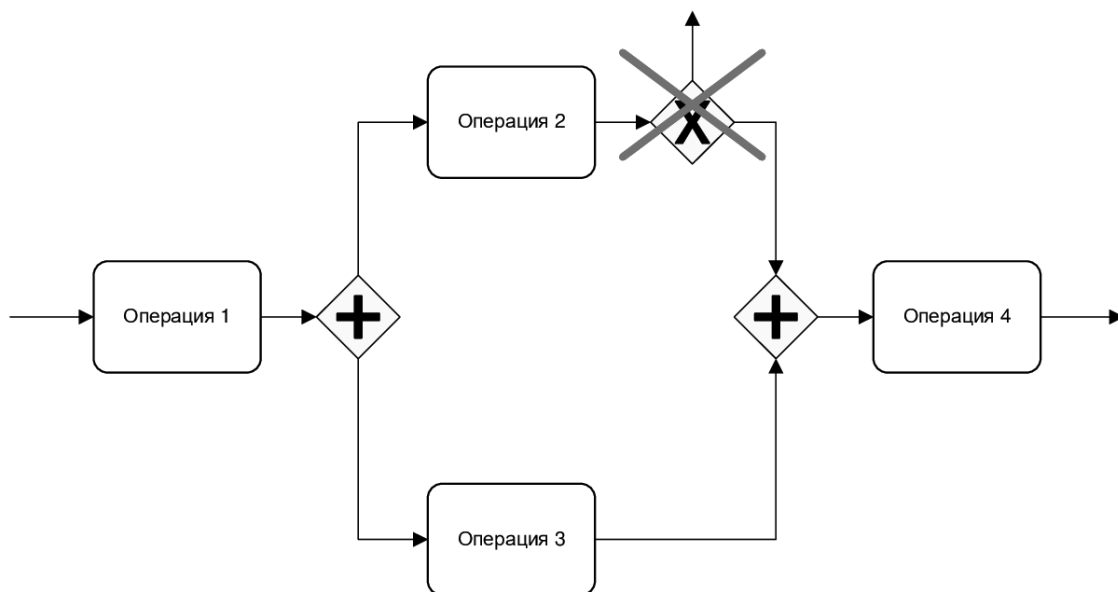


Рис. 14. Логическая ошибка на схеме. Пример 2.

Следующий пример можно назвать «Вечер пятницы».

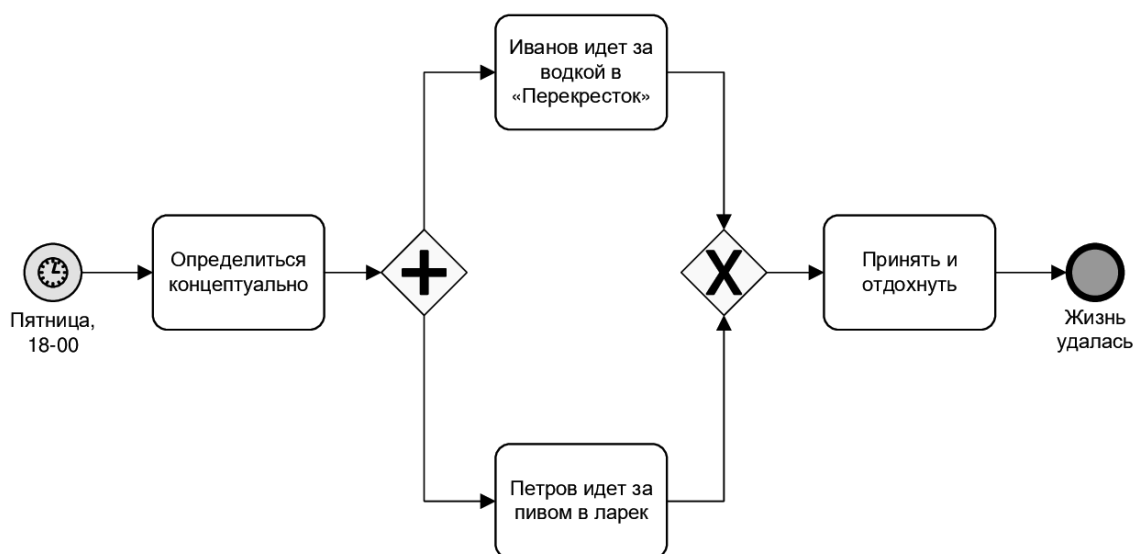


Рис. 15. «Вечер пятницы».

В пятницу вечером компания людей определяется, как будет проводить время. Потом одновременно (после шлюза «И») Иванов идет за водкой, а Петров за пивом. Далее мы видим шлюз «Исключающее ИЛИ». Это значит, что первый, кто вернется, тот и запустит следующий шаг процесса — отдых. Второго человека компания может уже и не дожидаться... Кстати, этот пример ошибкой не является. Так описать процесс можно, если это не нарушает логику процесса.

### 3.4. Некорректное использование шлюзов

На рис. 16 показана типичная ошибка применения шлюзов. Важно помнить, что шлюз не может объединять и разветвлять несколько потоков *одновременно!* Это некорректно.

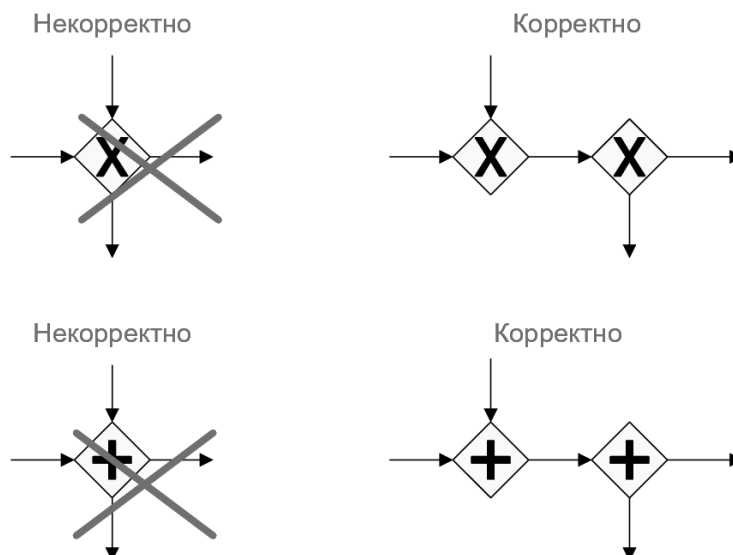


Рис. 16. Некорректное использование шлюзов.

### 3.5. Шлюзы для старта процесса

*Раздел 3.5. является сложным и может быть пропущен при первом чтении.*

Может ли процесс начинаться не одним событием, а несколькими событиями? Да, может. Если схема процесса разрабатывается для целей регламентации (т.е. передачи информации от человека — человеку), то наличие



нескольких разных стартовых событий допустимо, хотя и весьма нежелательно. Например, мы хотим показать на одной схеме несколько разных процессов (стартующих при разных обстоятельствах) вместо того, чтобы формировать и описывать в регламенте несколько графических схем. Однако, такой подход совершенно не годится, если мы создаем схему процесса для целей имитационного моделирования и/или автоматизации. Для того, чтобы это понять, нужно освоить понятие экземпляра процесса (раскрывается чуть ниже). Система автоматизации должна четко понимать, какое событие запускает экземпляр процесса. Если событий несколько, то система не сможет запустить экземпляр процесса до тех пор, пока правило запуска не будет четко определено.

Ниже на рис. 17 показаны различные возможные варианты.

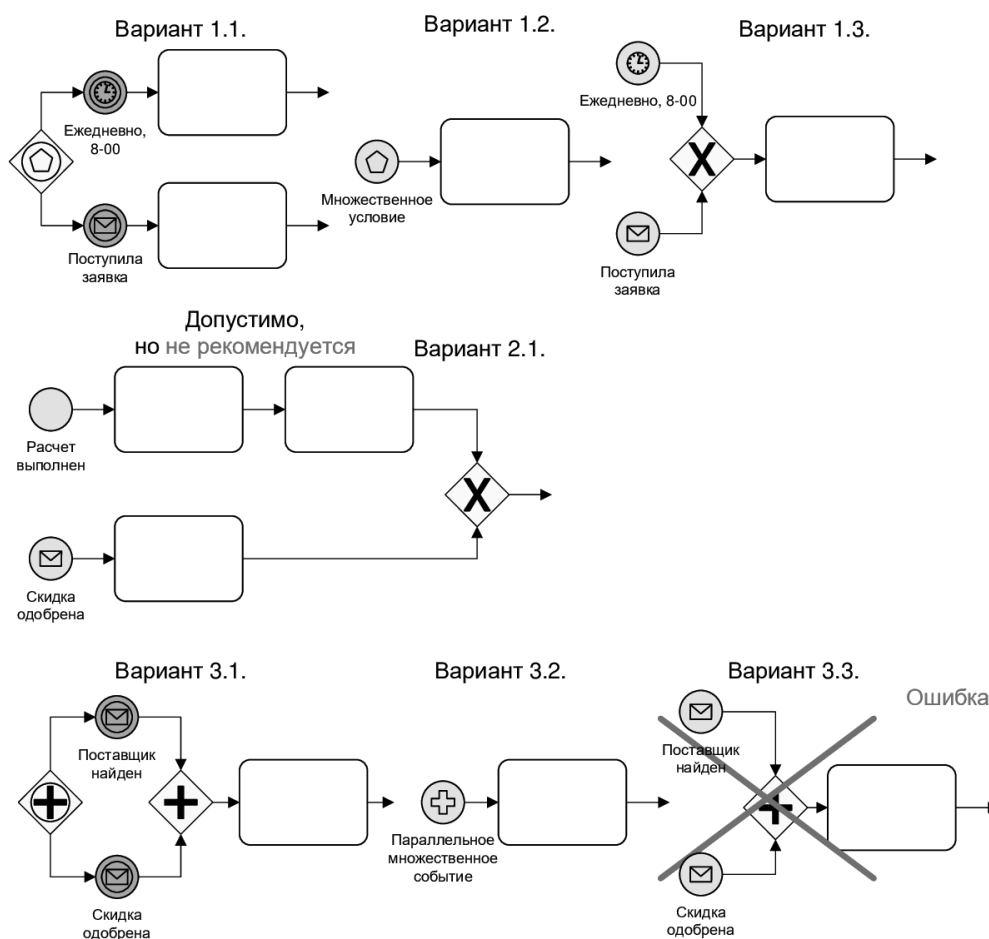


Рис. 17. Шлюзы для старта процесса.

Вверху рисунка 17 показаны три разных варианта старта процесса двумя различными событиями. Слева (Вариант 1.1.) показана ситуация, когда процесс может быть запущен двумя разными событиями. В этом случае ис-

пользуется специальный тип шлюза. Можно так же использовать специальный тип стартового события «Множественное условие» (Вариант 1.2.). Если вам просто нужно показать, что процесс стартует в случае возникновения одного из возможных событий, то можно использовать Вариант 1.3.

Если автоматизация процесса не является целью создания модели, то можно использовать Вариант 2.1.

Если необходимо показать, что процесс стартует только в случае возникновения сразу нескольких событий, т.е. объединить по «И», то можно использовать Вариант 3.1. и 3.2. Вариант 3.3. является ошибкой — так делать нельзя. Дело в том, что в случае Варианта 3.3. экземпляр процесса будет запущен одним из событий (тем, которое произойдет первым), но на шлюзе «И» процесс остановится и дальше не пойдет.

### 3.6. Головоломная задача

Предлагаем читателю выполнить небольшую задачку на освоение логики процесса. На рис. 18 представлен простой процесс.

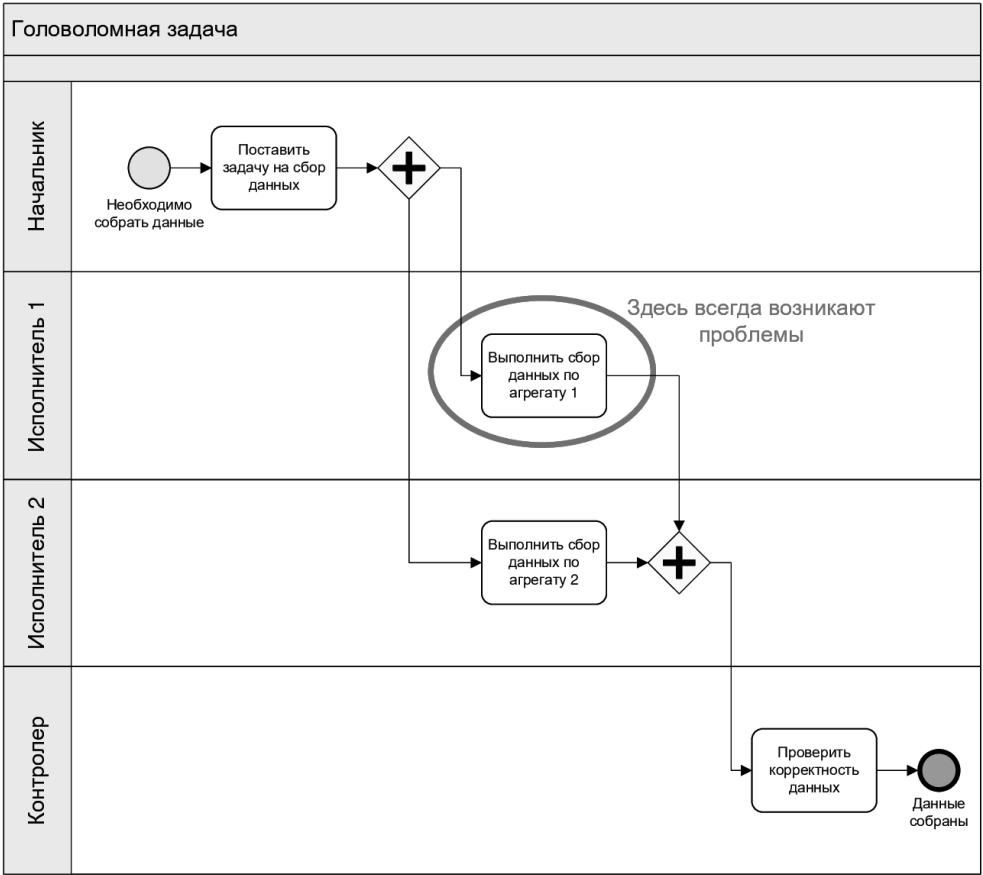


Рис. 18. Задачка на логику.

Сначала начальник ставит задачу, затем два сотрудника одновременно собирают данные по разным агрегатам. Потом контролер проверяет полученные данные.

Предположим, что Исполнитель 2 никогда не ошибается, а Исполнитель 1 может допустить ошибку. В случае если Исполнитель 1 допустил ошибку, после операции «Проверить корректность данных» нужно сделать корректный возврат так, чтобы операция «Выполнить сбор данных по агрегату 1» повторилась, а операция «Выполнить сбор данных по агрегату 2» — нет. В конце книги приведен один из возможных правильных ответов.

Ниже на рис. 19 представлена схема, которую обычно быстро рисуют, пытаясь найти ответ, но она неправильная. Предлагаю читателю самому ответить, почему.

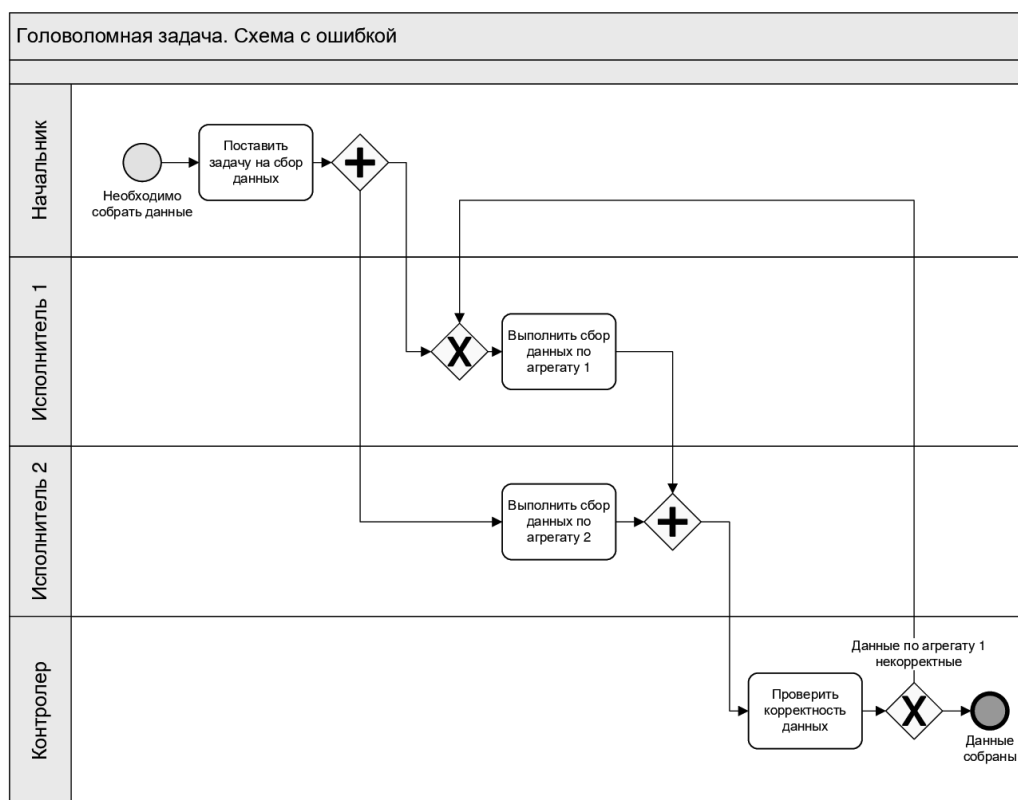


Рис. 19. Схема с логической ошибкой.

## 4. Движение документов на схеме процесса

*Как показать документооборот на схеме процесса. Почему нельзя оставлять «оборванные» входы и выходы. В чем разница между потоком работ и потоком документов.*

### 4.1. Движение документов внутри процесса

Печалька — BPMN не предназначен для моделирования потоков информации и документов.

А уж о перемещении материальных объектов и говорить нечего. Однако кое-что полезное на схемах показать можно.

Схема процесса на рис. 20 дополнена потоком документов. Например, «Документ» является исходящим для операции «Подготовить проект документа» и входящим для операции «Включить расчеты в проект документа».

Для отображения документа (информации, данных) используется специальная, интуитивно понятная пиктограмма. Пунктирные стрелки (их тип в BPMN — association) используются для привязки документов к операциям процесса.

Существует два возможных способа показать поток документов между операциями процесса.

Первый способ — просто показать при помощи пунктирных стрелок, что документ является исходящим для одной операции и входящим для другой — см. зеленый овал на рис. 20 — Вариант 1.

Второй способ (Вариант 2) — это привязать документ пунктиром к стрелке перехода между операциями процесса (это стрелка типа sequence).

При моделировании в Business Studio можно использовать оба эти варианта так, как удобно.

Если между операциями нет шлюзов, то такой вариант вполне можно использовать.

Заметим, что отображение потоков документов (информации) на схеме процесса Work Flow носит вторичный, вспомогательный характер. Документы делают схему более понятной, но не управляют потоком работ.

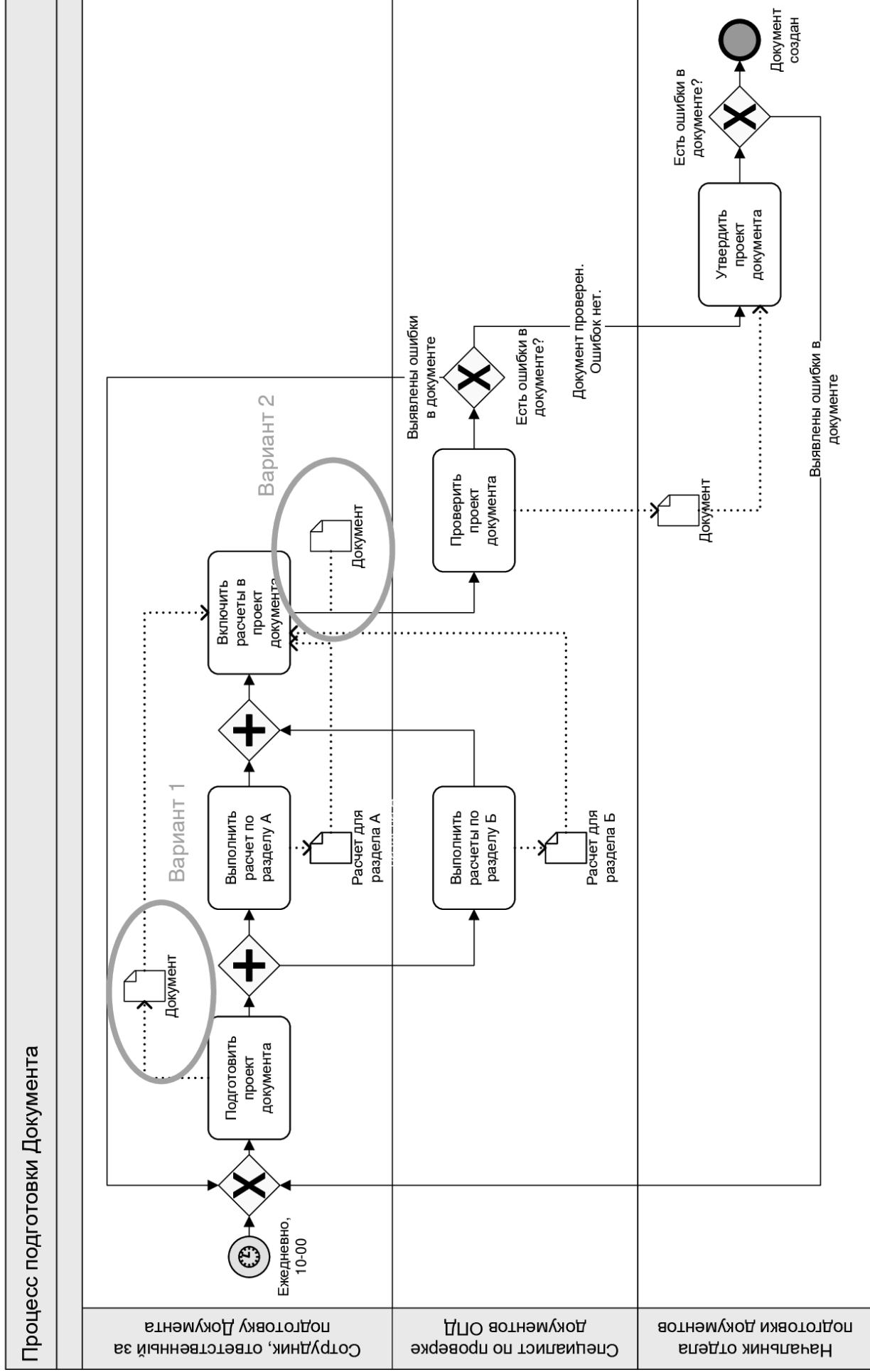


Рис. 20. Движение документов на схеме процесса.

Даже если ваша главная цель моделирования в BPMN — описание документооборота (это странно, но вполне возможно), не забывайте грамотно отображать поток работ.

## 4.2. Потоки документов между процессами

На рис. 21 показана условная схема взаимодействия двух разных процессов. Представим себе ситуацию, что Процесс 1 запускается независимо от Процесса 2 — в разное время и при своих обстоятельствах. По ходу выполнения Процесса 1 создается Информация ABC, которая используется внутри процесса и, кроме того, помещается в общее хранилище информации. Это может быть база данных, файл на диске и т. п.

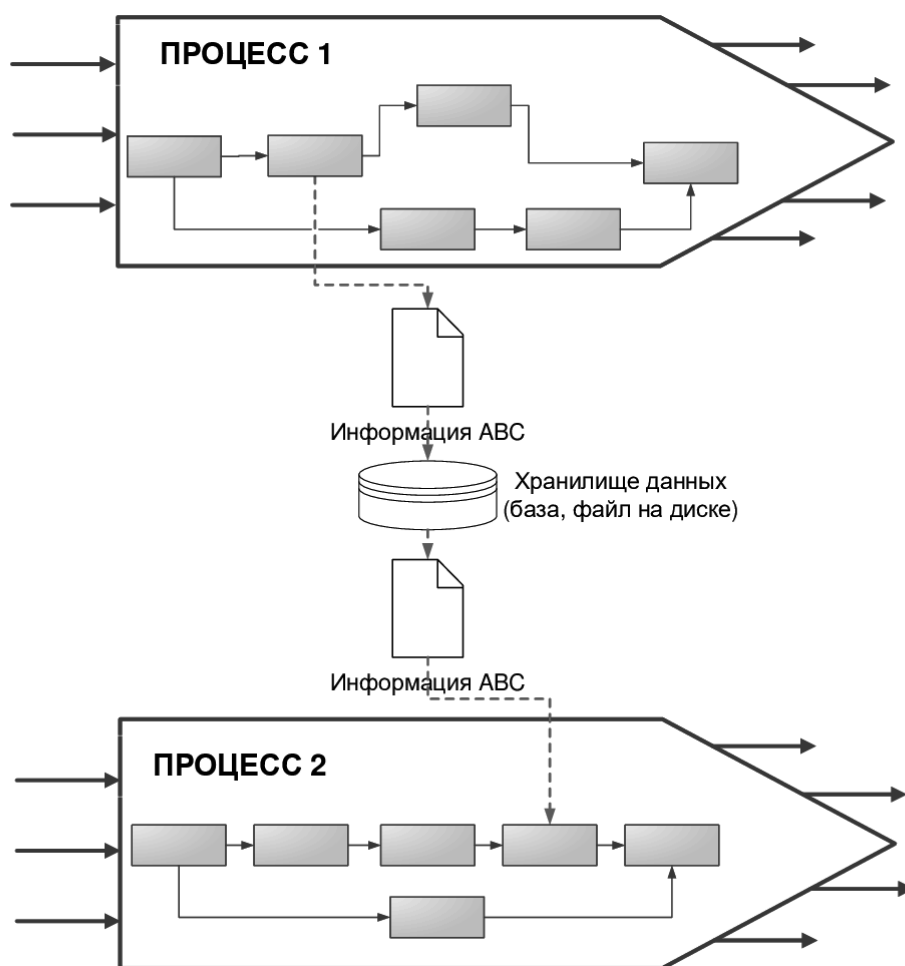


Рис. 21. Обмен информацией между процессами. Принципиальная схема.

Через некоторое время, *независимо* от Процесса 1, стартует Процесс 2. По ходу выполнения Процесс 2 обращается к хранилищу данных, берет оттуда Информацию ABC и использует ее для каких-то своих целей.

Можно ли говорить, что Информация ABC является выходом Процесса 1 и входом Процесса 2? Да, вполне. Хотя понятно, что эта информация поступает из одного процесса в другой не сразу, а через некоторое время, причем через хранилище данных.

Каким образом показать обмен информацией между процессами в BPMN? Пример представлен на рис. 22.

Обратите внимание, что на схеме появился новый объект — небольшой четырехугольник вверх под названием «Процесс А». Это так называемый *свернутый пул*. Он означает процесс, содержание которого в данный момент нас не интересует. Но нам важно показать взаимосвязь Процесса А и Процесса подготовки Документа.

Из свернутого пула «Процесс А» выходит пунктирная стрелка типа message. К ней тонкой пунктирной линией присоединен объект «Данные для расчета». Таким образом можно показать, что эта информация является выходом Процесса А и входом для Процесса подготовки Документа, в частности для его операции «Выполнить расчет по разделу А».

Справа внизу схемы показан другой свернутый пул, который символизирует Процесс Б. «Документ» является выходом операции «Утвердить проект документа» и входом для Процесса Б.

На рис. 22 показано, что «Документ» связан стрелкой с блинчиком под названием «Сервер компании (база документов)». Возможная интерпретация такой картинки — Документ в качестве выхода операции «Проверить проект документа» попадает в конкретное хранилище данных.

Строго говоря, все документы (информация) могут перемещаться между операциями процесса только через хранилище данных. Однако для упрощения схемы можно договориться, что эти хранилища можно показывать только тогда, когда завершается обработка документа (информации) и далее в рамках процесса с ней работа не ведется.

С другой стороны, если в процессе нужна какая-то информация (документ), то можно показать другой процесс (в виде свернутого пула), выходом которого она является. А можно показать, что информация берется из хранилища (базы) данных.

Если мы используем вариант свернутого пула, то схема показывает, какой процесс является поставщиком данных. Это повышает ее информативность с точки зрения анализа взаимодействия процессов организации. Если такой цели при создании модели нет, то можно использовать значки хранилищ (баз) данных.

Возможны два варианта, и в данном случае нет четких рецептов, какой из них лучше использовать.

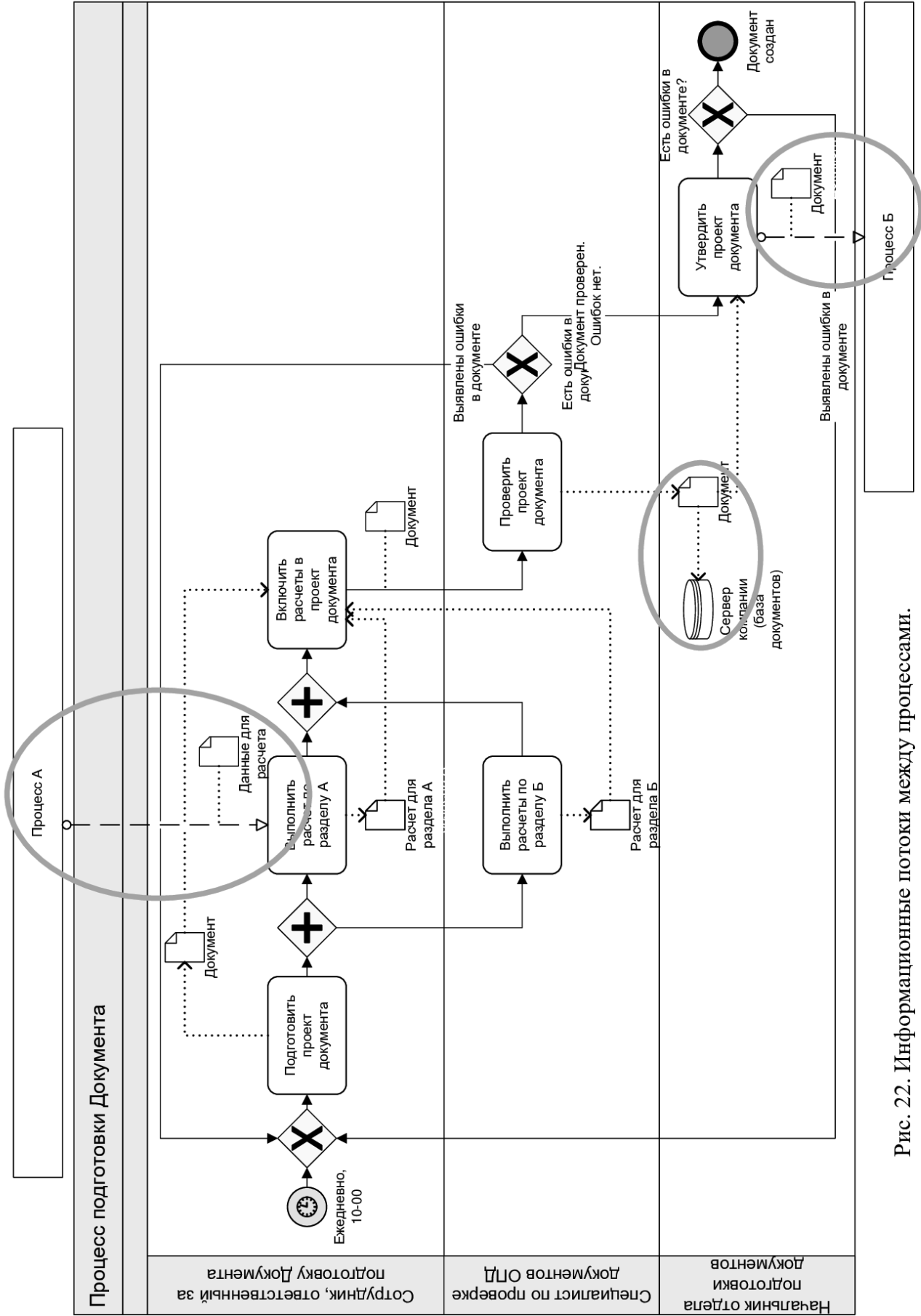


Рис. 22. Информационные потоки между процессами.



### 4.3. Разница между потоком работ и потоком документов

На рис. 23 представлена разница между потоком работ (Work Flow) и потоком документов. Операция 2 и Операция 3 связаны при помощи перемещения Документа. Между двумя операциями процесса можно показывать движение документов, как говорилось выше. Проблема в том, что после завершения выполнения Операции 2 следующая Операция 3 никогда не начнет выполняться. Почему? Дело в том, что между ними нет связи типа sequence flow, т.е. эти две операции не связаны единым потоком работ. Связь типа sequence flow в нотации BPMN служит для запуска операций процесса. Потоки информации/документов внутри пула в нотации BPMN не используются для запуска операций на выполнение.

В нотации BPMN документооборот не заменяет собой поток работ. Это важно. Кстати, представленная на рис. 23 ошибка является довольно типичной для сотрудников, которые только приступают к освоению нотации BPMN.

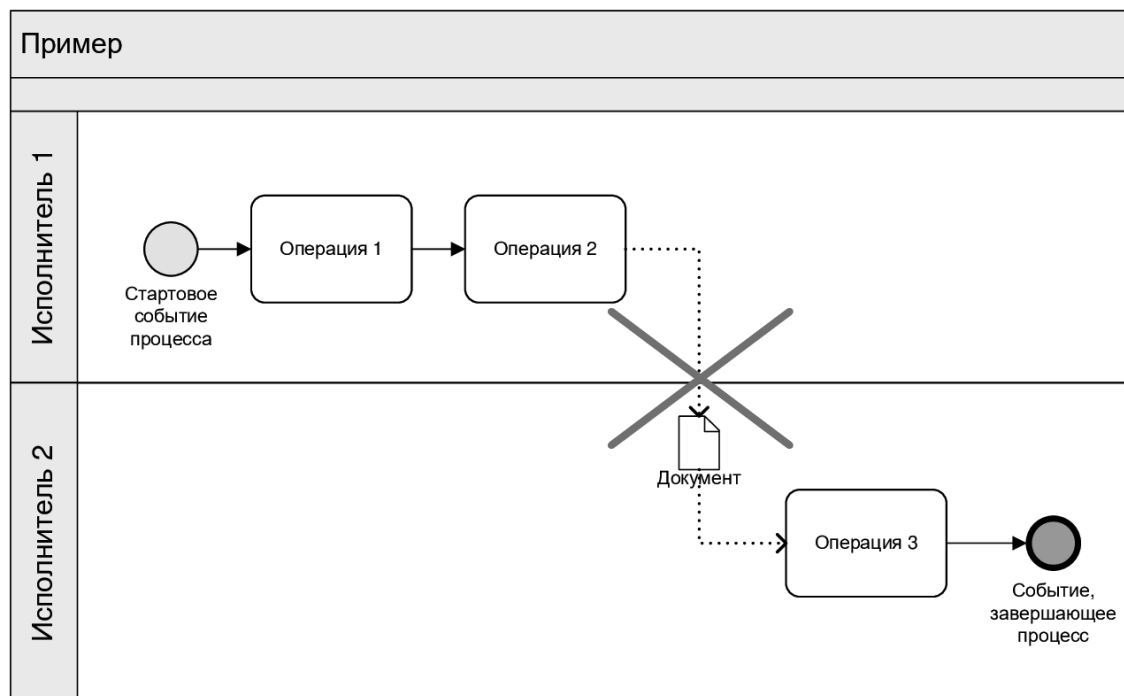


Рис. 23. Поток работ и поток документов.

## 5. Межпроцессное взаимодействие

*Понятие экземпляра процесса. Межпроцессное взаимодействие. Использование события типа «Отправка/получение сообщения». Разница между запуском процесса и простым обменом информацией.*

### 5.1. Понятие экземпляра процесса

Возможно, читатель уже знаком с этим понятием. Если нет, то его нужно пояснить. На рис. 24 показан некоторый процесс. С *определенной интенсивностью* в течение рабочего дня на вход процесса подаются, например, заявки. Это *нагрузка на процесс*.

Далее по установленному алгоритму (состоит из 6 операций) по ходу процесса входы преобразуются в выходы, например, производятся счета на оплату.

Утром, в 9—00 возникает первое событие — «Поступила заявка» (сама заявка как документ является входом процесса). Стартует Экземпляр №1 процесса — начинается обработка заявки по установленному алгоритму. В 9—30 утра возникает второе событие (тоже «Поступила заявка», но документ уже другой) — запускается Экземпляр №2. В 10—00 поступает третья заявка и стартует Экземпляр №3. В это время, как показано на рис. 24, Экземпляр №1 находится уже на шестой, последней операции процесса. В то же самое время в Экземпляре №2 выполняется уже третья операция процесса.

Таким образом, одновременно может быть запущено в работу несколько экземпляров процесса. К конкретному времени каждый из них будет находиться на разной стадии выполнения.

Количество экземпляров процесса, которые могут быть выполнены за единицу времени (например, за один день или один месяц) является мерой производительности процесса.

Что влияет на эту величину? Как сам алгоритм, так и наличие ресурсов, которые нужны для его выполнения. Это люди, оборудование, информационные системы. Поскольку всегда есть конкретное, счетное количество ресурсов, то производительность процесса всегда ограничена определенным значением.

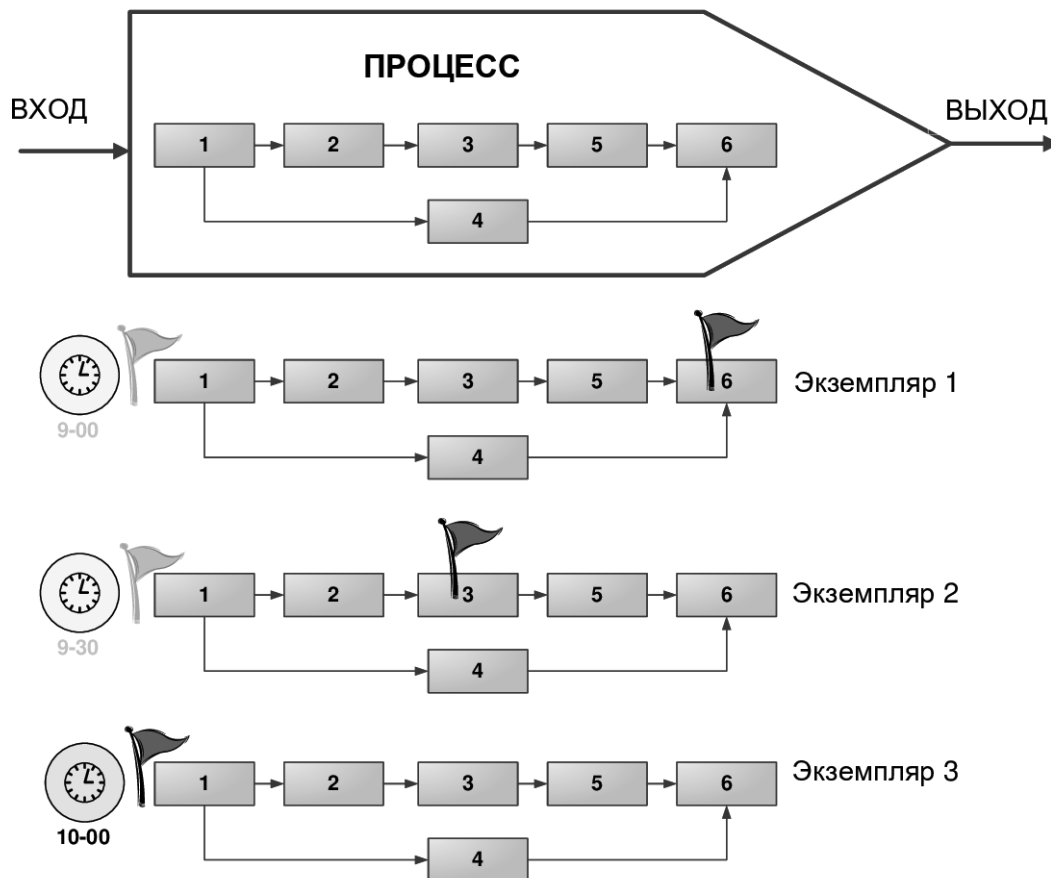


Рис. 24. Понятие экземпляра процесса.

Разрабатывая графическую схему процесса *важно помнить о нагрузке, ограничениях по ресурсам и производительности процесса*. Почему так? Главная цель описания — последующий анализ и оптимизация реально выполняемых процессов организации.

Почему понятие экземпляра процесса я обсуждаю именно сейчас? Потому что далее разбирается межпроцессное взаимодействие с точки зрения запуска одного процесса другим процессом. *Но взаимодействуют не абстрактные графические схемы, а конкретные экземпляры различных процессов.*

## 5.2. Старт процесса по событию получения сообщения

Рассмотрим рис. 25. Мы видим стартовое событие с маркером внутри. Это белый конверт. Событие соединено стрелкой типа message со свернутым пулом «Процесс-инициатор». Такая ситуация означает, что наш про-

цесс запускается другим процессом путем отправки сообщения. Другими словами, *экземпляр нашего процесса обязан стартовать сразу по факту получения сообщения из другого процесса*. Дополнительно на схеме показано, что отправка этого сообщения сопровождается передачей документа «Заявка».

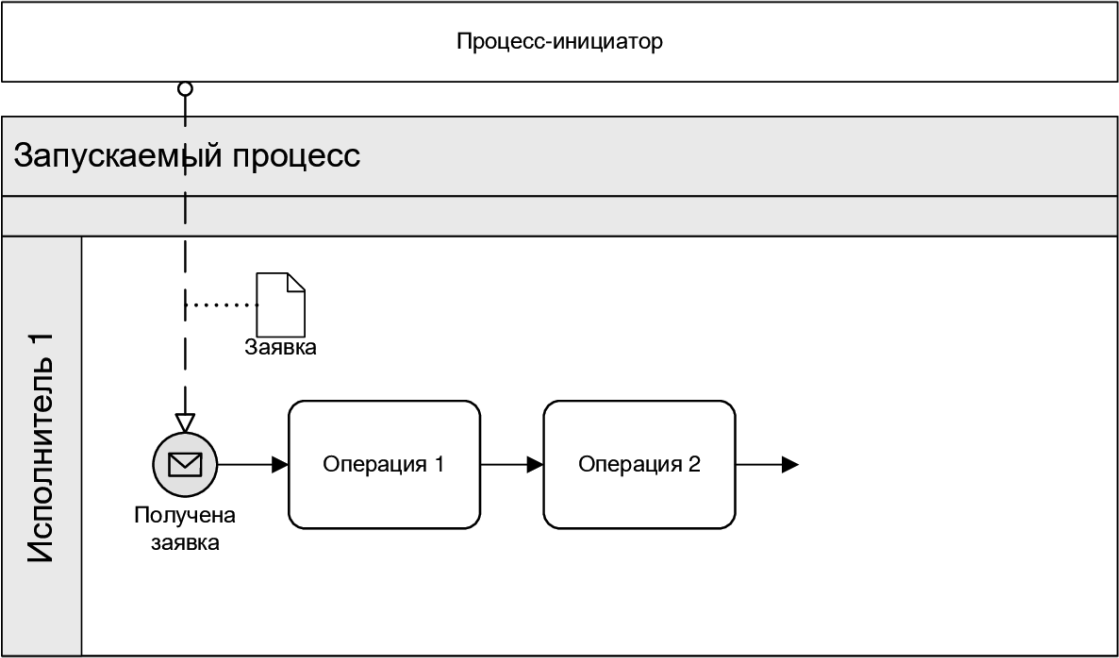


Рис. 25. Старт по событию получения сообщения.

Как запомнить для себя пиктограмму (маркер) стартового события-сообщения? Конверт открыт. Всё, что находится внутри конверта, нам видно. Там светло, лежат белые листы и т. п. Такого объяснения нет в нотации BPMN, но для себя так запоминать можно.

### 5.3. Завершение процесса событием отправки сообщения

На рис. 26 представлен фрагмент Процесса-инициатора. Он заканчивается событием отправки сообщения. Это означает, что процесс не просто завершается, а при завершении запускает на выполнение другой процесс.

Использован специальный тип события. Видно, что линия окружности толстая, а конверт темный.

Как запомнить, что темный конверт — это всегда отправка сообщения? Конверт темный, запечатанный. Что внутри — не видно, темно. Лично мне так проще запоминать.

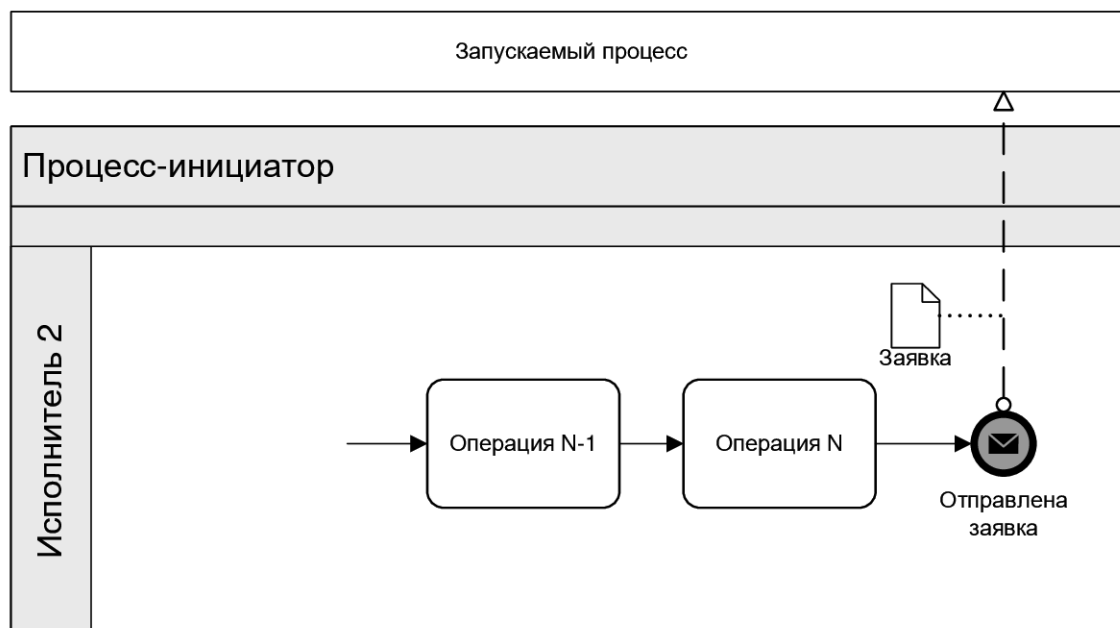


Рис. 26. Завершение процесса событием отправки сообщения.

На рис. 26 видно, что Процесс-инициатор завершается отправкой сообщения. Запускаемый процесс (см. рис 25) стартует с момента получения этого сообщения. Таким образом, два процесса взаимодействуют между собой, причем не только информационно, а с точки зрения управления. В других нотациях моделирования (таких, как классический CFFC с ромбиками или eEPC) такого нет и в помине.

Нотация BPMN кардинально отличается от других нотаций именно возможностью моделировать управляющие воздействия, координацией процессов между собой по событиям.

#### 5.4. Промежуточные события отправки и получения сообщений

На рис. 27 показано два промежуточных события процесса. Тот факт, что событие промежуточное, отображается двумя окружностями.

Первое событие с темным конвертом — это отправка сообщения. Интерпретировать данную ситуацию нужно так. После выполнения Операции

N, Процесс А отправляет сообщение в Процесс Б. Поток работ идет дальше до события с белым конвертом «Получен ответ». Это означает, что после отправки сообщения Процесс А *будет ждать* получения сообщения из Процесса Б. Только после получения сообщения из Процесса Б Процесс А сможет пойти дальше на Операцию N+1.

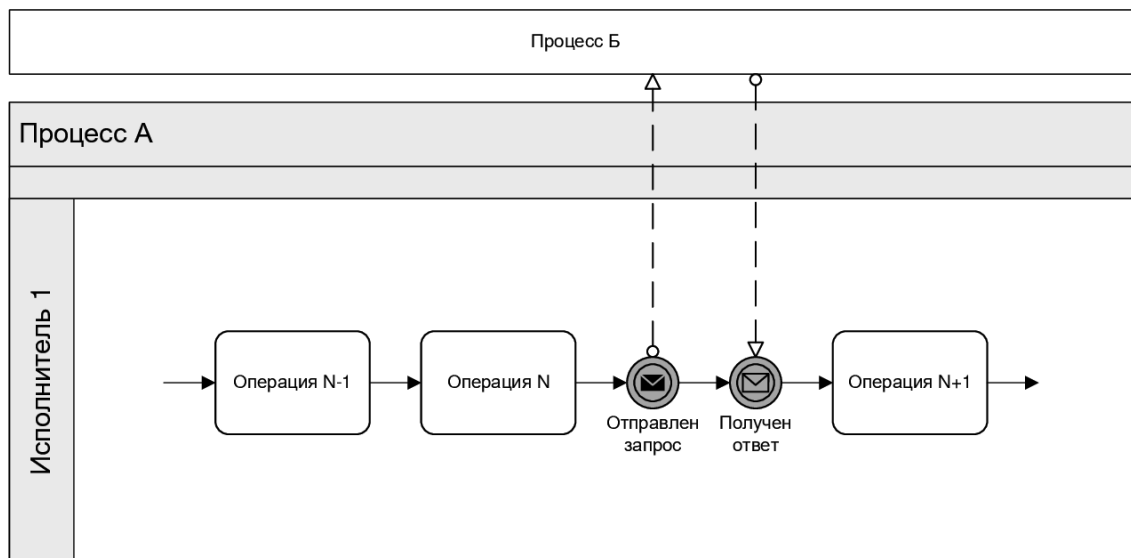


Рис. 27. Промежуточные события отправки и получения сообщений.

Фактически на рис. 27 представлена ситуация, когда по ходу выполнения работы необходимо инициировать экземпляр другого процесса, дождаться его выполнения, получить результаты и пойти дальше. Таким образом, *процесс может управлять другими процессами по ходу своего выполнения*.

Возможна ли ситуация, когда нужно просто отправить сообщение (запустить другой процесс) и сразу пойти дальше? Да, возможна. Это аналогично действию по принципу самонаводящейся ракеты — «выстрелил и забыл».

Возможна ли ситуация, когда по ходу процесса мы останавливаемся и ждем получения сообщения из другого процесса? Да, вполне. Т.е. не всегда чтобы что-то получить, мы должны что-то отправить.

По ходу выполнения отправляется запрос в Процесс А (т.е. инициируется на выполнение экземпляр Процесса А). После получения ответа выполняется операция «Выполнить расчет по разделу А».

Сравните эту ситуацию со схемой, представленной на рис. 22. Там было показано, что наш процесс просто получает информацию из Процесса А, но не запускает его на выполнение. Важно понять эту разницу.

Процесс на рис. 28 завершается отправкой сообщения в Процесс Б, т.е. *инициирует выполнение* этого процесса.

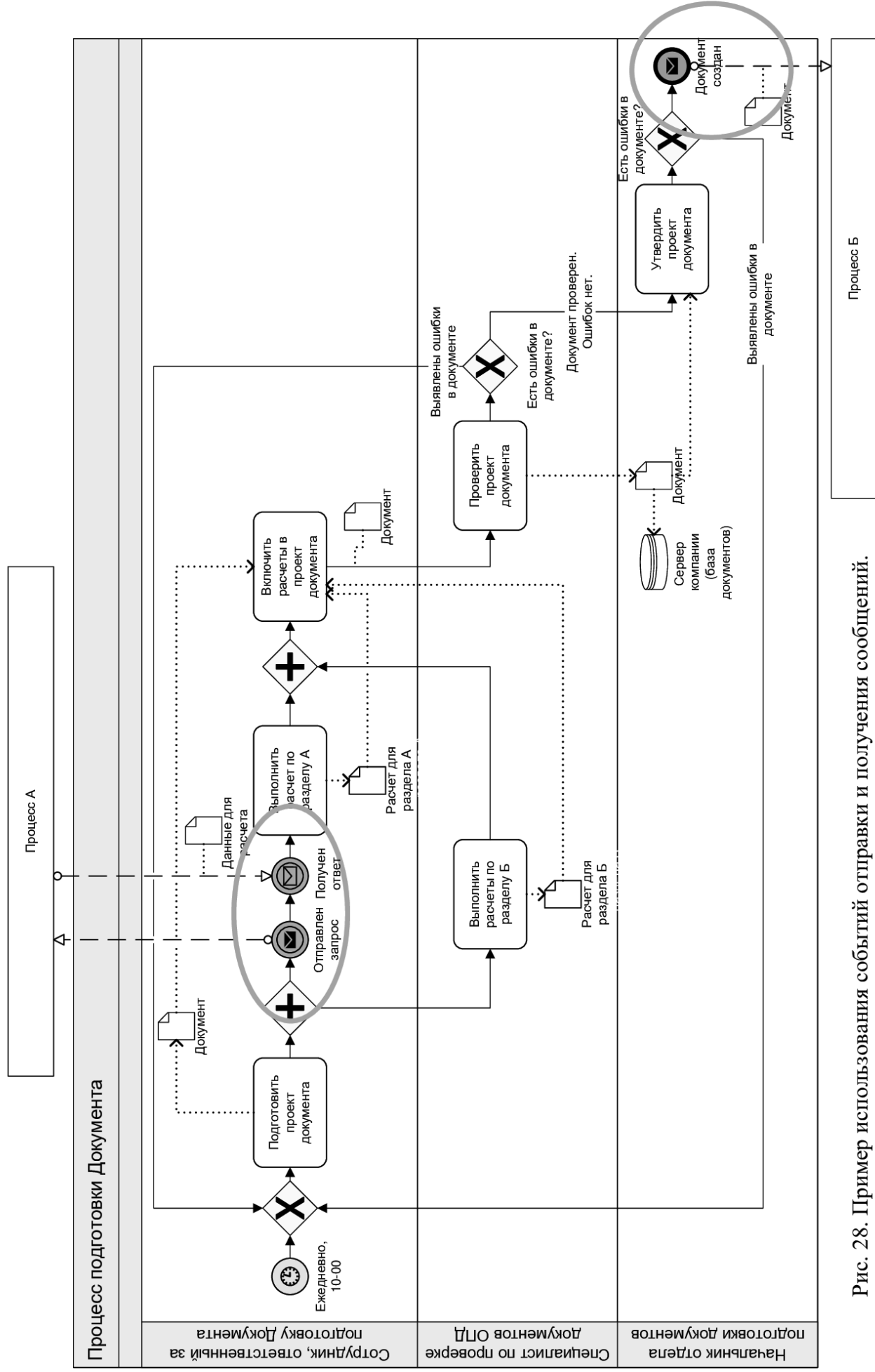


Рис. 28. Пример использования событий отправки и получения сообщений.

Отметим, что не желательно злоупотреблять на схеме событиями отправки/получения сообщений. Их нужно использовать только тогда, когда это реально необходимо.

Еще один аспект, который я хотел бы подчеркнуть особо. События-сообщения не обязательно должны запускать новые экземпляры других процессов. Они *могут просто инициировать выполнение операций в других процессах*, ожидающих сообщения. Но показать этот факт визуально на схеме процесса не получится.

### 5.5. Типовые ошибки при использовании событий отправки и получения сообщений

На рис. 29 показаны типовые ошибки, которые часто допускают при освоении нотации BPMN неопытные пользователи.

Во-первых, знак конвертика многими интуитивно понимается как отправка результата выполнения от одной операции процесса (документа, пакета документов, информации) к другой его операции. Поэтому просто берут и рисуют промежуточные события отправки сообщения, не связанные с внешними пулами. *Это очень грубая ошибка.* Она означает, что человек вообще не понимает ключевые особенности моделирования межпроцессного взаимодействия в нотации BPMN.

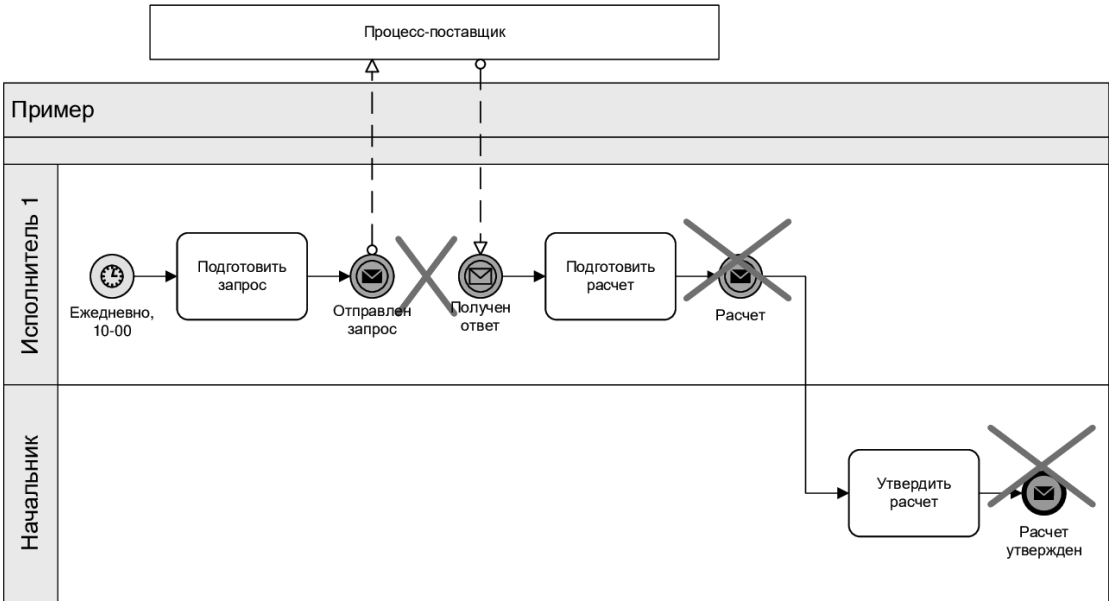


Рис. 29. Типовые ошибки при использовании событий отправки и получения сообщений.



На рис. 29 после операции «Подготовить расчет» показано событие отправки сообщения «Расчет». Это некорректно. Кстати, название этого события тоже не является корректным. Если вам очень хочется или вы уже назвали событие в терминах результата (существительное), то скорее всего вы допустили ошибку. Будьте внимательны.

В конце процесса показано событие отправки сообщения «Расчет утвержден». Формулировка события вполне адекватная, но на схеме не показано, какой процесс запускается этим событием. Это ошибка.

Еще одна ошибка случилась после операции «Подготовить запрос» и отправки сообщения. До этого места все правильно, но после события «Отправлен запрос» нет стрелки типа Sequence flow к событию «Получен ответ». Это означает, что процесс прервется и дальше не пойдет. Так делать нельзя.

## 6. Использование промежуточных событий-таймеров

*Промежуточные события-таймеры. Правила использования.*

По ходу реального процесса могут возникать различного рода задержки. Для того чтобы показать их на схеме, можно использовать промежуточные события-таймеры.

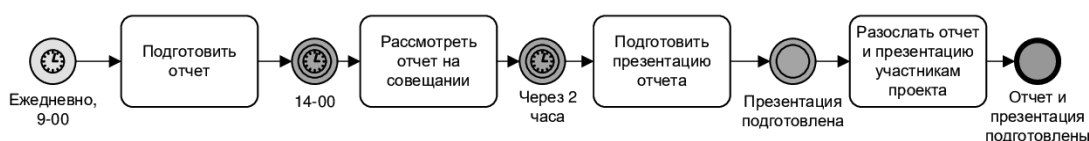


Рис. 30. Использование промежуточных событий-таймеров.

На рис. 30 показан пример корректного использования событий-таймеров. Процесс начинается ежедневно в 9—00 утра. Затем выполняется операция «Подготовить отчет». После нее вы видите промежуточное событие-таймер «14—00». Это означает, что при выполнении процесса возникает задержка до 14—00. Но имейте в виду, что условие события-таймера вычисляется с того момента, когда управление процесса переходит на это событие. То есть если операция «Подготовить отчет» не будет выполнена до 14 часов текущего дня (процесс перейдет на таймер после 14 часов), то таймер «14 часов» отработает только на следующий день. Задержка может быть обозначена как «25 числа месяца, 12—00» и т. п.

На практике наиболее часто используется динамическое назначение условий запуска события-таймера, которое определяется контекстом процесса (например, назначена дата встречи, и таймер установлен на эту дату).

Далее, после операции «Рассмотреть отчет на совещании» стоит событие-таймер «Через 2 часа». Это относительная задержка на 2 часа после выполнения операции. Так использовать таймеры тоже можно.

Далее, кстати, использовано промежуточное событие неопределенного типа «Презентация подготовлена». Те, кто описывал процессы в нотации eEPC и не могут жить без промежуточных событий процесса, могут использовать события неопределенного типа.

На схемах BPMN не запрещено использование промежуточных событий неопределенного типа, если вы считаете, что они повышают информативность схемы для человека. Но с точки зрения автоматизации процесса такого рода события ценности не имеют.

На рис. 31 представлен пример неправильного использования события-таймера. Нельзя делать таймер «Не позднее 3-х рабочих дней». Нечеткая формулировка. Не понятно, когда именно процесс должен продолжиться. Но зато такого типа формулировка крайне любима руководителями множества российских компаний, вероятно, за «гибкость». Но (называя вещи своими именами) это не гибкость, а *размывание ответственности за сроки выполнения работ и плохая организация процесса в целом*. Вот откуда у нас процессы согласования договоров по 3—4 недели и проч. Никакой эффективности.

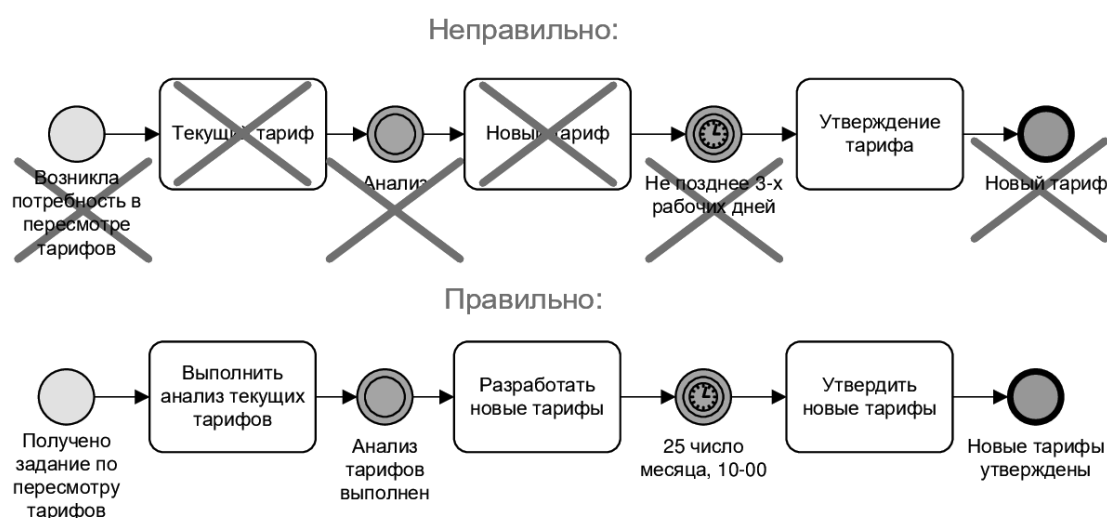


Рис.31. Примеры неправильного использования событий-таймеров.

Кроме ошибки с таймером, на рис. 31 красными крестиками показаны и другие ошибки:

- «Возникла потребность в пересмотре тарифов» — нечеткая формулировка; как возникла — путем медитации что ли?
- «Текущий тариф» и «Новый тариф» — ну нельзя так называть операции процесса, как, впрочем, их нельзя называть «Отдел продаж» или «Коммерческий директор»;
- «Анализ» и «Новый тариф» — неудачные названия; события весьма желательно именовать в терминах выполняемых действий, а не их результатов (существительные).

Возможные правильные варианты показаны в нижней части рис. 31.

# 7. Граничные события. Таймер. Эскалация. Ошибка

*Граничные события и правила их использования. Таймер. Эскалация. Ошибка.*

## 7.1. Граничные события-таймеры

Граничные события — весьма интересный и полезный инструмент нотации BPMN. В других нотациях его просто нет. Рассмотрим пример, представленный на рис. 32.

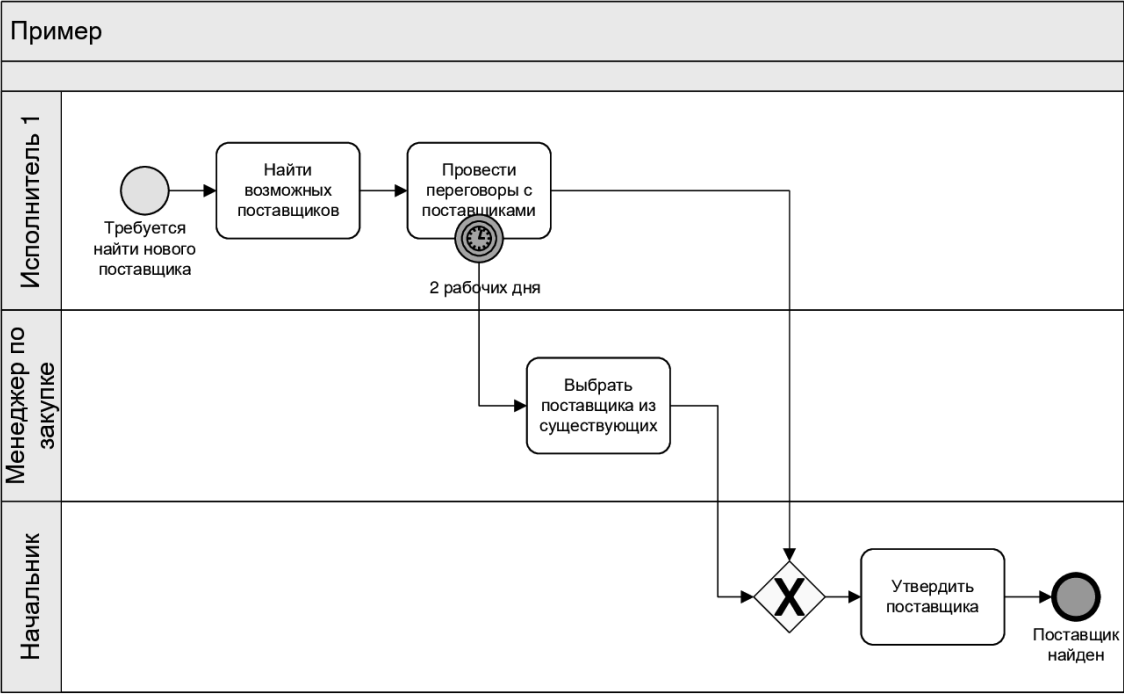


Рис. 32. Граничное событие-таймер.

К границе пиктограммы операции «Провести переговоры с поставщиками» прикреплено событие-таймер. Это и есть граничное событие. В слу-

чае если операция выполняется более двух рабочих дней (минут, часов — можно сформулировать так, как нужно), то *выполнение операции прерывается и осуществляется переход по другой ветке процесса*. В данном случае запускается операция «Выбрать поставщика из существующих». Таким образом мы можем прерывать выполнение операции в нужное нам время.

*Граничное событие-таймер* интересно тем, что оно *возникает во время выполнения операции процесса, а не после ее завершения*.

Очень часто неопытные пользователи BPMN допускают следующую ошибку — используют граничные события-таймеры либо в качестве индикаторов нормативного времени выполнения операции, либо в качестве каких-либо временных ограничений. Оба варианта использования некорректные — см. рис. 33. Но как хочется некоторым повесить красивые циферблаты со строгим указанием времени, чтобы всем не повадно было!

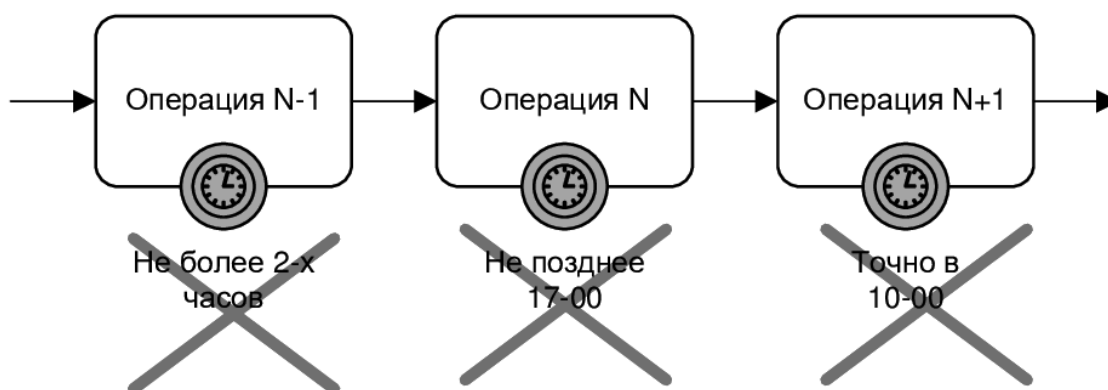


Рис. 33. Некорректное использование граничного события-таймера.

## 7.2. Граничное событие-эскалация

На рис. 34 представлено граничное событие-эскалация.

В случае, если при выполнении операции «Выполнить анализ отклонений» исполнитель выявил недопустимые (критические) отклонения параметров процесса, он может передать управление вышестоящему руководителю. Промежуточное событие с пиктограммой стрелки, направленной вверх, довольно легко запомнить.

С содержательной точки зрения передача управления вышестоящему руководителю не всегда хорошо. Этот факт означает отсутствие полномочий и ответственности у исполнителей процессов. Такая ситуация характерна для организаций с жесткой функциональной иерархией, функциональными колодцами и т. п. Внедрение процессного подхода призвано

изменить неэффективную систему работы. Поэтому не стоит увлекаться использованием событий типа «эскалация». Лучше подумать об оптимизации бизнес-процессов и орг. структуры компании.

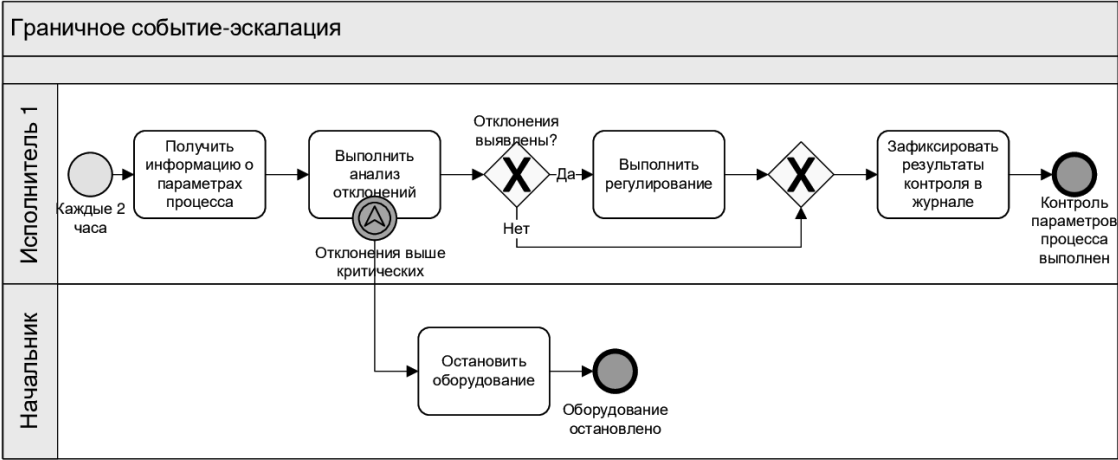


Рис. 34. Граничное событие-эскалация.

### 7.3. Граничное событие-системная ошибка

Граничное событие-системная ошибка (см. рис. 35) используется в случае, когда при выполнении операции может возникать критическая ошибка, приводящая к невозможности ее выполнения. В этом случае, операция прерывается, и управление передается на другую операцию процесса.

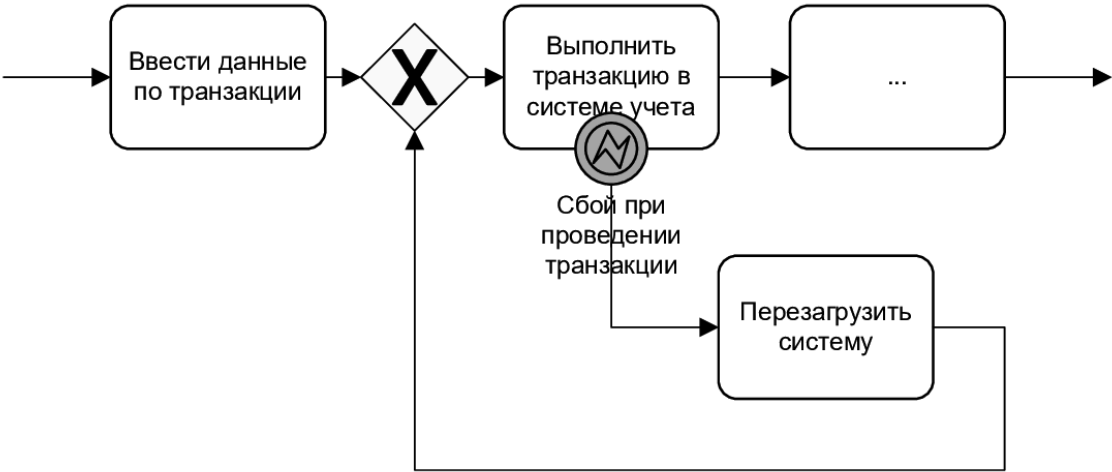


Рис. 35. Граничное событие-системная ошибка.

Довольно часто граничное событие-системная ошибка используют не по назначению, подменяя им обычную логику процесса, которую можно описать при помощи шлюзов — см. рис. 36. Это плохо, так может вызвать непонимание и некорректное использования данного типа событий.

На рис. 36 показаны два разных варианта описания фрагмента процесса. Сверху — вариант с некорректным использованием граничного события-системная ошибка. Чтобы никого не путать, я настоятельно рекомендую так НЕ делать.

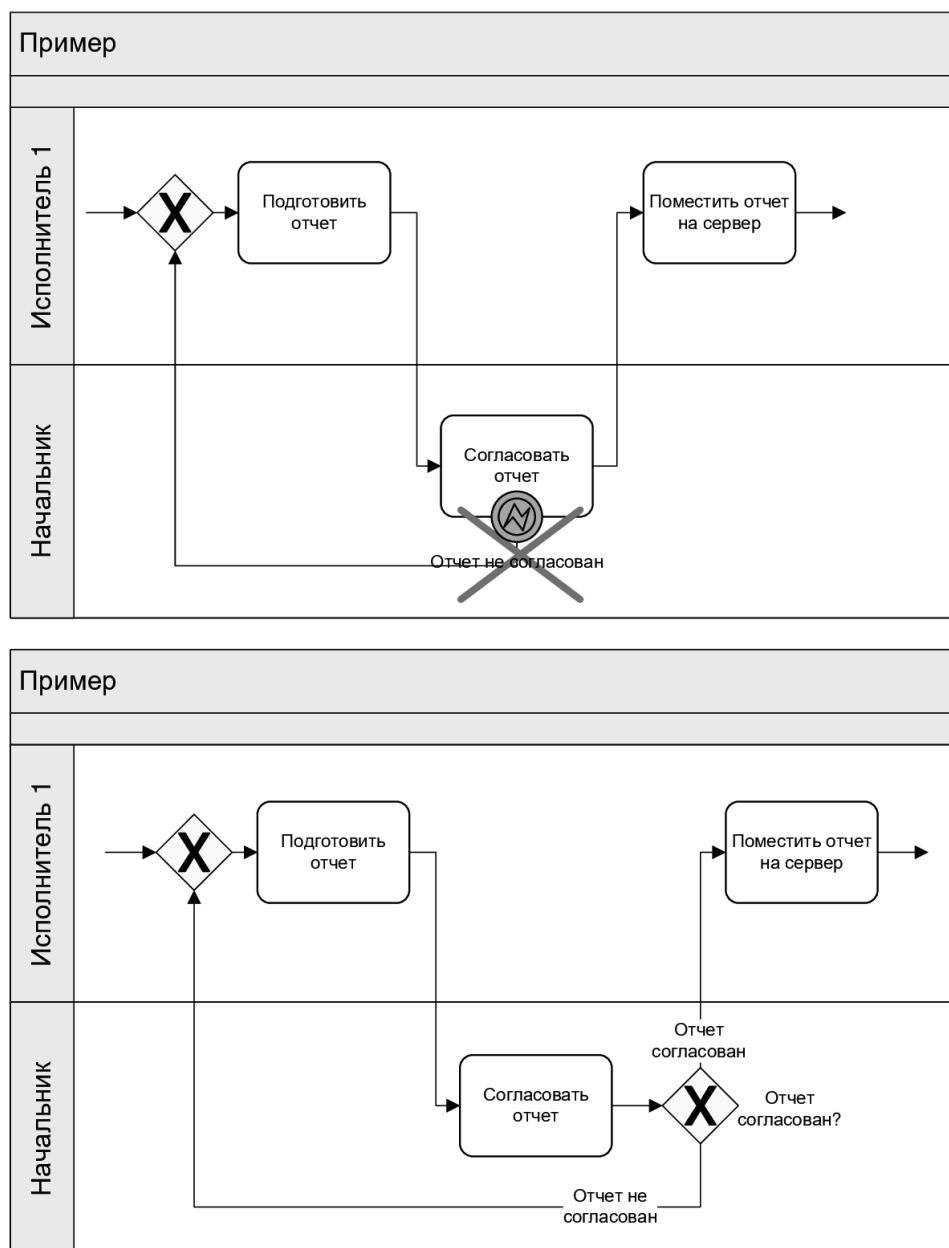


Рис 36. Пример некорректного использования граничных событий-системных ошибок.

## 8. Сложные шлюзы в процессе

На рис. 37 показан сложный (своей картинкой внутри — маркером, и содержанием) шлюз «Исключающее ИЛИ по событиям» (Event-Based Gateway).

После того, как по ходу работы в процессе возникает промежуточное событие отправки сообщения «Отправлен запрос», наш процесс останавливается и ждет возникновения одного из двух возможных событий.

Либо первым возникнет промежуточное событие получения сообщения «Получен ответ», либо сработает таймер «2 раб. дня».

Если раньше будет получено сообщение, то процесс пойдет дальше.

Если «Процесс А» в течение двух рабочих дней не будет выполнен и нужная информация не будет предоставлена, то начальник отдела применит административный ресурс (операция «Уведомить руководителя исполнителя процесса А»). Для этого после таймера «2 раб. дня» стоит шлюз «И», после которого выполняется операция «Уведомить исполнителя руководителя Процесса А» и возврат на шлюз (через технический шлюз «ИЛИ»), ожидающий получения сообщения.

Если во время выполнения операции «Уведомить руководителя...» придет необходимое сообщение из Процесса А, то схема рассматриваемого процесса работает корректно.

Минусом такой схемы является тот факт, что в реальной BPMS системе, если Начальник отдела подготовки документов не будет выполнять поступившие задачи «Уведомить...», каждые два рабочих дня будет возникать новая задача, и число их будет расти. Но, в конце концов, на каждого начальника есть другой начальник, который проверяет статусы выполнения операций в BPMS — бесконечного накопления просроченных задач можно будет избежать путем ручного управления.



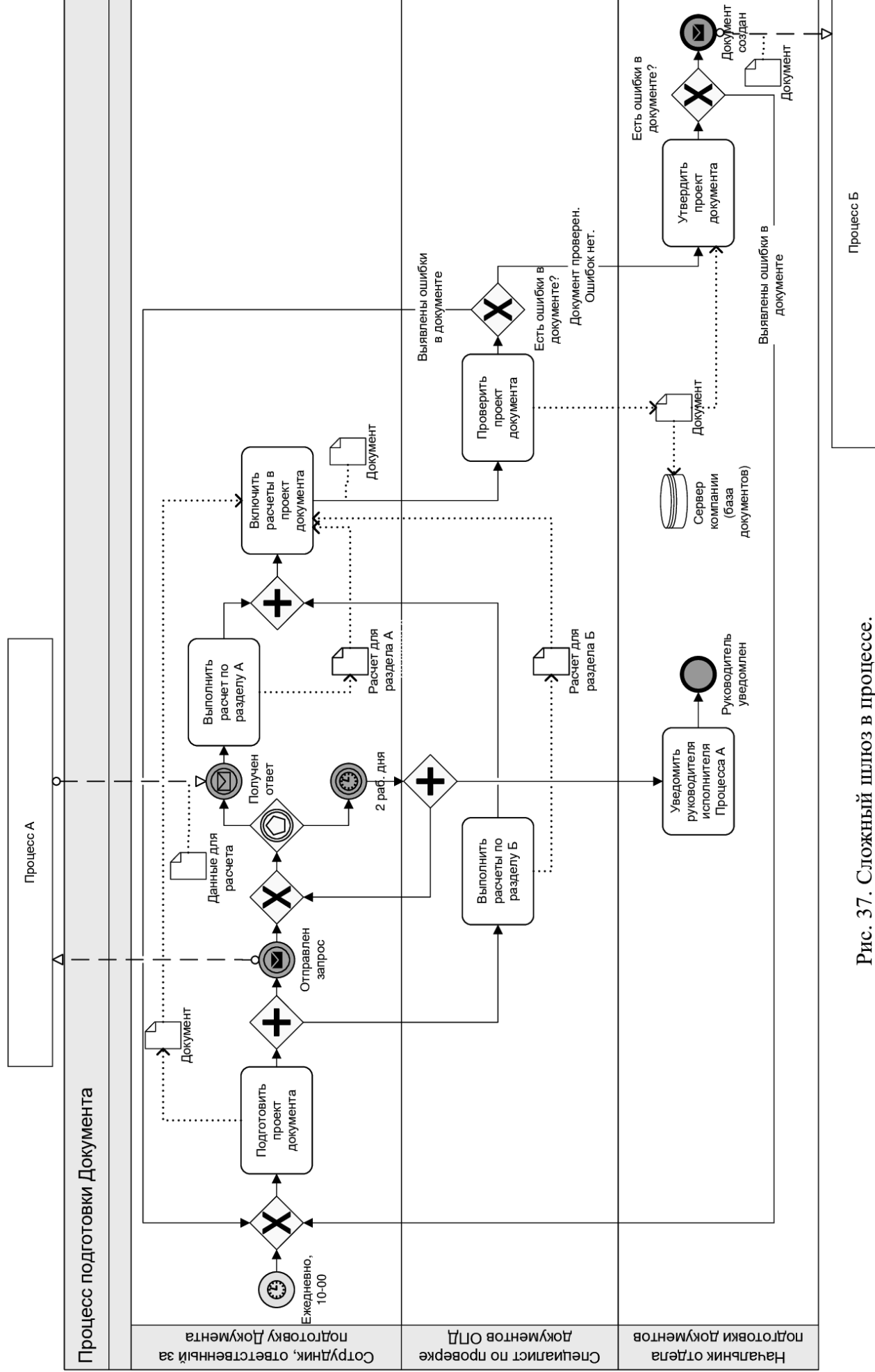


Рис. 37. Сложный шлюз в процессе.

На рис. 38 показаны возможные варианты выхода из бесконечного цикла ожидания ответа из другого процесса и повторной отправки запросов на предоставление информации (или для решения проблемы «зависания» задач, описанной выше для рис. 37). В случае, если после отправки сообщения в другой процесс проходит более 1 рабочего дня, выполняется проверка количества отправленных запросов. Если запросов было уже 3, то выполняется «Операция X» и процесс останавливается. Если запросов было менее 3, то осуществляется возврат и повторное выполнение «Операции N».

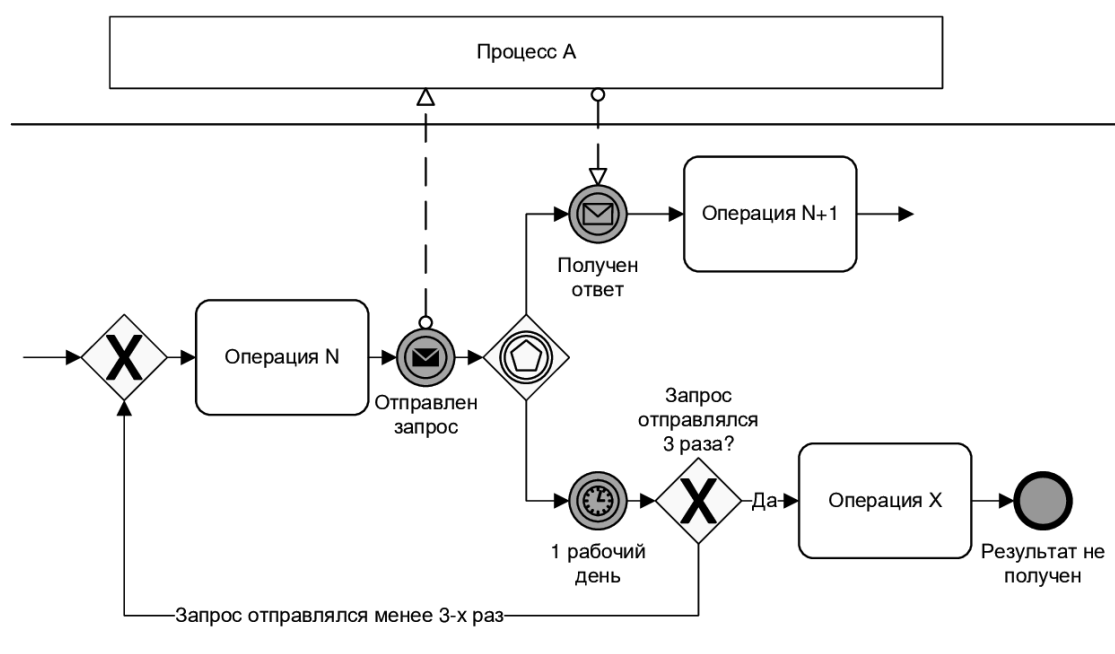


Рис. 38. Выход из бесконечного цикла ожидания ответа.

## 9. Декомпозиция процессов

Декомпозиция — это разделение сложного объекта на части. При описании процессов декомпозиция означает описание процесса на нижестоящем уровне, который может включать 10–15 шагов. А больше можно? Можно, но крайне нежелательно, т.к. схема перестает быть читаемой.

На рис. 39 показан фрагмент процесса. Внутри операции «Подготовить проект документа» вы видите небольшой квадратик с крестиком — он означает свернутый подпроцесс (collapsed subprocess). На рисунке видно, что на нижнем уровне этот подпроцесс включает три операции.

Обратите внимание, что стартовые и завершающие процесс события: а) неопределенного типа; б) контекстно поименованы.

Последнее требование — необязательное. Но корректное именование объектов на схеме (в т.ч. событий) повышает ее информативность. Помните об этом.

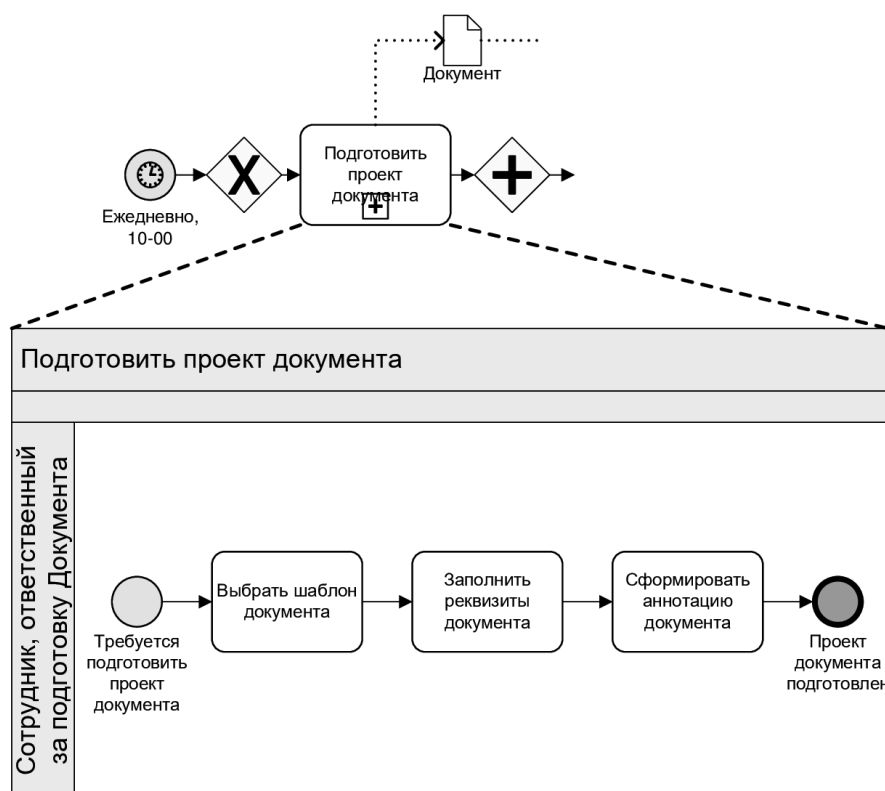


Рис. 39. Декомпозиция процесса в BPMN.

## 10. Описание сквозных процессов

*Архитектура процессов. Сквозные процессы. Участник процесса или другой процесс? Процессы как сервисы. Процессная грывжа.*

В организации много процессов и они связаны в систему. Понимание этой системы, архитектуры процессов позволяет получить максимальный эффект от применения методик процессного управления, в т.ч. автоматизации бизнес-процессов. В рамках архитектуры определяется множество процессов, и рассматриваются они на разных уровнях детализации.

*Наиболее важные для организации процессы являются сквозными или, другими словами, межфункциональными.*

Сквозной процесс — это процесс, исполнители которого находятся в различных функциональных подразделениях.

Кстати, если у вас нет четкой орг. структуры и должностей («Бирюзовая» организация), то не будет и сквозных процессов. Будут просто процессы, описанные при помощи ролей.

В данном разделе мы поговорим об описании сквозных процессов и проблемах, которые при этом возникают.

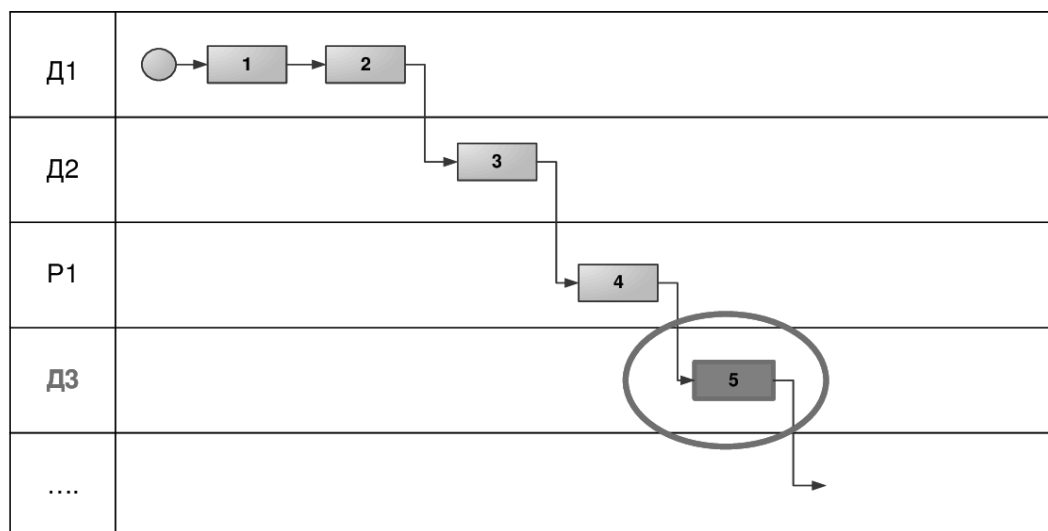
Рассмотрим пример, представленный на рис. 40. Мы описываем некоторый процесс. На первых трех дорожках показаны операции, которые выполняются сотрудниками нашего подразделения. В данном случае не важно, как названы дорожки — по должностям или ролям. Пусть Д1 — «Должность 1», «Р1» — «Роль 1» и т. п.

Д3 — «Должность 3» — это должность из другого структурного подразделения.

Нужно ли показывать операцию 5, которую выполняет данный сотрудник, на схеме? Если мы хотим создать модель сквозного бизнес-процесса (а не кусочек маленького, функционально раздробленного), то обязательно! Но как это сделать правильно?

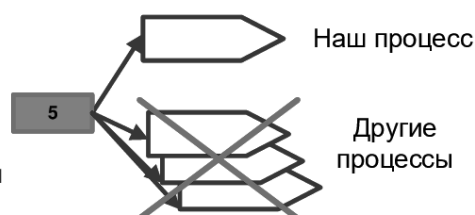
Оказывается, не все так просто, как может показаться на первый взгляд. Возможные варианты представлены на рис. 40.

*Первый вариант.* В случае если операция выполняется только для нашего процесса, ее результат не используется в других процессах и ее масштаб соответствует операциям нашего процесса, то дорожку «Должность 3» и операцию 5 целесообразно показывать на схеме процесса.



#### Вариант 1.

- 1) Операция выполняется только в этом процессе.
- 2) Результат выполнения операции не потребляется в других процессах.
- 3) Операция по масштабу соответствует всем остальным операциям нашего процесса.



#### Вариант 2.

- 1) Операция выполняется в других процессах.
- 2) Результат выполнения операции используется в других процессах.
- 3) Операция по масштабу существенно сложнее, чем операции нашего процесса.

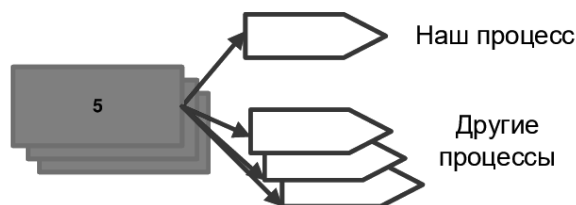


Рис. 40. Сквозной процесс.

Если на нашей схеме много операций, выполняемых в различных функциональных подразделениях, то мы получаем графическую схему классического сквозного процесса.

Кратко про масштаб операций. Сравните две операции на схеме одного процесса: «Подготовить бизнес-план» и «Отправить бизнес-план по e-mail руководителю». Есть разница? Все операции на схеме должны быть примерно одного масштаба. В противном случае, можно считать схему «сырой» и негодной для анализа и оптимизации процесса.

**Второй вариант.** Операция выполняется не только в нашем, но и в других процессах (не в экземплярах нашего процесса!). Типовые результаты ее выполнения используются другими процессами, а главное — операция су-

щественно сложнее, масштабнее по сравнению с другими операциями нашего процесса. Если ситуация такова, то такую *операцию целесообразно показать как другой, внешний процесс* (при помощи свернутого пула). В этом случае нужно будет воспользоваться промежуточными событиями отправки и получения сообщений. Фактически, мы будем из нашего процесса *инициировать (запускать) другой процесс как сервис*. После его выполнения, мы получим нужные нам результаты и продолжим выполнение нашего процесса.

Некоторые неопытные аналитики, восхитившись красотой рисования значков отправки/получения сообщений (конвертики прикольные), начинают рисовать их где надо и где не надо, сильно загромождая схему. Так делать не нужно. Прежде, чем показывать вместо простой операции другой процесс в виде свернутого пула, нужно внимательно проанализировать ситуацию. Вполне возможно, что никакие действия моделировать вообще не нужно — только показать входящий документ, который уже давно подготовлен и лежит в соответствующей базе данных. В общем, *от глубины понимания реального процесса зависит качество полученной на выходе графической схемы*.

На рис. 41 представлена третья возможная ситуация (Вариант 3), когда прямо на схеме подробно описано, как именно выполняется операция 5.

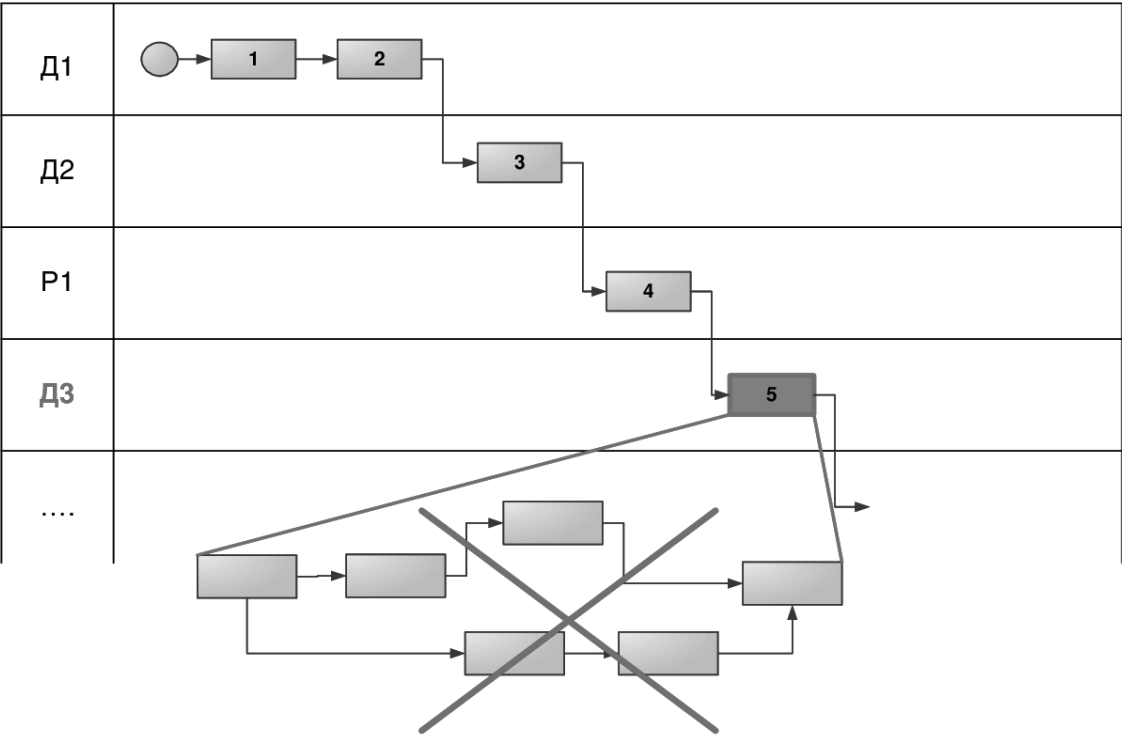


Рис. 41. Процессная грыжа.

Т.е. вместо запуска другого процесса, его начинают целиком перерисовывать на схеме процесса. При этом часто делают это с ошибками, т.к. не знают этот «другой» процесс в достаточной степени, либо просто спешат быстрее закончить работу над схемой. Это плохо.

Пример процессной грыжи. Нужно описать процесс закупки запасных частей для технического обслуживания оборудования. Вместо того чтобы показать процесс «Согласование и подписание договора» в виде отдельного процесса-сервиса (свернутый пул), его пытаются включить внутрь разрабатываемой схемы. Запустить этот процесс как сервис из других процессов будет нельзя. В каком-то другом процессе придется описывать согласование договора повторно и т. п. Это неудобно.

В случае если операцию процесса целесообразно запускать, как отдельный процесс, можно пойти двумя путями:

**Вариант А.** Вынести операцию 5 из пула и показать как отдельный процесс (свернутым пулом), запускаемый событием отправки сообщения.

**Вариант Б.** Описать операцию 5 в виде отдельного, сложного процесса в соответствующей части архитектуры процессов компании и использовать его на схеме процесса как ссылку на этот процесс<sup>1</sup>.

В обоих вариантах операция 5 будет отдельным процессом, который смогут использовать другие процессы, когда это потребуется.

С точки зрения документирования (создания регламента процесса), Вариант Б не так удобен, поскольку в регламент попадет (частично) описание типового процесса (процедуры), которое уже описано (или может быть описано) в другом регламенте. Впрочем, если использовать Business Studio, то проблему можно обойти путем настройки соответствующего шаблона отчета.

В целом, повторное использование типовых процессов — это хорошая практика.

---

<sup>1</sup> Например, в среде моделирования Business Studio это будет так называемый «Процесс-ссылка» или, другими словами, типовой процесс, который может быть использован в разных моделях.

## 11. Анализ качества графических схем процессов

### 11.1. Формальный анализ качества схемы

Представленных выше знаний достаточно, чтобы начать использовать нотацию BPMN для моделирования процессов. Но после того как вы сделаете схему процесса, не нужно сразу с радостным криком бежать показывать ее всем участникам рабочей группы проекта. Желательно сначала проверить качество схемы, найти и исправить грубые ошибки. Для этого вам поможет методика, представленная ниже.

Для проверки качества схемы можно использовать чек-лист — см. таблицу.

*Чек-лист «Формальная оценка качества процесса».*

№	Наименование требования	Комментарий	Кол-во
1	Корректность формулировок названий объектов на схеме		
2	Корректность описания входов/выходов		
3	Корректность описания событий		
4	Логические ошибки		
5	Физическая нереализуемость		
6	Возвраты в прошлое		
7	Не учтена необходимость обработки нескольких объектов		
8	Наличие «процессной грыжи»		
9	Однородность масштаба операций		
10	Аккуратность исполнения схемы. Визуальная наглядность		
	<b>Всего несоответствий:</b>		

Используется чек-лист следующим образом. Вы берете графическую схему и начинаете последовательно искать на ней ошибки из каждого раздела чек-листа. Если найдена ошибка, то делаете комментарий (что имен-



но нашли) и указываете количество ошибок. При этом целесообразно прямо на схеме выделять эти ошибки, например, ссылками с использованием шрифта красного цвета.

Чек-лист нужен, чтобы пройти по всем аспектам и ничего не забыть. После проверки нужно устранить выявленные ошибки и еще раз проверить модель по чек-листу.

Пункты 1—3. Примеры несоответствий из п. 1—3 таблицы представлены на рис. 42. События названы некорректно: 1) «Наступило утро» — неточная формулировка события-таймера; 2) «Бюджет» — некорректная формулировка — нельзя называть события в терминах документа; 3) «План закупок» — некорректная формулировка. 4) «Утвержден план закупок» — исполнитель не поймет, что ему делать, если план уже утвержден, а задание пришло к нему. Правильно поставить задачу «Утвердить план закупок» либо «Проверить утвержденный план закупок» (в зависимости от назначения операции).

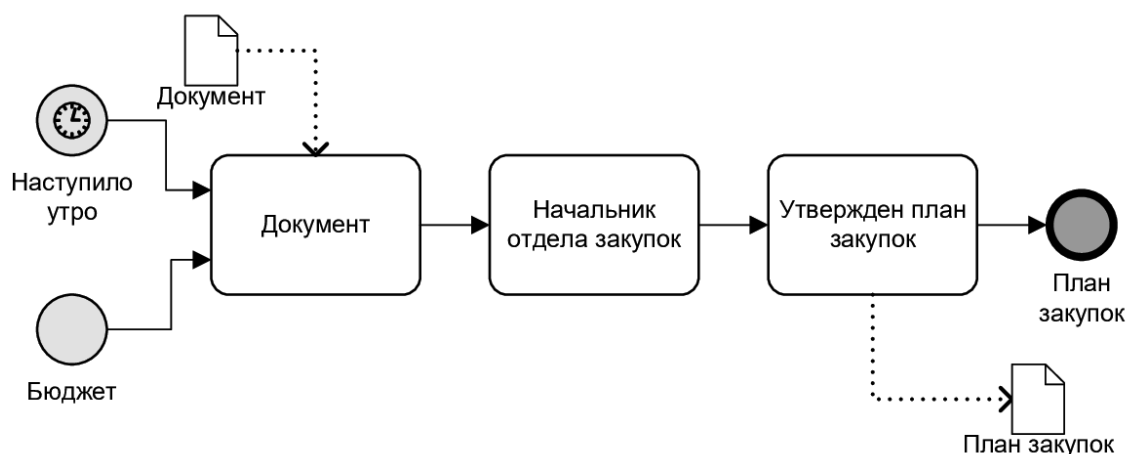


Рис. 42. Несоответствия в формулировках, входах/выходах, описании событий.

События «Наступило утро» и «Бюджет» запускают процесс без какого-либо оператора — неоднозначность — некорректное использование событий на схеме.

Операции процесса названы некорректно — нельзя называть их в терминах документов, подразделений или событий.

Документы «повисли» в воздухе — несоответствие — нужно показать откуда приходят, и куда попадают документы (другие процессы, базы данных).

Пункт 4. Примеры логических ошибок на схемах были представлены в разделе 3.

Пункт 5. Физическая нереализуемость — это ситуация, когда один исполнитель должен согласно схеме процесса одновременно выполнять два разных действия — см. рис. 43.

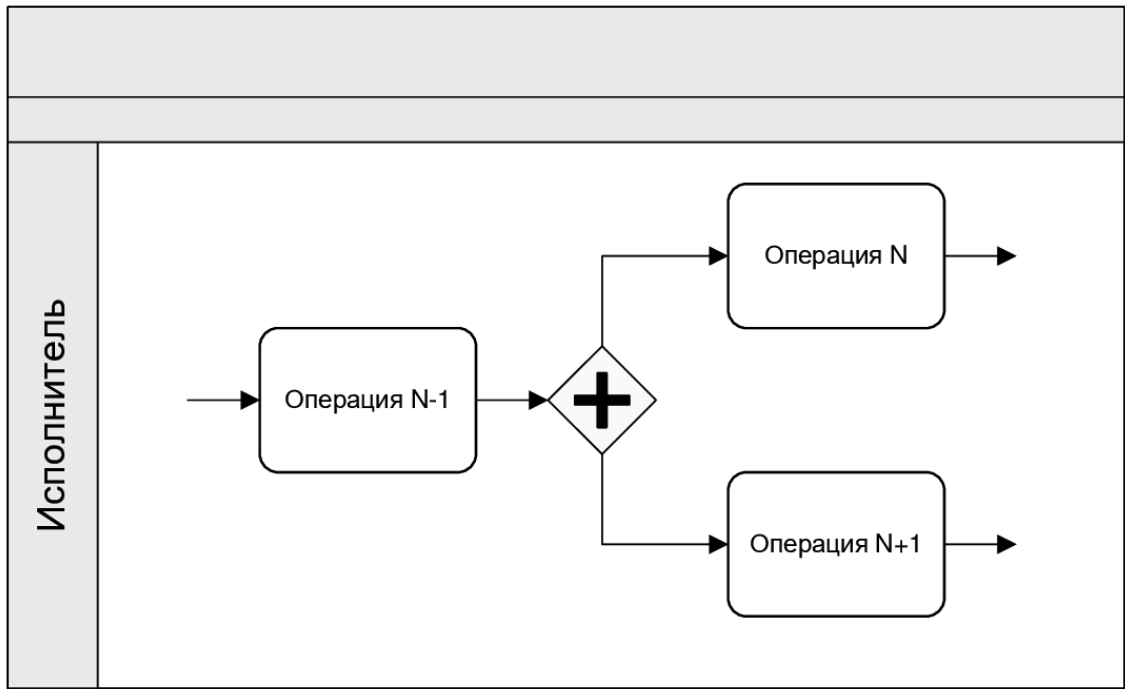


Рис. 43. Физическая нереализуемость.

Некоторые считают такую ситуацию допустимой, мол, исполнитель сам решит, какую операцию делать первой. Но это некорректно.

Как быть в такой ситуации? Вместо параллельного делать последовательное исполнение операций.

Пункт 6. Возврат в прошлое — довольно смешная ситуация, пример которой показан на рис. 44. Машину времени у нас пока не изобрели, поэтому лучше так не делать.

Исполнитель готовит отчет, который рассматривается руководителем после 14—00 (например, на планерке). Если есть замечания к отчету, то процесс возвращается на начало. Операция «Подготовить отчет за месяц» будет выполнена повторно, но попадет на планерку в 14—00 уже следующего дня (когда ее может уже и не быть).

Возможный правильный вариант выхода из данной ситуации показан в нижней части рисунка 44.

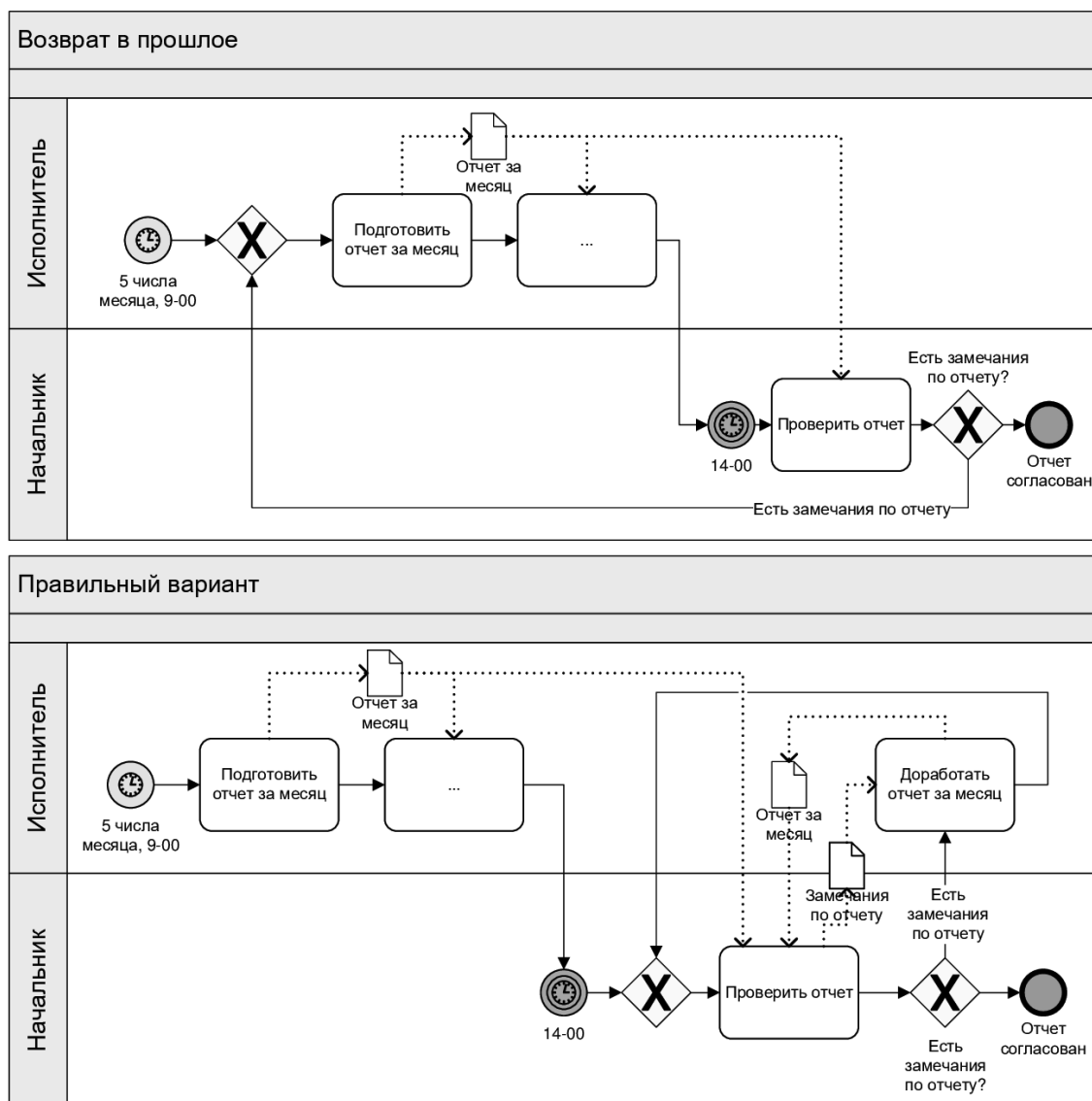


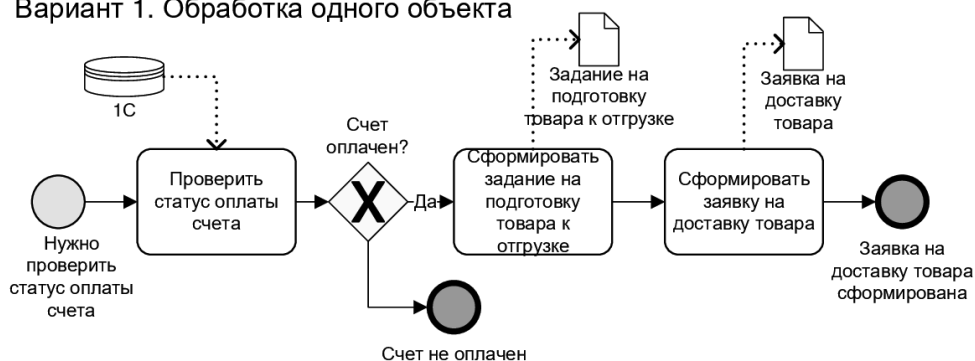
Рис. 44. Возврат в прошлое.

Пункт 7. Не учтена необходимость обработки нескольких объектов. Эту ситуацию довольно сложно объяснить человеку, который только что приступил к описанию процессов при помощи графических схем.

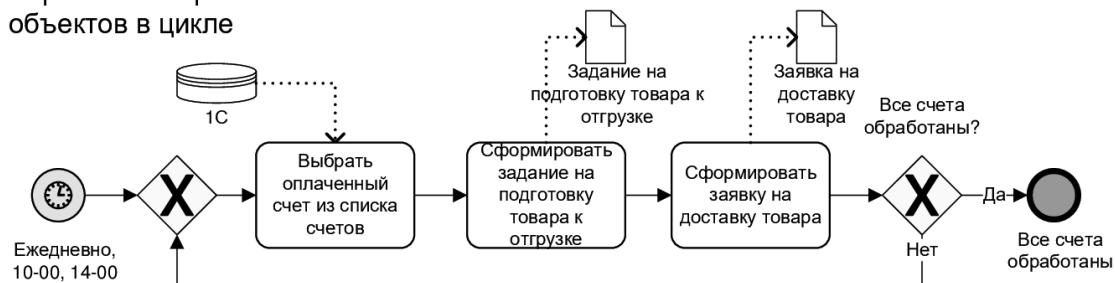
Проблема в том, что интуитивно человек проектирует процесс для какого-то одного объекта обработки. По факту таких объектов может быть несколько. Но поскольку процесс спроектирован для обработки одного объекта, то он запустится один раз (один единственный экземпляр), а остальные объекты останутся необработанными.

Рассмотрим пример, представленный на рис. 45.

### Вариант 1. Обработка одного объекта



### Вариант 2. Обработка нескольких объектов в цикле



### Вариант 3. Последовательная обработка нескольких объектов

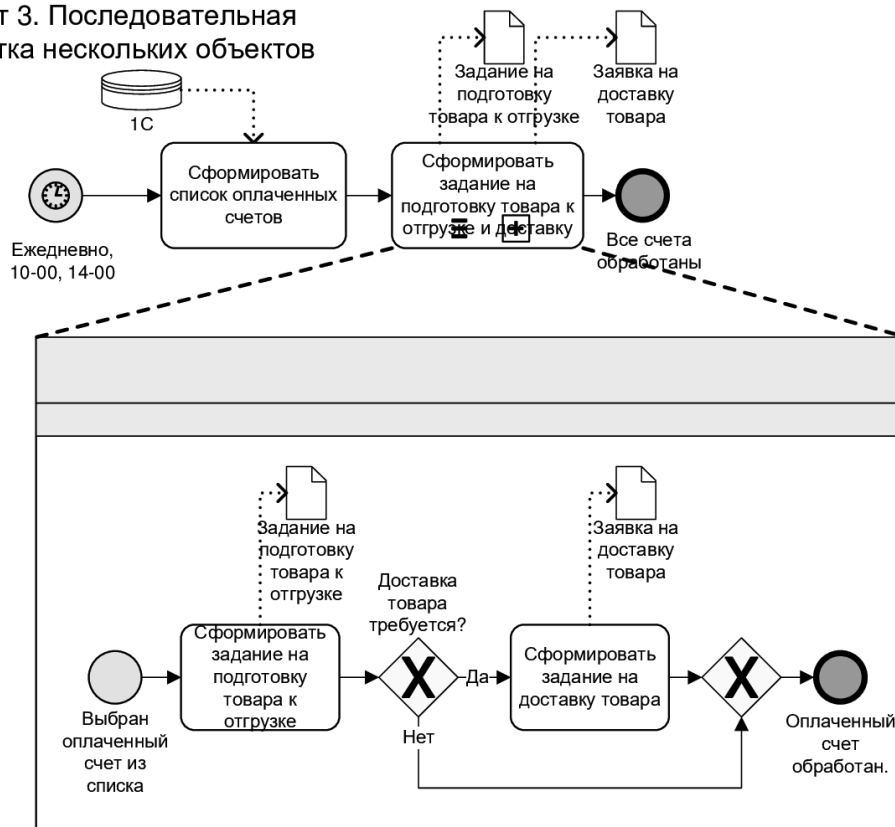


Рис. 45. Обработка нескольких объектов.

В некотором процессе продаж Исполнитель должен проверить оплату выставленных счетов. Если счета оплачены, то нужно дать задание на подготовку к отгрузке товара и сформировать заявку на доставку (если это необходимо). Работу надо выполнить *по каждому* оплаченному счету.

Вариант 1 процесса, представленного на рис. 45, выполнится только для одного счета. Все остальные ранее выставленные и оплаченные счета останутся необработанными.

Если цель создания графической схемы была показать общий порядок работы исполнителя, то задачу можно считать решенной (т.н. аналитическая схема). Если же наша цель заключалась в автоматизации процесса, то полученную схему нельзя считать корректной.

На рис. 45 показан Вариант 2 процесса, когда необходимая работа выполняется внутри цикла, причем по всем счетам.

Вариант 3 схемы, представленный на рис. 45, очень простой. Вы видите специальные маркеры в виде нескольких горизонтальных черточек — в нотации BPMN это последовательная обработка объектов (sequential multi-instance). Это означает, что операция «Сформировать задание на подготовку товара к отгрузке и доставку» выполнится столько раз, сколько счетов в списке оплаченных. На том же рис. 45 показана декомпозиция рассматриваемой операции.

Пункт 8. Наличие «процессной грыжи». Про «процессную грыжу» я говорил в разделе 10. Если вы ее диагностировали, то так оставлять категорически нельзя — нужно переделывать графическую схему процесса.

Пункт 9. Однородность масштаба операций. Если на схеме процесса присутствуют операции разного масштаба (неоднородность по масштабу), то схему нужно еще раз внимательно проанализировать и продумать возможность либо детализации ряда операций, либо укрупнения. Возможно, будет целесообразно сначала укрупнить схему, а потом декомпозировать операции на следующий уровень.

Пункт 10. Аккуратность и визуальная наглядность схемы. Красота — это страшная сила. Гораздо проще согласовать красивую и понятную схему, чем неряшливую и плохо читаемую. Поэтому призываю бороться за красоту и визуальную наглядность схем. Как этого добиться? Графически элементы должны быть одного размера и корректно расположены, названия не перекрываться, шрифт должен читаться и т. п. Требования довольно очевидные.

Для самоконтроля рекомендую проверить себя — найти ошибки на схеме процесса, представленной на рис. 46.

Правильные ответы представлены в конце данной книжки.

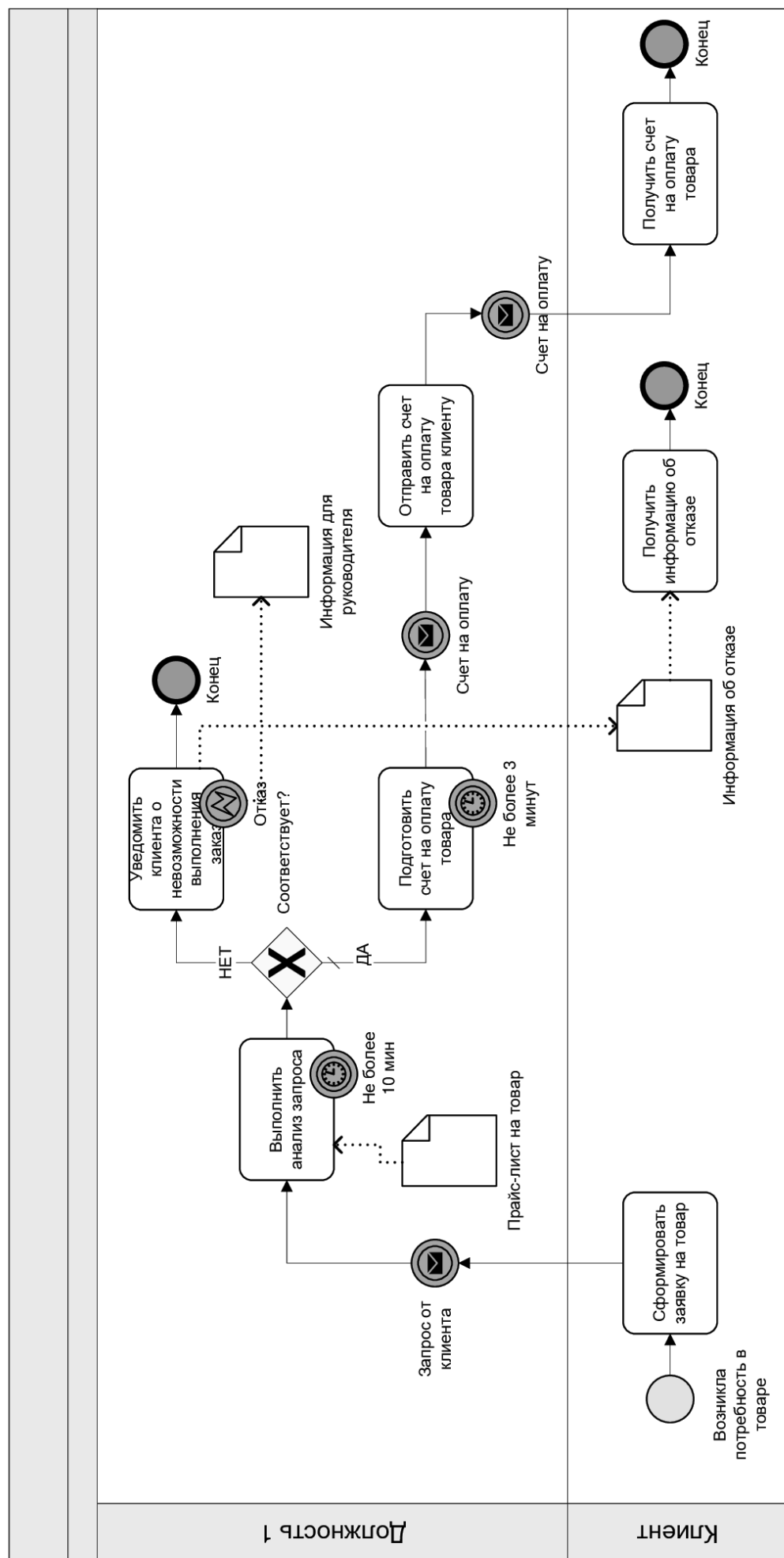


Рис. 46. Найдите ошибки на схеме.

## 11.2. Содержательный анализ схемы процесса

Поговорим теперь о содержательном анализе схемы процесса. Детальный анализ процесса должен делаться с использованием количественных данных. Но разработанную схему можно сразу начать анализировать, используя метод «пристального взгляда». При этом целесообразно делать это по чек-листу (представлен ниже), чтобы не пропустить важные аспекты.

Чек-лист «Содержательный анализ процесса».

№	Наименование	Комментарий
1	Операции, создающие ценность	Проверяется наличие на схеме процесса операций, добавляющих ценность (для клиента и/или организации)
2	Операции, не создающие ценность	Проверяется наличие на схеме процесса операций, не добавляющих ценность («муда»).
3	Передача результата потребителю	Проверяется, завершается ли процесс передачей результата следующему процессу (потребителю).
4	Возвраты	Выявляются возвраты и определяются их причины. Если возврат технологический (см. Вариант 2 на рис 45), то это допустимо. Остальные возвраты всегда являются признаками неэффективно выполняемого процесса.
5	Дублирование операций	Проверяется прямое, либо косвенное дублирование работы (может быть выявлено по дублированию результатов выполнения операций).
6	Чрезмерный контроль	Проверяется целесообразность операций контроля, в т.ч. сверки, согласования, утверждения и т.п.
7	Узкие места	Проверяется наличие узких мест (в первую очередь по ресурсам)
8	Автоматизация	Проверяется наличие автоматизированных операций процесса, выявляются факты передачи данных из одной информационной системы в другую (и связанные с этим проблемы).
9	Роботизация	Проверяется наличие роботизированных операций. Выявляется возможность роботизации операций.

Отдельно остановимся на ситуации чрезмерного контроля, т.к. она весьма распространена, особенно в гос. компаниях.

Пример схемы процесса, содержащего чрезмерный контроль, представлен на рис. 47. Исполнитель создает документ, Контролер его проверяет, Начальник согласует, а Большой начальник утверждает.

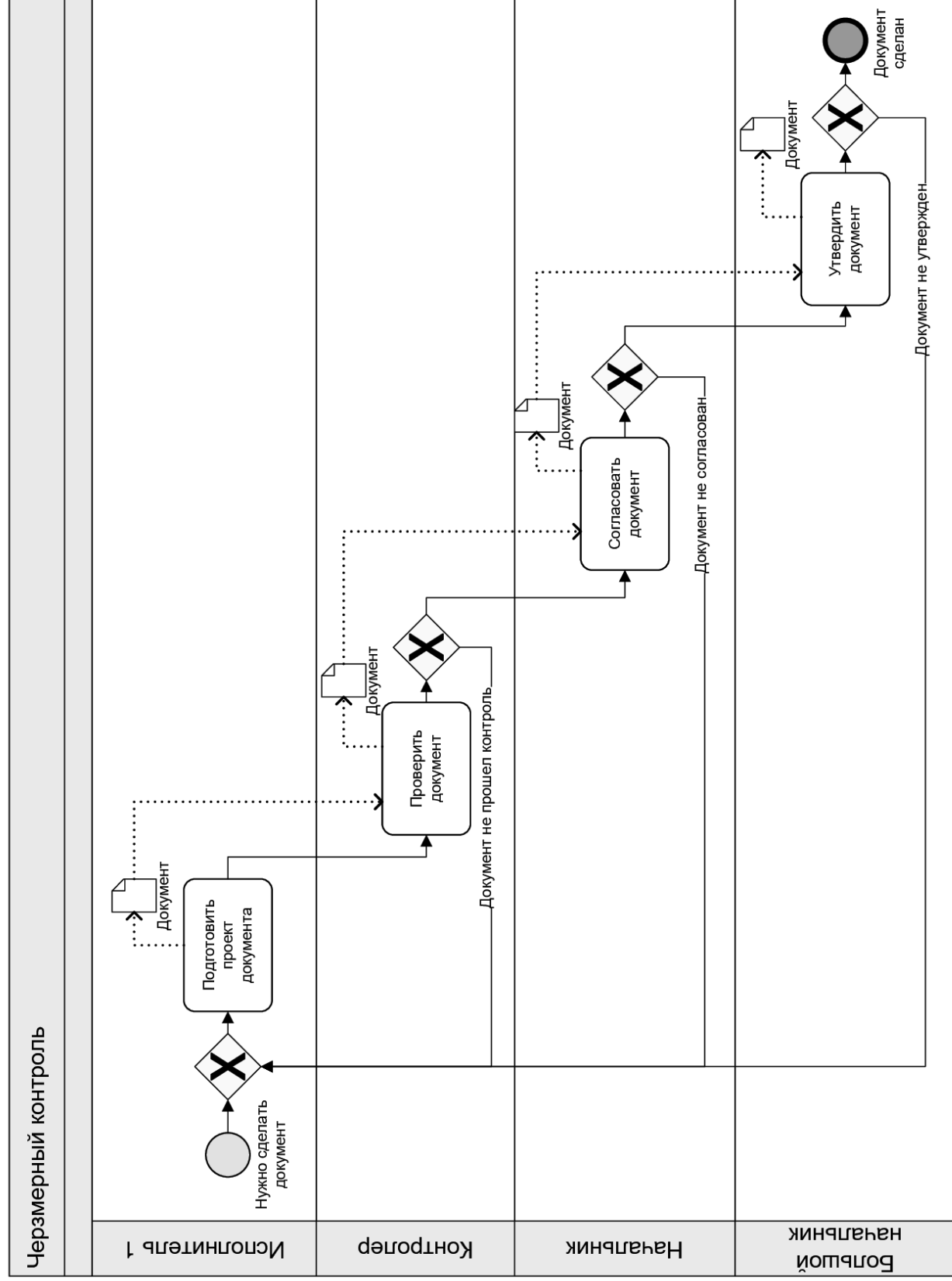


Рис. 47. Чермерный контроль.



Вообще, чрезмерный контроль является следствием постоянного страха, взаимного недоверия и разобщенности, которые характерны для жестких иерархических функциональных структур.

В любом случае, чрезмерный контроль, выполняемый людьми, приводит к размытию ответственности, снижению качества результата и росту потерь (неоправданных затрат — «муды»). Если вы столкнулись с таким процессом, то нужно отдельно проанализировать каждый шаг и выявить, *что конкретно делается, по какой методике, с использованием каких данных, какая ценность добавляется к общему результату процесса*. Возможно, это поможет реорганизовать процесс и сделать его гораздо эффективнее.

На рис. 48 показан пример процесса некоторой компании (схема мелкая, но важно не ее содержание, а результаты анализа).

Анализ схемы сразу позволяет выявить следующие особенности процесса:

- на схеме 5 возвратов.
- количество повторяемых шагов при однократном возврате: 14.
- количество информационных систем: 5 (MS Excel, 1С, АСУ «Транспорт», «Финансы», SAP).
- доля контрольных операций: 57% (сверки, согласования, визирования).
- доля операций по корректировке данных/документов: 19%.

Итого, доля операций процесса, не добавляющих ценность, составляет 76%.

Такой процесс сложно назвать эффективным. Он явно нуждается в оптимизации.



## 12. Комплексный пример описания группы процессов

На рис. 49 показан процесс Подачи заявки на оплату. Инициатор платежа находит поставщика какого-то нужного ему товара (материала и т.п.), получает и акцептует счет на оплату. Эти действия остаются за кадром — процесс запускается с момента, когда инициатору нужно оплатить счет. Он запускает новый экземпляр процесса, оформляет заявку на оплату, в т.ч. указывает статью бюджета и прикладывает скан-копию счета на оплату.

Далее выполняется последовательное согласование заявки. Как Начальник инициатора, так и Экономист ФЭО, оба могут отправить заявку обратно Инициатору на доработку, либо отклонить заявку. В случае если заявка отклонена, Инициатор уведомляется об этом, выполняя операцию «Принять информацию по отклонению заявки». Это важно, так как иначе он так и не узнает, чем завершился процесс, который он инициировал.

В случае, если заявка вне бюджета, но срочная и важная, то может потребоваться дополнительное обоснование. В этом случае Экономист запрашивает его у Инициатора и возникает новый цикл согласования.

Если заявка согласована Начальником и Экономистом ФЭО, то Экономист заносит заявку в график платежей на неделю (это может быть, например, файл MS Excel, размещенный на сервере компании).

Далее процесс ждет получения сообщения из процесса «Утверждение графика платежей». Это нужно для того, чтобы Инициатор был уведомлен о статусе своего платежа — он может быть поставлен в оплату или передвинут на следующую неделю и т. п. Затем процесс снова ждет получения сообщения из процесса «Оплата счетов». Это нужно для того, чтобы Инициатор оперативно получил информацию о том, что счет оплачен.

На рис. 50 показан процесс «Утверждение графика платежей». Он запускается один раз в неделю по таймеру. По ходу выполнения данный процесс инициирует другой процесс «Оплата счетов» путем отправки сообщения «График платежей передан на оплату».

Далее Экономист ФЭО отправляет информацию по статусу *каждого платежа в соответствующие экземпляры процесса* «Подача заявки на оплату».

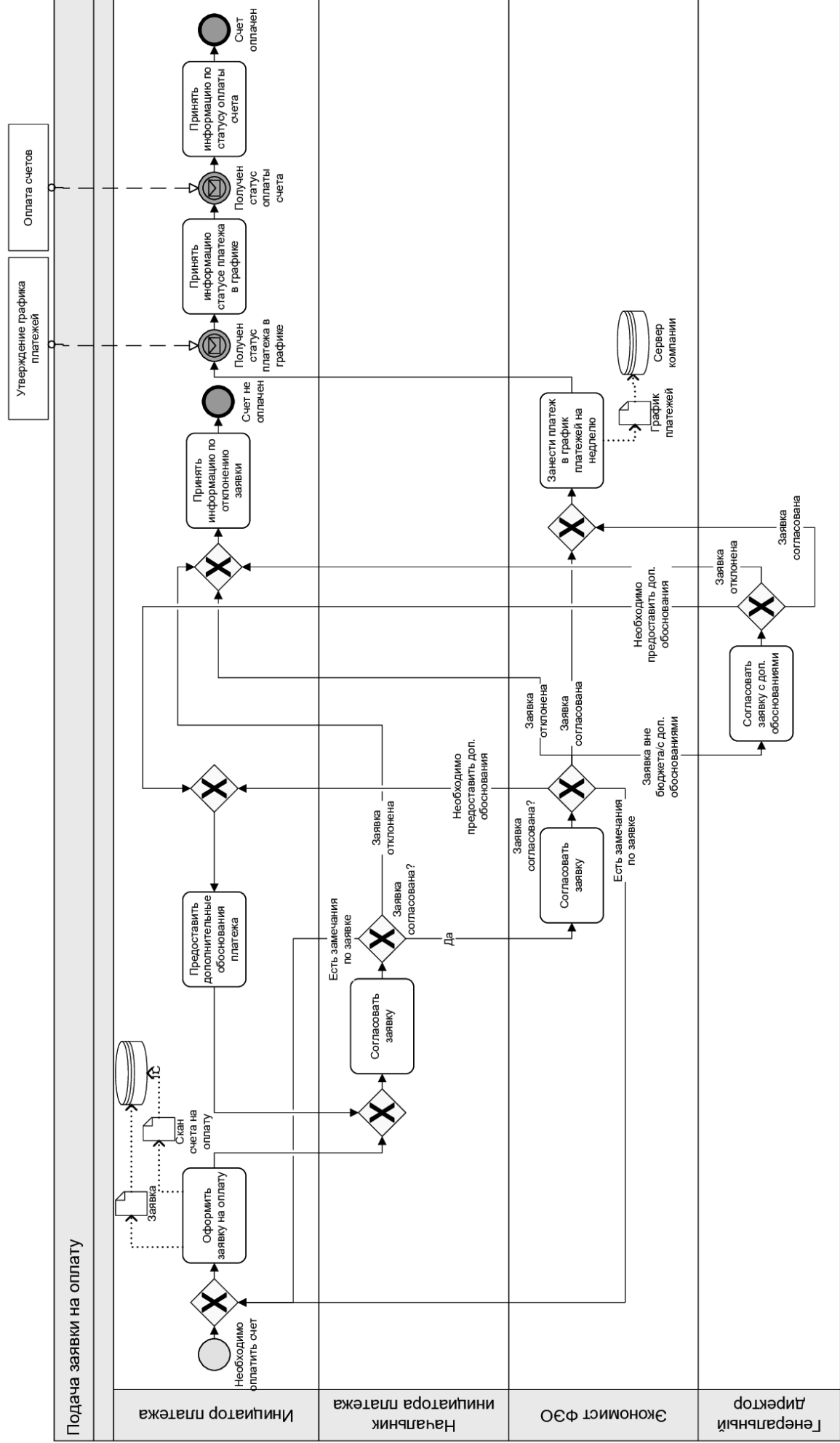


Рис. 49. Процесс подачи заявки на оплату.

На рис. 51 показан процесс «Оплата счетов». Как я отмечал выше, он запускается на исполнение при получении сообщения от процесса «Утверждение графика платежей». Завершается процесс отправкой информации об оплате конкретных счетов в конкретные экземпляры процесса «Подача заявки на оплату».

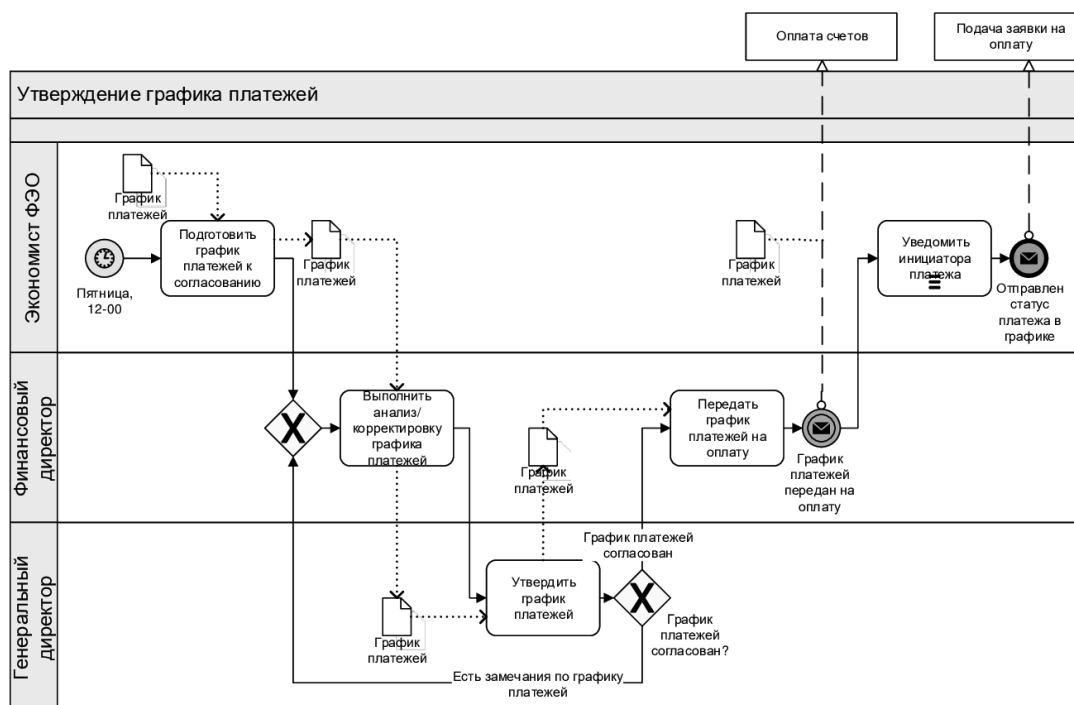


Рис. 50. Процесс согласования графика платежей на неделю.

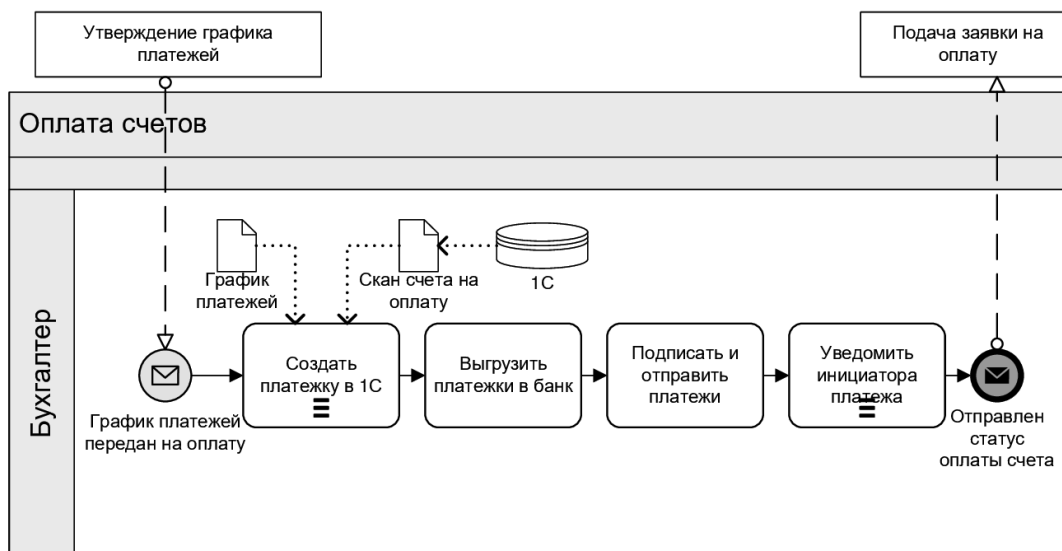


Рис. 51. Процесс оплаты.

Итак, на рис. 49—51 показана группа связанных между собой процессов. Первый из них стартует произвольным образом, по необходимости. Второй — по таймеру. Третий — путем получения сообщения. Все процессы связаны между собой. Самым сложным аспектом для восприятия в данном случае является тот факт, что процессы «Утверждение графика платежей» и «Оплата счетов» передают информацию в разные экземпляры процесса «Подача заявки на оплату».

## 13. Организация работы по описанию процесса

Описание процессов компании — работа серьезная.

В зависимости от размера компании и масштаба проекта могут применяться различные подходы к организации работы по описанию процессов.

Например, можно:

- организовать сбор информации путем проведения интервью (выполняют профессиональные бизнес-аналитики);
- проводить моделирующие сессии с привлечением участников и владельцев процессов;
- обучить сотрудников подразделений моделировать процессы и потом контролировать качество;
- прочие.

Если лично перед вами поставлена задача сформировать графическую схему процесса в нотации BPMN, то можно использовать, например, следующую методику. Когда вы приобретете опыт моделирования, то скорректируете эту методику на основе своего опыта.

### *1. Определите контекст процесса.*

Сформулируйте название процесса, лучше глаголом. Определите цель описания процесса — это важно. Определите точку зрения на процесс (лучше описывать процесс с точки зрения его владельца, но возможны и другие подходы). Определите основные входы/выходы. Определите стартовые и завершающие события. Определите необходимых участников процесса (должности, роли) с учетом точки зрения на процесс.

### *2. Подготовьте диаграмму процесса.*

Создайте модель процесса в нотации BPMN, используя подходящий инструмент (например, MS Visio или специальное программное обеспечение для бизнес-моделирования). Создайте дорожки при помощи должностей (ролей). Создайте свернутые пулы для процессов, с которыми взаимодействует процесс.

Если сотрудник создает по ходу процесса уникальный результат, нужный только для получения результата описываемого процесса в целом,

этого сотрудника можно показать как участника процесса (должность или роль). Если результат работы сотрудника не уникальный, а типовой и используется в других процессах, то нужно выявить процесс, при выполнении которого создается результат, и показать этот процесс на схеме как свернутый пул. Такого сотрудника и соответствующую дорожку на схеме в этом случае не показывать. Измените размеры схемы с учетом предварительного понимания сложности процесса.

### ***3. Определите операции процесса.***

Составьте перечень операций процесса, выполняемых его участниками. Для определения операций процесса можно: а) ознакомиться с внутренними и внешними нормативно-методическими документами по процессу; б) вспомнить, что вам известно о выполнении процесса; в) провести встречи с участниками (исполнителями) и задать им вопросы, как выполняется процесс. Далее нужно показать эти операции на схеме процесса. Между ними необходимо оставить достаточно большое место для событий и операторов логики. Общее количество операций на схеме — не более 12–15.

Старайтесь не детализировать операции без практической необходимости (Вместо трех операций «Получить документ», «Проверить документ», «Отправить документ» делайте одну — «Проверить документ» — по входам/выходам всё и так будет понятно).

### ***4. Опишите взаимодействие между процессами.***

Если процесс может быть инициирован в начале или управляться по ходу выполнения другими процессами (события поступления сообщений) или запускает процессы (события отправки сообщений), то укажите эти события в соответствующей части схемы процесса.

### ***5. Опишите логику процесса.***

Определите точки ветвления по ходу процесса. Добавьте на схему необходимые операторы логики («ИЛИ», «И»). Свяжите операции процесса, шлюзы и события между собой при помощи стрелок типа sequence.

### ***6. Опишите информационные потоки.***

Определите и покажите на схеме информационные потоки: а) внутри процесса; б) между процессами.

### ***7. Визуально отредактируйте схему.***

Отредактируйте схему так, чтобы она была понятной с первого взгляда, наглядной и красивой. Схема должна легко читаться человеком с нормальным зрением при распечатке в формате А4.



### 8. Проверьте схему процесса по чек-листу.

Выполните проверку схемы по чек-листу. Внесите изменения в схему и перейдите на шаг 7.

Надо быть готовым к тому, что качественная, соответствующая реальному процессу схема может быть получена только после 2–3 итераций описания и согласования.

Если вашу схему согласовали с первого раза, то вы либо супер-профессионал, либо эта работа в компании никому не нужна.

В завершение хочу напомнить про важность формулировки цели создания модели и точки зрения на нее.

Посмотрите на рис. 52:

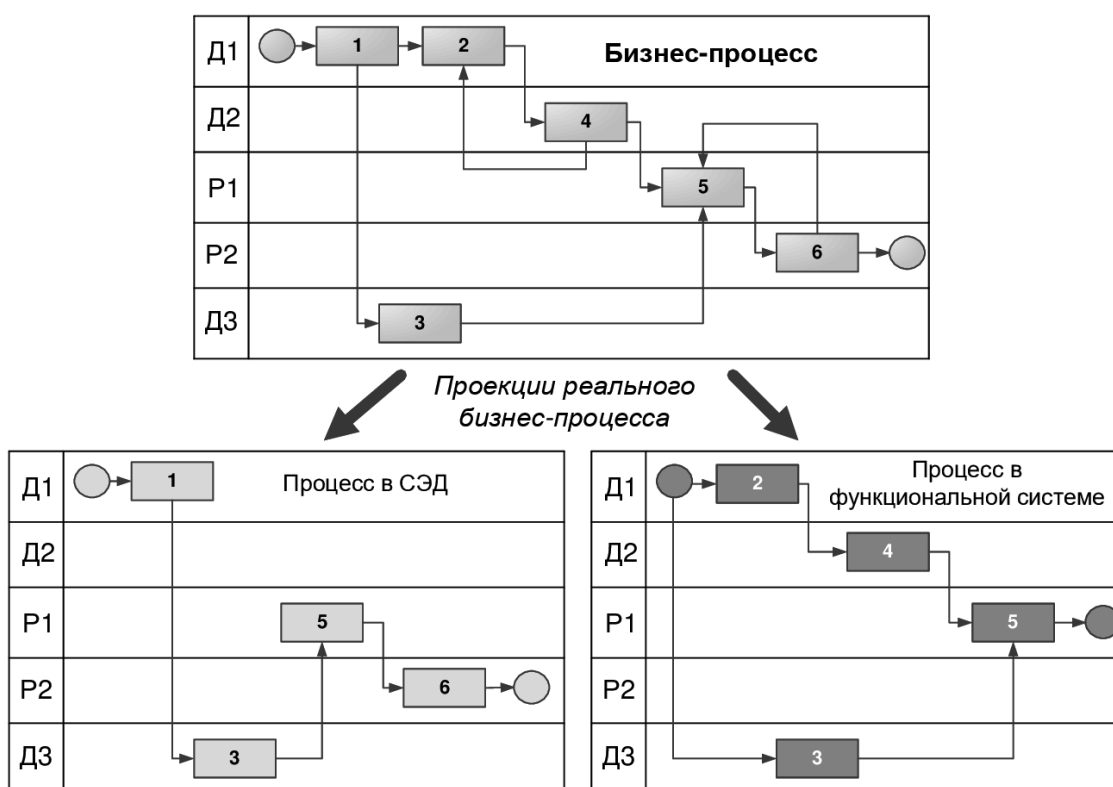


Рис. 52. Различные точки зрения на процесс.

Вверху рисунка представлена схема реального бизнес-процесса. Она включает ВСЕ необходимые действия, выполняемые различными исполнителями в разных информационных системах.

С точки зрения специалиста, настраивающего систему электронного документооборота (СЭД), процесс будет выглядеть, например, как показано на рисунке слева внизу.

С точки зрения сотрудника, внедряющего функциональную информационную систему (например, поддержки ведения бухучета и т.п.), процесс будет выглядеть как показано справа внизу.

Получилось три разных схемы! Помните, для чего вы создаете графическую модель процесса — это важно.

Удачи вам при описании бизнес-процессов в нотации BPMN!

## 14. Что почитать по BPMN и процессному подходу?

Для более подробного ознакомления с BPMN и процессным подходом в целом можно использовать следующие материалы:

### *По нотации BPMN:*

1. INTERNATIONAL STANDARD ISO/IEC 19510 First edition 2013-07-01 Information technology — Object Management Group Business Process Model and Notation.
2. Фёдоров И. Г. Моделирование бизнес-процессов в нотации BPMN 2.0., М.: МЭСИ, 2013 г.
3. Системы управления бизнес-процессами и административными регламентами. Андрей Михеев ООО «Альт Линукс» 2011 г.
4. Свод знаний по управлению бизнес-процессами. BPM СВОК 3.0. М.: Альпина Паблишер, 2016.
5. Коалиция OGM/BPMI, графическая нотация BPMN. [www.bpmn.org](http://www.bpmn.org).
6. Интерактивный курс по обучению и сертификации BPMS Bizagi <http://elearning.bizagi.com/course/>.

### *В целом по процессному управлению:*

1. Репин В. В. Бизнес-процессы. Моделирование, внедрение, управление. М.: Манн, Иванов и Фербер, 2014.
2. Свод знаний по управлению бизнес-процессами. BPM СВОК 3.0. М.: Альпина Паблишер, 2016.

### *По регламентации и стандартизации бизнес-процессов:*

1. Репин В. В. Бизнес по правилам: регламенты должны работать. М.: ИНФРА-М, 2018.

## 15. Правильные ответы на задачи

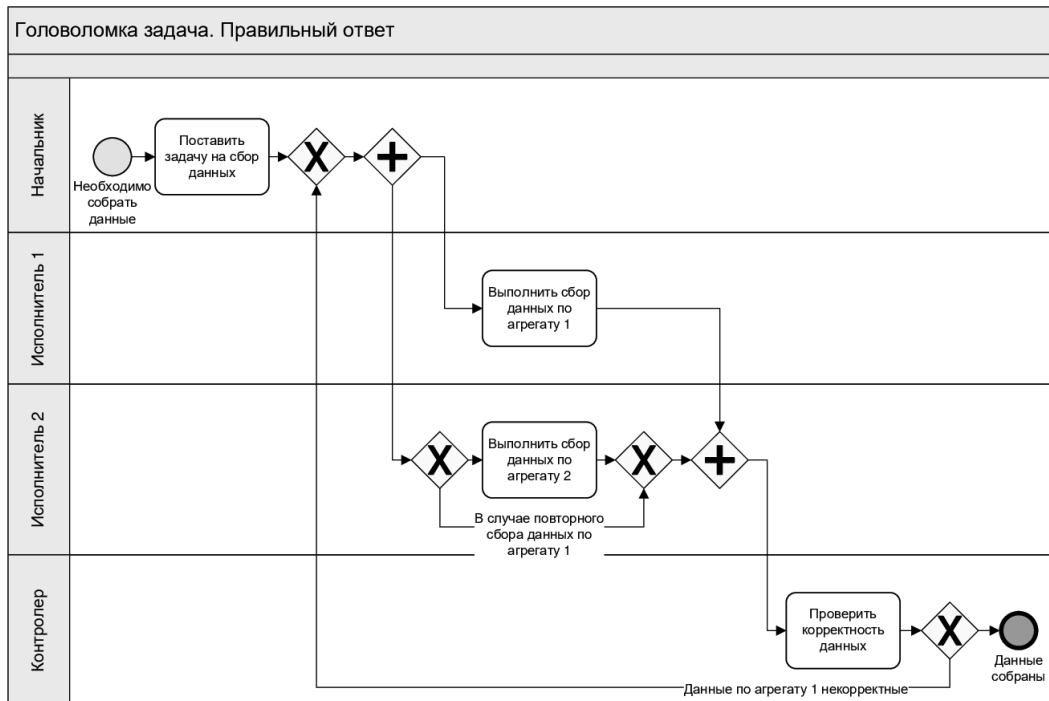


Рис. 53. Возврат в одну из параллельных веток. Правильный вариант А.

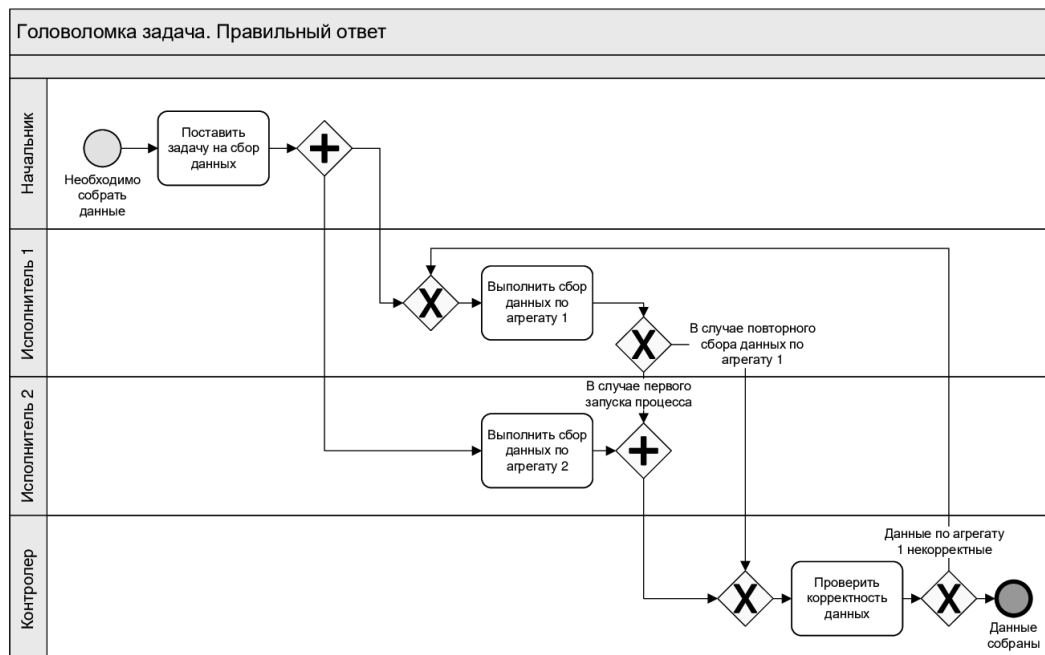


Рис. 54. Возврат в одну из параллельных веток. Правильный вариант Б.

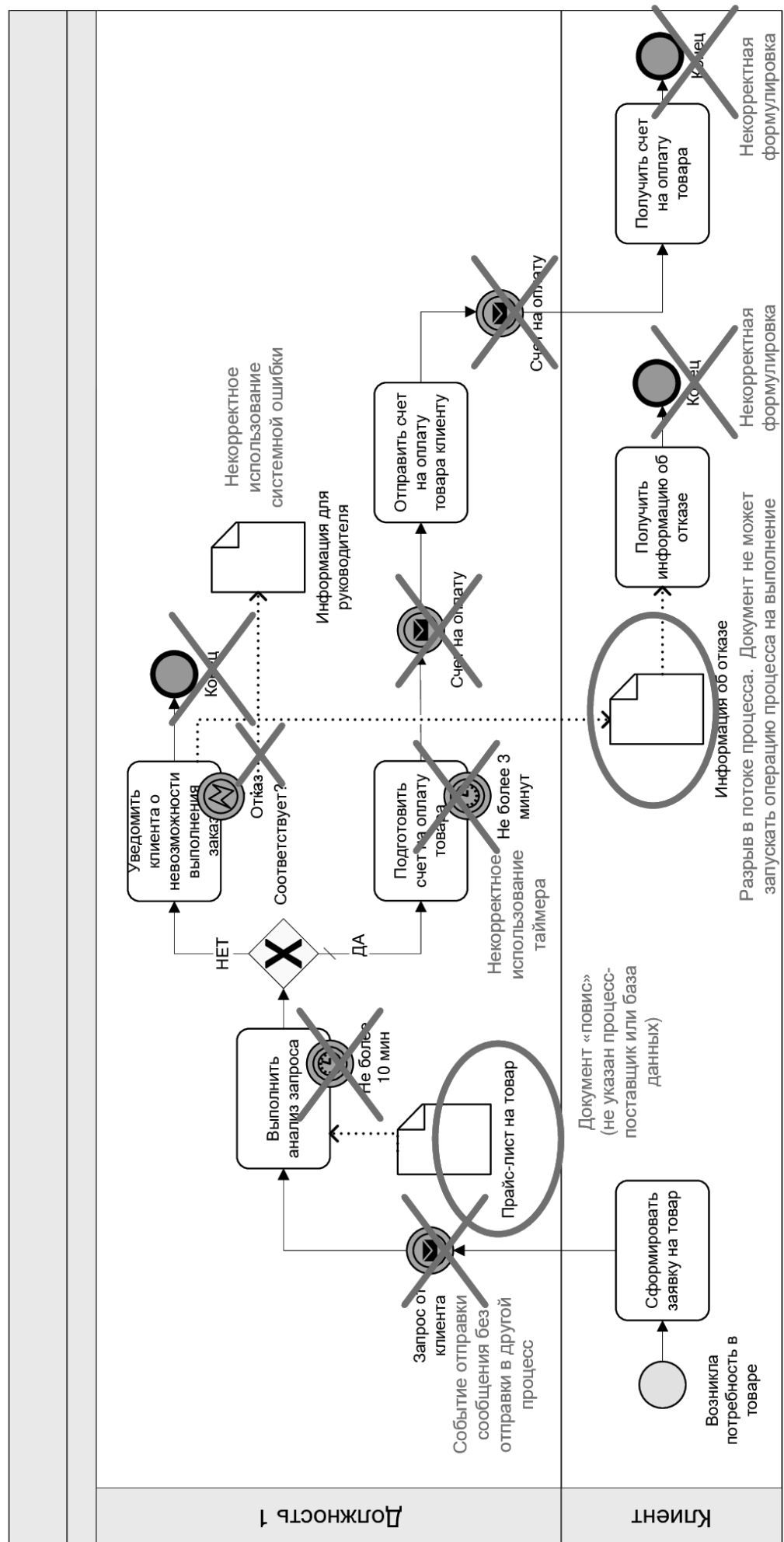






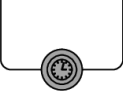








Рис. 55. Ошибки на схеме процесса.

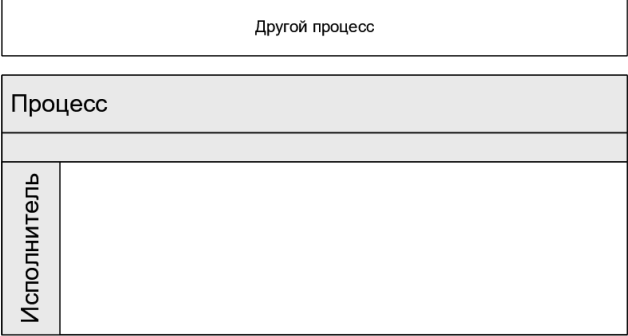
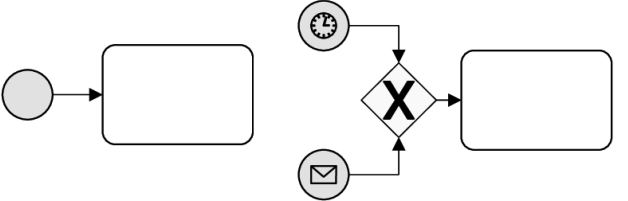
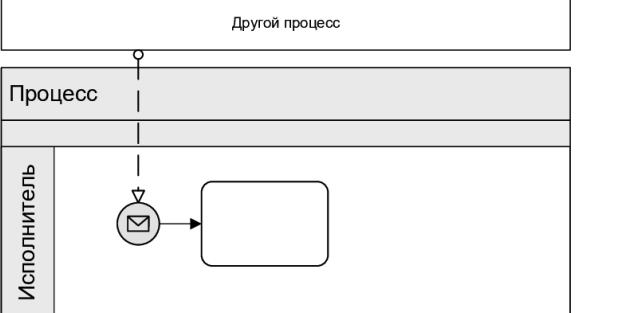
## 16. Приложение №1. Основные элементы нотации BPMN

### 16.1. Основные элементы нотации BPMN

Элемент	Пиктограмма	Описание
Стартовое событие		Стартовое событие неопределенного типа.
		Событие-сообщение (старт процесс по получению сообщения из другого процесса (свернутого пула)).
		Стартовое событие-таймер.
Промежуточное событие		Промежуточное событие-сообщение обработчик (управляющее воздействие из другого процесса).
		Промежуточное событие-сообщение инициатор (вызов события-обработчика).
		Промежуточное событие-таймер.
Граничное событие		Граничное событие-таймер
Конечное событие		Завершающее процесс событие неопределенного типа.
		Завершающее процесс событие-сообщение инициатор (вызов события-обработчика).
Операция процесса		Операция процесса (задача).
		Подпроцесс – декомпозированная операция процесса.
Шлюзы		Исключающее «ИЛИ»
		Неисключающее «ИЛИ»

		<p>«И»</p> <p>Исключающее «ИЛИ» с ожиданием одного из возможных событий</p> <p>Исключающее «ИЛИ» для запуска процесса</p> <p>«И» для запуска процесса</p>
Потоки		<p>Поток управления (sequence)</p> <p>Сообщение (message)</p> <p>Информационный поток (association)</p>

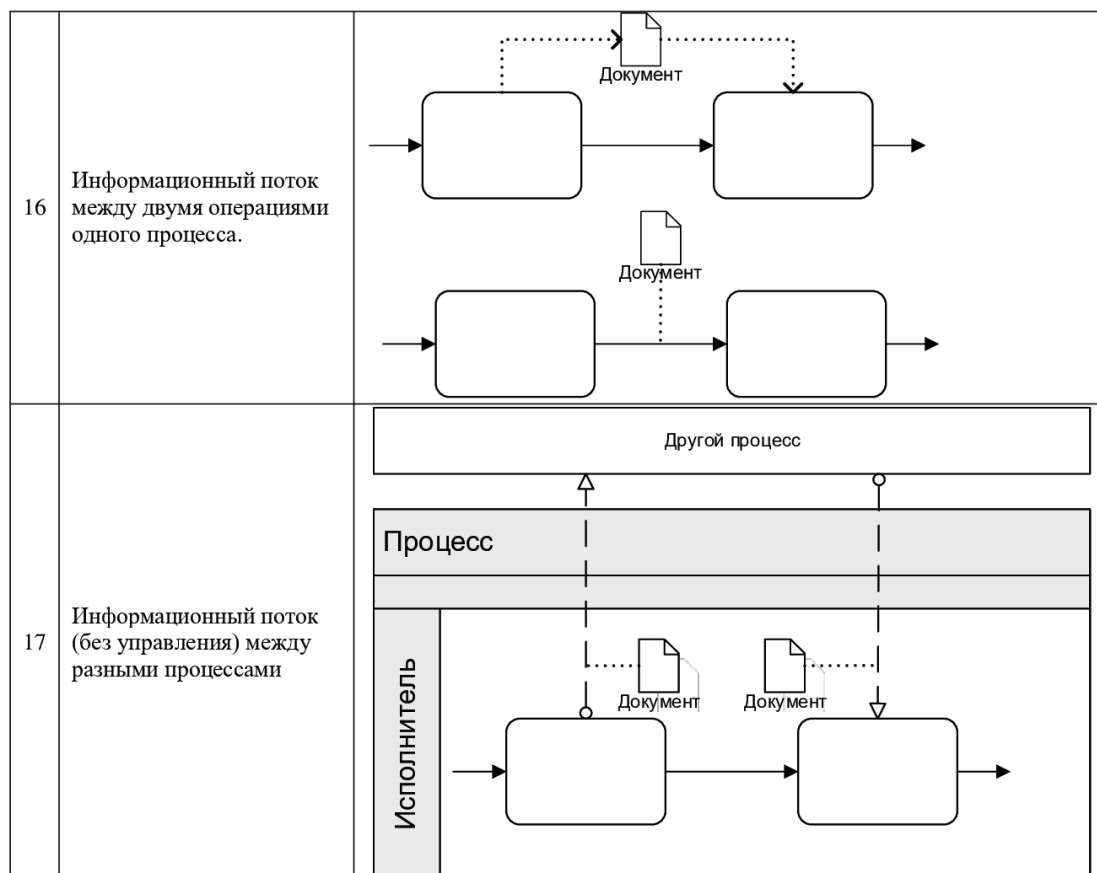
## 16.2. Примеры использования элементов нотации BPMN

№	Описание фрагмента нотации	Графический пример
1	Пул предназначен для отображения процесса. Дорожка предназначена для отображения ролей и должностей - исполнителей процессов. Свернутый пул обозначает внешний (по отношению к текущей диаграмме) процесс.	
2	Стартовое событие (без маркера) и операция процесса. У процесса может быть одно начальное событие или несколько начальных событий объединенных логикой.	
3	Стартовое событие, связанное с поступлением информации в процесс из другого процесса (инициация процесса другим процессом)	

4	Стартовое событие-таймер	<div>Процесс</div> <div>Исполнитель</div>
5	Промежуточное событие процесса, воздействующее на другой процесс путем отправки сообщения.	<div>Другой процесс</div> <div>Процесс</div> <div>Исполнитель</div>
6	Промежуточное событие процесса, связанное с получением сообщения из другого процесса (управляющее воздействие)	<div>Другой процесс</div> <div>Процесс</div> <div>Исполнитель</div>
7	Промежуточное событие-таймер	<div>Процесс</div> <div>Исполнитель</div>
8	Граничное событие-таймер	
9	Событие неопределенного типа завершающее процесс	



10	Завершающее процесс событие, воздействующее на другой процесс путем отправки сообщения.	
11	Шлюз «И», ветвление на несколько параллельных потоков.	
12	Шлюз «И», слияние нескольких параллельных потоков.	
13	Шлюз исключающее «ИЛИ», ветвление на несколько альтернативных потоков.	
14	Шлюз не исключающее «ИЛИ», ветвление на несколько потоков.	
15	Сходящийся шлюз исключающее «ИЛИ»	





Владимир Репин, консультант по управлению, тренер, к.т.н., доцент, Генеральный директор ООО «Владимир Репин Менеджмент», член АВРМР Russian Chapter.

Провел 365 семинаров и тренингов в 67 городах России и стран СНГ. Обучил 4529 человек, автор 6 книг по бизнес-процессам.

Область профессиональных интересов — внедрение процессного подхода к управлению, в том числе:

- построение системы управления бизнес-процессами компании;
- разработка архитектуры бизнес-процессов;
- комплексная поставка и настройка программного продукта Business Studio, проведение тренингов по системе;
- моделирование, анализ и оптимизация процессов;
- оптимизация и автоматизация бизнес-процессов на платформе Elma;
- регламентация и стандартизация процессов;
- разработка целей и показателей для управления процессами;
- обучение руководителей и специалистов методам процессного управления;
- организация работы Процессного офиса;
- экспертно-методическое сопровождение проектов внедрения процессного управления (экспертиза моделей процессов и проектов регламентов, проведение скайп-консультаций и проч.)

С автором книги можно связаться по e-mail: [info@bpm3.ru](mailto:info@bpm3.ru).

**Владимир Репин**

**Моделирование бизнес-процессов в нотации BPMN**  
Пособие для начинающих. Часть I