

Логические операторы

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

Содержание

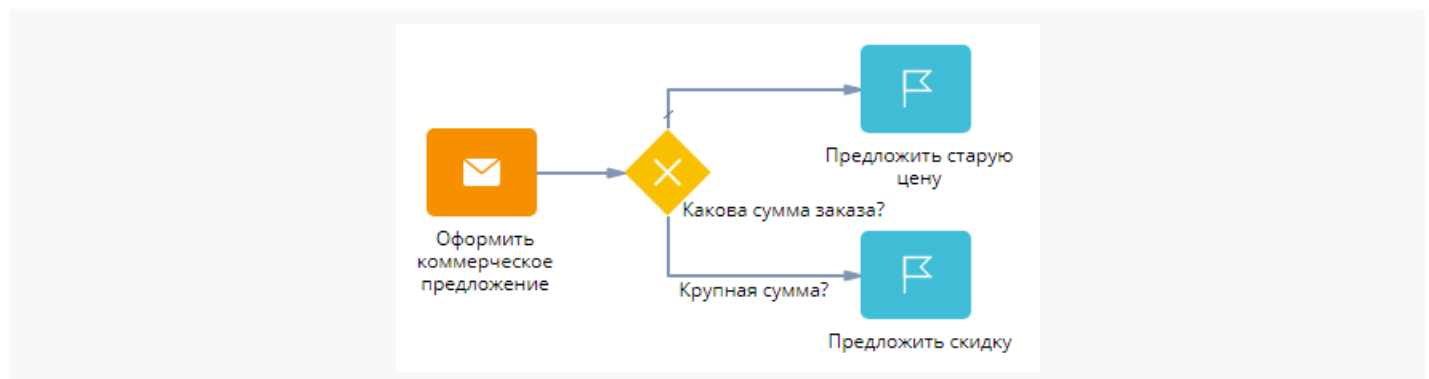
Элемент процесса [Исключающее “ИЛИ”]	4
Элемент процесса [Включающее “ИЛИ”]	5
Элемент процесса [Логическое “И”]	5
Элемент процесса [Исключающее “ИЛИ” по событиям]	6

Элемент процесса [Исключающее “ИЛИ”]

ПРОДУКТЫ: **ВСЕ ПРОДУКТЫ**

Используйте логический оператор [*Исключающее “ИЛИ”*] в случаях, когда из всех возможных альтернативных потоков процесса может быть выбран только один. Например, товар клиенту можно продавать либо по стандартной цене, либо со скидкой, в зависимости от суммы заказа ([Рис. 1](#)).

Рис. 1 — Пример использования логического оператора [*Исключающее “ИЛИ”*] для ветвления



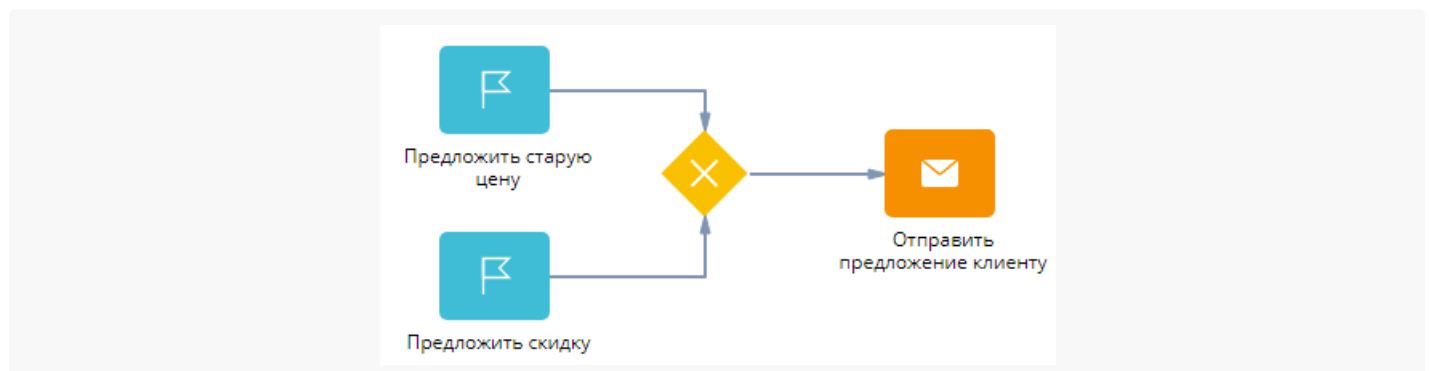
В этом случае будет выполнено только одно действие: либо “Предоставить скидку”, либо “Предложить стандартную цену”.

При ветвлении оператор [*Исключающее “ИЛИ”*] требует наличия исходящего [потока по умолчанию](#). Переход по потоку по умолчанию осуществится в случае, если не будет выполнено условие ни одного из исходящих [условных потоков](#).

При слиянии потоков управления данный логический оператор объединяет несколько альтернативных потоков в один. При этом каждый входящий поток управления направляется в единый исходящий поток.

Например, после того, как клиенту были предложены условия поставки (со скидкой или без), необходимо подготовить коммерческое предложение ([Рис. 2](#)).

Рис. 2 — Пример использования логического оператора [*Исключающее “ИЛИ”*] для слияния



При использовании логического оператора [*Исключающее “ИЛИ”*] для слияния, процесс продолжится при срабатывании любого из входящих потоков.

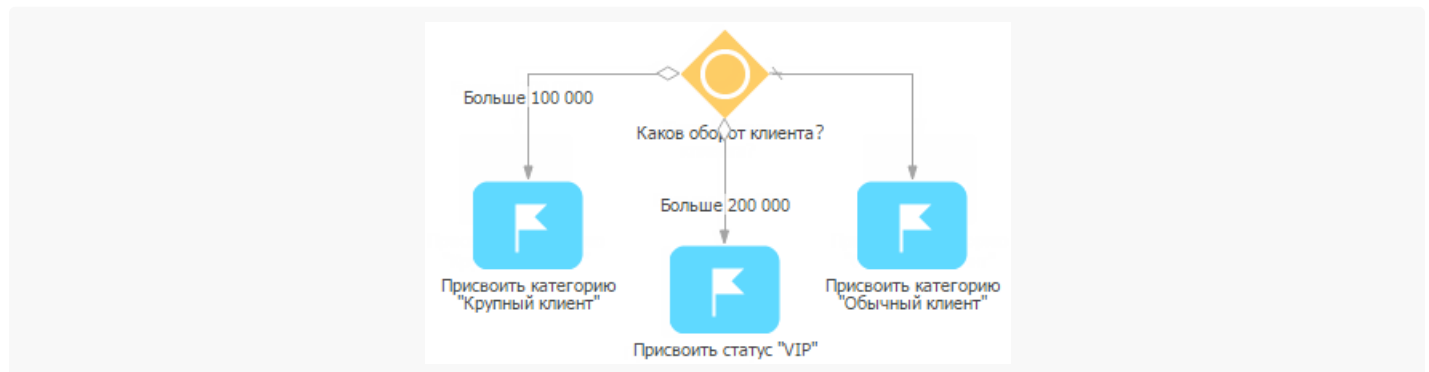
Элемент процесса [Включающее “ИЛИ”]

ПРОДУКТЫ: **ВСЕ ПРОДУКТЫ**

Вы можете использовать логический оператор [*Включающее “ИЛИ”*] для создания альтернативных потоков в процессах, которые могут выполняться параллельно. При ветвлении данный тип оператора активирует те исходящие [условные потоки](#), условия которых выполняются. При этом может быть активирован один или более исходящих потоков.

Например, если годовой оборот клиента более 100 000, его следует отнести к крупным клиентам, а если оборот больше 200 000 — клиенту присваивается статус VIP ([Рис. 1](#)).

Рис. 1 — Пример использования логического оператора [*Включающее “ИЛИ”*] для ветвления



Таким образом, клиент с оборотом 150 000 будет добавлен к крупным клиентам. Если оборот клиента будет больше 200 000 — клиент будет добавлен к крупным клиентам и параллельно получит статус VIP. Если же оборот клиента меньше или равен 100 000, сработает [поток по умолчанию](#).

При ветвлении логический оператор [*Включающее “ИЛИ”*] требует наличия исходящего потока по умолчанию. Переход по потоку по умолчанию осуществится в случае, если не будет выполнено условие ни одного из исходящих условных потоков.

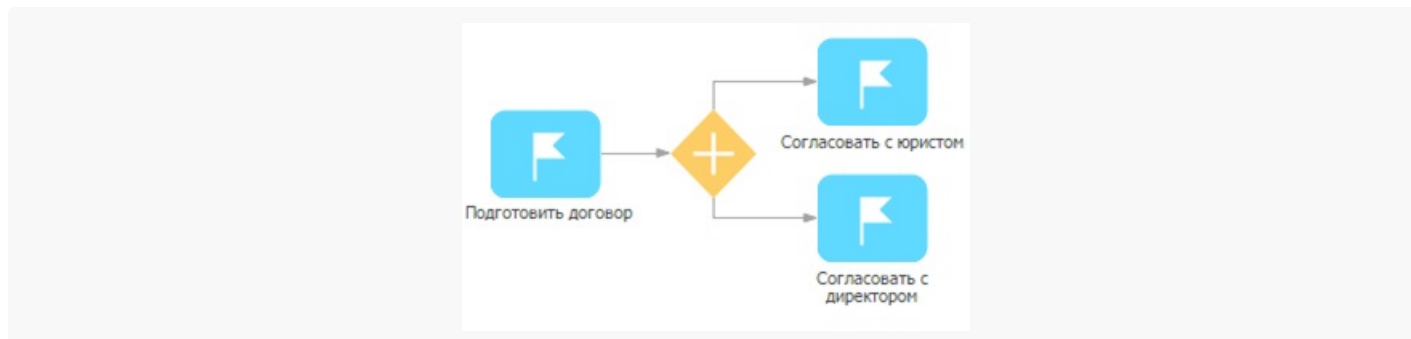
При слиянии логический оператор [*Включающее “ИЛИ”*] работает аналогично [оператору \[*Исключающее “ИЛИ”* \]](#).

Элемент процесса [Логическое “И”]

ПРОДУКТЫ: **ВСЕ ПРОДУКТЫ**

При ветвлении оператор [*Логическое “И”*] используется для создания нескольких параллельно выполняющихся ветвей процесса. Например, после подготовки текста договора необходимо согласовать его и с юристом, и с директором ([Рис. 1](#)).

Рис. 1 — Пример использования оператора [*Логическое “И”*] для ветвления



В этом случае после выполнения действия “Подготовить договор” параллельно начнется выполнение действий “Согласовать с юристом” и “Согласовать с директором”.

Вы также можете использовать оператор [Логическое “И”] для слияния параллельных потоков процесса в один, если для продолжения процесса необходимо выполнение всех параллельных потоков. Например, договор может быть подписан после согласования с юристом и директором ([Рис. 2](#)).

Рис. 2 — Пример использования оператора [Логическое “И”] для слияния



В этом случае действие “Подписать договор” начнется после выполнения действий “Согласовать с юристом” и “Согласовать с директором”. Процесс не будет продолжен, пока не будут выполнены оба эти действия.

Для слияния оператор [Логическое “И”] используется с [потоками управления](#).

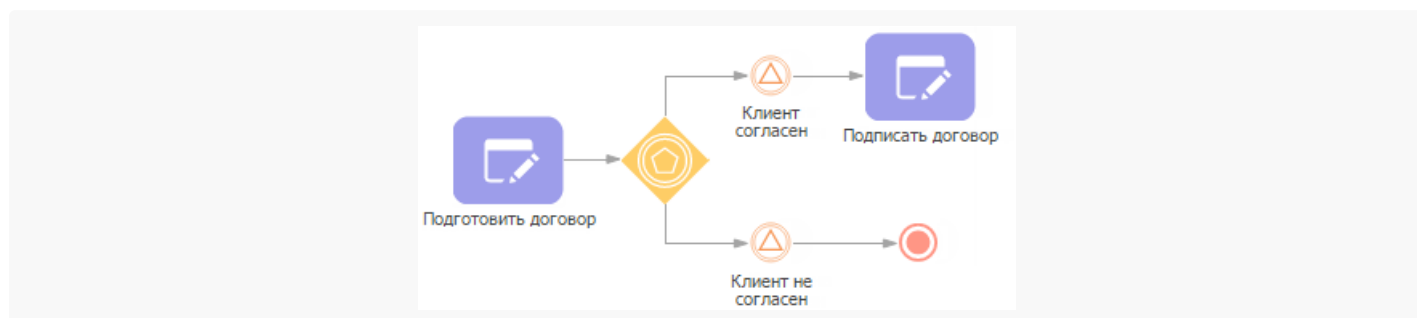
Элемент процесса [Исключающее “ИЛИ” по событиям]

ПРОДУКТЫ: [ВСЕ ПРОДУКТЫ](#)

Логический оператор [Исключающее “ИЛИ” по событиям] используется для ветвления процесса в случае, когда альтернативные пути определяются не [условными потоками](#), а промежуточными обрабатывающими событиями: получением различных [сообщений](#), [сигналов](#) либо [таймеров](#).

Это могут быть случаи, когда решение о выборе одной из альтернатив принимается вне процесса. Например, процесс подписания договора ожидает от процесса проведения переговоров сигнал о решении, которое принял клиент. При этом дальнейшие действия процесса зависят от того, каким будет это решение ([Рис. 1](#)).

Рис. 1 — Пример использования логического оператора [Исключающее “ИЛИ” по событиям]



Процесс пойдет по тому пути, для которого первым наступит соответствующее событие. В данном случае, если клиент ответил положительно, то осуществится переход к действию “Подписать договор”, если отрицательно — то процесс завершится. В любом случае произойдет обработка только одного из ожидаемых событий — того, которое наступит раньше.

Исходящими потоками логического оператора [Исключающее “ИЛИ” по событиям] являются [потоки управления](#). Логика срабатывания оператора определяется промежуточными событиями: [\[Обработка сообщения \]](#), [\[Обработка сигнала \]](#) либо [\[Обработка таймера \]](#).

При активации первого обрабатываемого события оператор перестает реагировать на остальные ожидаемые события.