

- История изменений
- Описание
- Технические параметры
- Формат сообщения
- Мониторинг
- Алгоритм работы
 - Шаг 1 - Чтение сообщений из топика
 - Шаг 2 - Валидация сообщений
 - Шаг 3 - Фильтрация сообщений
 - Шаг 3.1 - Обработка сообщения из retry-топика
 - Шаг 4 – Обработка сообщений
 - Шаг 4.a – Сохранение изменений регистров при отсутствии счета или начального решения по счету
 - Шаг 4.b – Сохранение изменений регистров и обновление счета
 - 1. Проверка закрытого ОД
 - 2. Анализ изменений регистров
 - Шаг 4.b.1 – Изменение регистра AUTH_BLOCKED при авторизационной операции (есть блок authRef)
 - Шаг 4.b.2 – Изменение регистра AUTH_BLOCKED при НЕавторизационной операции (отсутствует блок authRef)
 - Шаг 5 – Изменение offset
- Аудит
 - Шаг 1 - Сообщение о начале операции
 - Шаг 2 - Сообщение о завершении операции
 - Шаг 3 - Публикация сообщения

История изменений

Описание

Получение и обработка информации о балансах из WAY4 в КОЮЛ Остаток через GW Синхронизатор остатков (m071).

Технические параметры

Параметр	Значение
Тип взаимодействия	kafka
Протокол	TCP
Формат данных	JSON
Входящий топик	ccop.t002.m071.account_balance_verification_event_v1
Ретрай топик	ccop.t016.m041. account_balance_verification_event_v1_retry

Poll интервал ретрай топка	600 000 ms (10 минут)
Ключ	хэш от accountId
Имя БД	ccor_database
Источники запроса	Синхронизатор остатков (ccor-m071-hms-sync-acc-balance)
Потребитель запроса	КО ЮЛ Остаток (ccor-m041-hms-acc-balance)
Логирование	Да
Интеграция	intg1-r3-026
Размер пачки сообщений (max.poll.records)	500
Время ожидания пачки (max.poll.interval.ms)	300 000 ms

Формат сообщения

Параметр	Тип	Валидация	Обязательность	Описание
accountId	number	number(18)	Да	ID счета
changedAt	string	date-time формат значения YYYY-MM-DDTHH:mm:ss.SSS	Да	Дата и время изменения статуса в Way4
effectiveDate	string	date	Да	Операционный день
requestId	string	uuid	Да	Внутренний идентификатор запроса из Way4
operationId	string	number(18)	Да	Внутренний идентификатор операции из Way4
balanceUpdates[]	array		Да	Массив изменений регистров баланса
balanceUpdates[]/ {}/balanceType	string		Да	Тип регистра, уникален в рамках массива
balanceUpdates[]/ {}/amount	string	decimal	Да	Сумма изменений по регистру (со знаком)

balanceUpdates[]/ {}/newBalance	string	decimal	Да	Значение баланса после изменения (со знаком)
balanceUpdates[]/ {}/authRef{}	Object		Нет	Объект о бязательный, если изменение вызвано авторизацией. Может приходить только для balanceType: AUTH_BLOCKED
balanceUpdates[]/ {}/authRef{}/ cardIdW4	string	varchar (255)	Да	Внутренний и дентификатор карты Way4
balanceUpdates[]/ {}/authRef{}/ /authDateTime	string	date-time	Да	Дата-время в сообщении
balanceUpdates[]/ {}/authRef{}/rrn	string	varchar (255)	Да	Ссылочный номер операции
balanceUpdates[]/ {}/authRef{}/ acquirerId	string	varchar (255)	Да	Идентификатор эквайрера
balanceUpdates[]/ {}/authRef{}/a pprovalCode	string	varchar(6)	Да	Идентификатор успешной операции авторизации

```
{
  "accountId": 334080100,
  "changedAt": "2024-03-31T17:00:008.1234567+03:00",
  "effectiveDate": "2000-10-31",
  "requestId": "26691BBA-2ECC-8E34-4E06-708B40A39971",
  "operationId": 240169010420,
  "balanceUpdates": [
    {
      "balanceType": "OWN_FUNDS",
      "amount": "-90091",
      "newBalance": "50762"
    },
    {
      "balanceType": "AUTH_BLOCKED",
      "amount": "-80000",
      "newBalance": "10345",
      "authRef": {
        "cardIdW4": "1234567890125678",
        "authDateTime": "2024-03-31T17:00:008.1234567+03:00",
        "rrn": "rrn...",
        "acquirerId": "acquirerId...",
        "approvalCode": "1g3kt6"
      }
    }
  ]
}
```

```
]
}
```

Мониторинг

В рамках мониторинга формируются метрики

Метрика	Описание	Логика формирования
message_missing_account_total	Счетчик сообщений с отсутствующим ID счета	Увеличивается на 1, если получено сообщение от потока изменения балансов с отсутствующим id счета.
message_negative_hold_total	Счетчик сообщений (операций), пытающихся перевести hold в отрицательное значение	Увеличивается на 1, когда значение регистра HOLD в БД (account_balance.hold) + сумма изменения регистра (balanceUpdates /amount) <0
message_amount_hold_unmatch_total	Счетчик сообщений с одинаковыми значениями по объекту authRef и разным значением amount	Увеличивается на 1, когда от way4 и потока авторизации приходят сообщения с одинаковыми значениями объекта authRef и полем isCah=false, но разными значениями amount (по модулю)
message_repeated_records_total	Счетчик повторных сообщений матчинга	Увеличивается на 1, когда найдено несколько записей с тем же authRef

Алгоритм работы

Схема алгоритма
Поток балансов-4313

1. Сервис проводит ФЛК полученного сообщения
 - a. Проверки структуры полученной DTO, включая обязательность параметров
 - b. Проверки полученных параметров на соответствие типам и условиям валидации.
2. Если правила валидации нарушены, то сервис логирует сообщение об ошибке с текстом "Нарушены условия валидации по счету {accountId}. requestId:{requestId} operationId: {operationId}", level=ERROR или нарушена структура DTO, то сервис логирует сообщение об ошибке с текстом "Не все обязательные параметры получены в сообщении или сообщение не соответствует DTO по счету {account_id}. requestId:{requestId} operationId: {operationId}", level=ERROR и переходит к обработке следующего сообщения из пачки.
3. Если хотя бы одно сообщение прошло валидацию, переходим к **Шаг 3**, иначе алгоритм завершает работу

Шаг 3 - Фильтрация сообщений

1. Сервис определяет из какого топика пришло сообщение. Если сообщение пришло из ретрай-топика, то выполняется алгоритм из **шага 3.1**, иначе сервис переходит в **пункт 2**
2. Сервис фильтрует сообщения по проверке наличия хотя бы одного обязательного регистра:
 - a. В сообщении есть хотя бы один объект в массиве balanceUpdates, у которого balanceType один из *OWN_FUNDS*, *AUTH_BLOCKED* или *CR_LIMIT*, тогда проверка пройдена, иначе сообщение отфильтровывается и логируется с текстом "В сообщении по счету {accountId} отсутствует нужный тип баланса. Сообщение игнорируется. requestId:{requestId} operationId:{operationId}", level=WARN
3. Сервис проверяет идемпотентность.
 - a. Выполняется поиск сохраненных операций с requestId из сообщения: `account_operation.request_id == requestId` из сообщения И `account_operation.created_at >= (timestamp получения запроса в сервисе) - 12 часов`
 - i. Если не нашлось ни одной записи, удовлетворяющей условию, в `account_operation`, то считаем сообщение уникальным и переходим на **шаг 4**
 - ii. Иначе (записи найдены), считаем запрос идемпотентным и отфильтровываем сообщение с записью в логах "По счету {accountId} найдено совпадение по requestId:{requestId}, сообщение игнорируется.operationId:{operationId}", level=INFO

Шаг 3.1 - Обработка сообщения из retry-топика

1. Выполняется поиск сохраненных операций с requestId из сообщения: `account_operation.request_id == requestId` (**Модель данных БД**)
 - 1.1 Если найдена хотя бы одна запись, удовлетворяющая условию, то переходим к проверке из пункта 2.
 - 1.2. Иначе если записей не найдено, то отфильтровываем сообщение с записью в логах "Из {ИМЯ ретрай топика} по счету {accountId} не найдено записей в account_operation.requestId: {requestId} operationId:{operationId}", level=ERROR и переходим на **шаг 5**
2. Проверяем отобранные в пункте 1 записи на значение `error_flg`:
 - 2.1 Если хотя бы у одной записи (из найденных записей с одинаковым `request_id`) `error_flg = TRUE`, то для обработки таких записей переходим в **Шаг 4**

2.2. Иначе все записи с одинаковым request_id имеют error_flg = FALSE. Сообщение отфильтровываем.

Шаг 4 – Обработка сообщений

1. Сервис обрабатывает поочередно каждое сообщение из отфильтрованной на **Шаге 3** пачки.
2. Для каждого сообщения выполняется поиск счета accountId из сообщения == account_balance.account_id
 - a. Если счет НЕ найден **И** сообщение НЕ из ретрай-топика, то выполняется сохранение операций в БД в **Шаг 4.a**, логирование сообщения с текстом "В базе отсутствует счет {accountId} из сообщения. requestId:{requestId} operationId:{operationId}", level=WARN и увеличение метрики message_missing_account_total) и отправка сообщения в ретрай-топик.
Иначе (счет не найден и сообщение пришло из ретрай-топика) сервис повторно отправляет сообщение в ретрай-топик.
 - b. Если счет найден, но отсутствует первичная загрузка балансов (account_balance.initialized_flg = false) **И** сообщение НЕ из ретрай-топика, то выполняется обновление БД как в **Шаге 4.a** без увеличения метрики и логирования. Сервис отправляет сообщение в ретрай-топик.
Иначе (счет найден, но отсутствует первичная загрузка балансов И сообщение пришло из ретрай-топика) сервис повторно отправляет сообщение в ретрай-топик.
 - c. Если счет найден И первичная загрузка балансов была сделана (account_balance.initialized_flg = true), то выполняется сохранение операций в БД и обновление счета как в **Шаге 4.b**
3. После обработки всех сообщений из пачки Сервис переходит к **Шагу 5**

Связь версии счета и операций

Версия счета (account_balance.version) увеличивается на 1, если при обработке сообщения с изменением балансов по этому счету есть хотя бы одна выполненная операция (executed_flg = true). Тогда все операции, связанные с этим сообщением, будут также связаны с новой версией счета (account_operation.version = новое значение после увеличения на 1).

Иначе (нет ни одной выполненной операции), все операции будут привязываться к текущей версии счета (если он найден и это не начальное решение, в которых account_operation.version = NULL), т.к. фактически балансы счета не менялись.

Шаг 4.a – Сохранение изменений регистров при отсутствии счета или начального решения по счету

1. Сохранение изменения регистров (элемент balanceUpdates) как отдельных операций (создание записей в account_operation)
 - a. для изменения регистра с balanceType == OWN FUNDS

Создание записи в account_operationВыполняется создание записи

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= NULL
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= BASE
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= NULL
register_balance_new	= balanceUpdates/newBalance из параметров запроса
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	=requestId из параметров запроса
effective_date	= effectiveDate
executed_flg	= FALSE
error_flg	= FALSE TRUE

auth_ref_id	=NULL
auth_ref_match_id	=NULL

- b. для изменения регистра с `balanceType == AUTH_BLOCKED` и объекта "authRef" (если он есть)

Создание записи в account_operation_w4_auth_ref

Атрибут	Логика заполнения
auth_ref_id	= генерируется сервисом
account_id	= accountId из параметров запроса
card_id_w4	= balanceUpdates/authRef/cardIdW4
auth_date_time	= balanceUpdates/authRef/authDateTime
rrn	= balanceUpdates/authRef/rrn
acquirer_id	= balanceUpdates/authRef/acquirerId
approval_code	= balanceUpdates/authRef/approvalCode
request_id	= requestId из параметров запроса
operation_id	= operationId из параметров запроса

Создание записи в account_operation

Выполняется создание записи

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= NULL
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD

amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= NULL
register_balance_new	= balanceUpdates/newBalance из параметров запроса
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate
executed_flg	= FALSE
error_flg	= FALSE TRUE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id
auth_ref_match_id	= NULL

с. для изменения регистра с balanceType == CR_LIMIT

Создание записи в account_operation

Выполняется создание записи

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= NULL
operation_source	= M071

operation_direction	Если amount>0, тогда = DECREASE, иначе = INCREASE (наоборот)
register_type	= BLOCK
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance без знака из параметров запроса
register_balance_old	= NULL
register_balance_new	= balanceUpdates/newBalance без знака из параметров запроса
r_id	= NULL
span_id	= NULL
operation_id	= balanceUpdates/operationId из параметров запросов
request_id	= requestId из параметров запроса
effective_date	= effectiveDate
executed_flg	= FALSE
error_flg	= FALSE TRUE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Шаг 4.b – Сохранение изменений регистров и обновление счета

1. Проверка закрытого ОД

1. Проверяем, закрыт ли ОД

1.1 Если account_balance.last_effective_date записи \geq effectiveDate из запроса И account_balance.last_effective_date not null, то считаем, что оперендь по счету уже закрыт

1.1.1 Если сообщение пришло из retry топика, то выполняем проверку по authRef

1.1.1.1 Если в сообщении пришло изменение регистра 'AUTH_BLOCKED', и он содержит объект "authref", то Переходим в **Шаг 4.b.1** для выполнения матчинга операции

1.1.1.2 Иначе (не содержит объект "authref") , то переходим в **Шаг 5**

1.1.2 Иначе (не из retry-топика) сохраняем изменение по каждому регистру как невыполненную операцию (операция сохраняется в account_operation, но account_balance не меняется).

Создание записи в account_operation

Выполняется создание записи для каждого регистра

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE (наоборот для CR_LIMIT)
register_type	= BASE или HOLD или BLOCK в зависимости от регистра
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса (для CR_LIMIT без знака)
register_balance_old	= account_balance.base (текущее значение до операции) или = account_balance.hold (текущее значение до операции) или = account_balance.block (текущее значение до операции) или
register_balance_new	= balanceUpdates/newBalance из параметров запроса (для CR_LIMIT без знака)
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= TRUE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id (для AUTH_BLOCKED при наличии объекта authRef)
auth_ref_match_id	= NULL

Создание записи в account_operation_w4_auth_ref

Атрибут	Логика заполнения
auth_ref_id	= генерируется сервисом
account_id	= accountId из параметров запроса

card_id_w4	= balanceUpdates/authRef/cardIdW4
auth_date_time	= balanceUpdates/authRef/authDateTime
rrn	= balanceUpdates/authRef/rrn
acquirer_id	= balanceUpdates/authRef/acquirerId
approval_code	= balanceUpdates/authRef/approvalCode
request_id	= requestId из параметров запроса
operation_id	= operationId из параметров запроса

1.1.2.1 Если в сообщении пришло изменение регистра 'AUTH_BLOCKED', и он содержит объект "authref", то **Переходим в Шаг 4.b.1** для выполнения матчинга операции

1.1.2.2 Иначе (не содержит объект "authref") , то переходим в **Шаг 5**

1.2 Иначе ОД открыт (account_balance.last_effective_date записи < effectiveDate из запроса), переходим в п. 2

2. Анализ изменений регистров

2.1 для изменения регистра с balanceType == OWN_FUNDS

а. Если **сообщение получено НЕ из ретрай-топика И** (account_balance.base_changed_at < changedAt ИЛИ account_balance.base_changed_at IS NULL) (в сообщении более новое изменение по регистру), то создаем запись операции, выполняем обновление регистра, повышаем версию счета :

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= BASE
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.base (текущее значение до операции)
register_balance_new	= balanceUpdates/newBalance из параметров запроса
r_id	= NULL

span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Обновление account_balance и account_balance_hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= changedAt из сообщения
hold_changed_at	= account_balance.hold_changed_at
block_changed_at	=account_balance.block_changed_at
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	= balanceUpdates/newBalance
hold_old	= account_balance.hold
hold_new	= account_balance.hold
block_old	=account_balance.block
block_new	=account_balance.block
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
---------	-------------------

account_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	=changedAt из параметров запроса
hold_changed_at	не изменяется
block_changed_at	не изменяется
version	= version+1
base	= balanceUpdates/newBalance
hold	не изменяется
block	не изменяется
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

b. Если **сообщение получено из ретрай-топика И** (account_balance.base_changed_at < changedAt ИЛИ account_balance.base_changed_at IS NULL) (в сообщении более новое изменение по регистру), то выполняем обновление регистра, обновление записи на выполненную в account_operation и повышаем версию счета:

Обновление записи в account_operation

Атрибут	Логика заполнения
account_operation_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	= account_balance.version (новое значение)
operation_source	не изменяется
operation_direction	не изменяется
register_type	не изменяется

amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется
r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	=не изменяется
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	не изменяется
auth_ref_match_id	не изменяется

Обновление account_balance и account_balance_hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= changedAt из сообщения
hold_changed_at	= account_balance.hold_changed_at
block_changed_at	=account_balance.block_changed_at
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	= balanceUpdates/newBalance
hold_old	= account_balance.hold
hold_new	= account_balance.hold
block_old	=account_balance.block

block_new	=account_balance.block
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	=changedAt из параметров запроса
hold_changed_at	не изменяется
block_changed_at	не изменяется
version	= version+1
base	= balanceUpdates/newBalance
hold	не изменяется
block	не изменяется
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

с. Если **сообщение получено НЕ из ретрай-топика И** **account_balance.base_changed_at >= changedAt** сохраняем операцию как невыполненную (операция сохраняется в account_operation, но account_balance не меняется) и логируем с текстом "Есть сохраненная операция по счету {accountId} с более поздним changedAt по регистру {balanceType}.requestId:{requestId} operationId:{operationId}", level=WARN

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса

account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= BASE
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.base (текущее значение до операции)
register_balance_new	= balanceUpdates/newBalance из параметров запроса
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= FALSE TRUE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

d. Если сообщение получено из ретрай-топика И account_balance.base_changed_at >= changedAt переходим к следующему регистру

2.2 для изменения регистра с balanceType == CR_LIMIT

a. Если сообщение получено НЕ из ретрай-топика И (account_balance.block_changed_at < changedAt ИЛИ account_balance.block_changed_at IS NULL) (в сообщении более новое изменение по регистру), то создаем запись операции, выполняем обновление регистра, повышаем версию счета :

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071

operation_direction	Если amount>0, тогда = DECREASE, иначе = INCREASE (наоборот)
register_type	= BLOCK
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= = balanceUpdates/newBalance без знака из параметров запроса
register_balance_old	= account_balance.block (текущее значение до операции)
register_balance_new	= balanceUpdates/newBalance без знака из параметров запроса
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Обновление account_balance и account_balance_hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= account_balance.hold_changed_at
block_changed_at	= changedAt из сообщения
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold

hold_new	= account_balance.hold
block_old	=account_balance.block
block_new	= balanceUpdates/newBalance без знака из параметров запроса
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	не изменяется
block_changed_at	=changedAt из параметров запроса
version	= version+1
base	не изменяется
hold	не изменяется
block	= balanceUpdates/newBalance без знака из параметров запроса
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

b. Если **сообщение получено из ретрай-топика И** (account_balance.block_changed_at < changedAt ИЛИ account_balance.block_changed_at IS NULL) (в сообщении более новое изменение по регистру), то выполняем обновление регистра, обновление записи на выполненную в account_operation и повышаем версию счета:

Обновление записи в account_operation

Атрибут	Логика заполнения
account_operation_id	не изменяется

created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	= account_balance.version (новое значение)
operation_source	не изменяется
operation_direction	не изменяется
register_type	не изменяется
amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется
r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	= не изменяется
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	не изменяется
auth_ref_match_id	не изменяется

Обновление account_balance и account_balance_hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= account_balance.hold_changed_at
block_changed_at	= changedAt из сообщения

version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold
block_old	=account_balance.block
block_new	= balanceUpdates/newBalance без знака из параметров запроса
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	не изменяется
block_changed_at	=changedAt из параметров запроса
version	= version+1
base	не изменяется
hold	не изменяется
block	= balanceUpdates/newBalance без знака из параметров запроса
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

с. Если **сообщение получено НЕ из ретрай-топика И** account_balance.block_changed_at >= changedAt сохраняем операцию как невыполненную (**операция сохраняется в account_operation,**

но `account_balance` не меняется) и логируем с текстом "Есть сохраненная операция по счету {`accountId`} с более поздним `changedAt` по регистру {`balanceType`}.requestId:{`requestId`} operationId:{`operationId`}", level=WARN

Создание записи в `account_operation`

Атрибут	Логика заполнения
<code>account_operation_id</code>	= генерируется сервисом
<code>created_at</code>	= <code>now()</code> at time zone 'utc'
<code>updated_at</code>	= <code>now()</code> at time zone 'utc'
<code>register_changed_at</code>	= <code>changedAt</code> из параметров запроса
<code>account_id</code>	= <code>accountId</code> из параметров запроса
<code>version</code>	= <code>account_balance.version</code> (новое значение)
<code>operation_source</code>	= M071
<code>operation_direction</code>	Если <code>amount > 0</code> , тогда = DECREASE, иначе = INCREASE (наоборот)
<code>register_type</code>	= BLOCK
<code>amount</code>	= <code>balanceUpdates/amount</code> без знака из параметров запроса
<code>way4_register_balance_new</code>	= <code>balanceUpdates/newBalance</code> из параметров запроса
<code>register_balance_old</code>	= <code>account_balance.block</code> (текущее значение до операции)
<code>register_balance_new</code>	= <code>balanceUpdates/newBalance</code> из параметров запроса
<code>r_id</code>	= NULL
<code>span_id</code>	= NULL
<code>operation_id</code>	= <code>operationId</code> из параметров запроса
<code>request_id</code>	= <code>requestId</code> из параметров запроса
<code>effective_date</code>	= <code>effectiveDate</code> из параметров запроса
<code>executed_flg</code>	= FALSE
<code>error_flg</code>	= TRUE
<code>auth_ref_id</code>	= NULL
<code>auth_ref_match_id</code>	= NULL

d. Если сообщение получено из ретрай-топика И `account_balance.base_changed_at` >= `changedAt` переходим к следующему регистру

2.3 для изменения регистра с `balanceType == AUTH_BLOCKED`

- a. Если в сообщении есть тэг `balanceUpdates.authRef`, то переходим к **Шаг 4.b.1** – Изменение регистра `AUTH_BLOCKED` при авторизационной операции (есть блок `authRef`)

- б. Если в сообщении НЕТ тэга balanceUpdates.authRef, то переходим к **Шаг 4.b.2** – Изменение регистра AUTH_BLOCKED при НЕ авторизационной операции (отсутствует блок authRef)

Шаг 4.b.1 – Изменение регистра AUTH_BLOCKED при авторизационной операции (есть блок authRef)

В сообщении от синхронизатора балансов (m071) есть объект balanceUpdates.authRef.

1. Сохранение объекта в account_operation_w4_auth_ref. Если сообщение пришло НЕ из ретрай-топика, то сохраняем объект в account_operation_w4_auth_ref. Иначе сообщение пришло ИЗ ретрай-топика, переходим в пункт 2.

Создание записи в account_operation_w4_auth_ref

Атрибут	Логика заполнения
auth_ref_id	= генерируется сервисом
account_id	= accountId из параметров запроса
card_id_w4	= balanceUpdates/authRef/cardIdW4
auth_date_time	= balanceUpdates/authRef/authDateTime
rrn	= balanceUpdates/authRef/rrn
acquirer_id	= balanceUpdates/authRef/acquirerId
approval_code	= balanceUpdates/authRef/approvalCode
request_id	= requestId из параметров запроса
operation_id	= operationId из параметров запроса

2. Ищем объект в таблице account_operation_auth_ref **База данных** по полному соответствию полей:

Атрибут из запроса	Поле в account_operation_auth_ref
balanceUpdates[]/{ }/authRef{ }/cardIdW4	card_id_w4
balanceUpdates[]/{ }/authRef{ }/rrn	rrn
balanceUpdates[]/{ }/authRef{ }/acquirerId	acquirer_id

balanceUpdates[/{}/authRef{}/approvalCode	approval_code
-------------------------------------------	---------------

3. Если находится единственная запись, то считаем пришедшую операцию из way4 отбивкой по операции на изменение AUTH_BLOCKED в системе 2218 (которую провели через оркестратор

а. Сравниваем суммы операций по найденным authRef: сравниваем **по модулю** значение account_operation.amount из найденного объекта в БД и balanceUpdates/amount из объекта сообщения.

- і. Если суммы равны, то считаем данное изменение по регистру отбивкой по операции, проведенной ранее. **Выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов И закрыт или открыт ОД):**
 - **ОД открыт И сообщение НЕ из retry топика:** сохраняем полученную операцию без исполнения и выполняем матчинг с первоначальной операцией из оркестратора

Создание записи в account_operation

1. Выполняется создание записи по полученной операции

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M071
operation_direction	Если balanceUpdates/amount > 0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance

register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	Если operation_direction == INCREASE, тогда = account_balance.hold (текущее значение до операции) + account_operation.amount, Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount,
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= FALSE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id
auth_ref_match_id	= account_operation_auth_ref.auth_ref_id найденной пары

2. Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

- Иначе ((ОД открыт И операция пришла из retry-топика) ИЛИ ОД закрыт):
обновляем полученную операцию без исполнения и выполняем матчинг с первоначальной операцией из оркестратора

Обновление account_operation

Выполняется обновление записи по полученной операции

Атрибут	Логика заполнения
account_operation_id	не меняется
created_at	не меняется

updated_at	= now() at time zone 'utc'
register_changed_at	не меняется
account_id	не меняется
version	не меняется
operation_source	не меняется
operation_direction	не меняется
register_type	не меняется
amount	не меняется
way4_register_balance_new	не меняется
register_balance_old	не меняется
register_balance_new	не меняется
r_id	не меняется
span_id	не меняется
operation_id	не меняется
request_id	не меняется
effective_date	не меняется
executed_flg	= FALSE
error_flg	= FALSE
auth_ref_id	не меняется

auth_ref_match_id	= account_operation_auth_ref.auth_ref_id найденной пары
-------------------	------------------------------------------------------------

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

ii. Иначе (суммы отличаются) проверяем, закрыт ли ОД

1. Если ОД закрыт , то для записей с account_operation.
auth_ref_match_id=NULL И account_operation.operation_source='M071'
выполняем обновление и матчинг операции.

Иначе (записи уже сматчены или являются корректировкой), по ним не производим изменений.

Обновление записи в account_operation

Атрибут	Логика заполнения
account_operation_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	не изменяется
operation_source	не изменяется
operation_direction	не изменяется
register_type	не изменяется
amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется

r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	не изменяется
executed_flg	не изменяется
error_flg	не изменяется
auth_ref_id	не изменяется
auth_ref_match_id	= account_operation_auth_ref.auth_ref_id найденной пары

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

2. Иначе (**ОД открыт**), проверяем проведена ли эта операция по кэшу (account_operation_auth_ref.is_cash_flg = TRUE)
 - а. Если операция была **по кэшу** (account_operation_auth_ref.is_cash_flg = TRUE), то считаем полученное изменение из way4 отбивкой с учетом комиссии
 - і. Если корректировка по комиссии НЕ уводит значение регистра в <0 (account_balance.hold + сумма корректировки* ≥ 0) (*Сумма корректировки= разница между balanceUpdates/amount (из сообщения из way4) и account_operation.amount (первоначальная операция, учитывая operation_direction)*), то выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов):
 - Если пришла из потока балансов (НЕ из retry топика), то создаем операцию с изменением на сумму корректировки для учета комиссии, сохраняем полученную из way4 операцию как отдельную невыполненную (без изменения account_balance), обновляем в первоначальной операции auth_ref_match_id, вносим изменения в значение регистра hold в account_balance

Создание записи корректировки в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом

created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M041
operation_direction	Если разница между balanceUpdates/amount (из сообщения из way4) и account_operation.amount (с учетом operation_direction) (первоначальная операция) >0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= модуль разницы между balanceUpdates/amount (из сообщения из way4) и account_operation.amount (первоначальная операция)
way4_register_balance_new	= NULL
register_balance_old	= account_balance.hold (текущее значение до операции)

register_balance_new	Если operation_direction == INCREASE, тогда = account_balance.hold (текущее значение до операции) + account_operation.amount сумма корректировки amount из этого же объекта (таблица "Создание записи корректировки в account_operation") , Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount сумма корректировки amount из этого же объекта (таблица "Создание записи корректировки в account_operation")
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Создание записи по операции из way4 в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом

created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance
register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	Если operation_direction == INCREASE, тогда = account_balance.hold (текущее значение до операции) + account_operation.amount, Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount,
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса

request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= FALSE
auth_ref_id	= account_operation_w4_auth_ref. auth_ref_id
auth_ref_match_id	= account_operation.auth_ref_id найденной пары

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

Обновление account_balance и account_balance_hist

1. Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance. base_changed_at
hold_changed_at	= changedAt из параметров запроса
block_changed_at	=account_balance. block_changed_at
version	= account_balance.version (новое значение)

base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold (новое значение)
block_old	=account_balance.block
block_new	=account_balance.block
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

2. Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= version+ 1
base	не изменяется

hold	Если у корректирующей операции account_operation. operation_direction == INCREASE, то = account_balance.hold (текущее значение до операции) + account_operation.amount (корректирующей операции), иначе = account_balance.hold (текущее значение до операции) - account_operation.amount (корректирующей операции)
block	не изменяется
hold_change_status	не изменяется
initialized_flg	не изменяется
overdraft_flg	не изменяется
last_effective_date	не изменяется

- Иначе (операция пришла из retry-топика), то обновляем запись корректировки по комиссии в account_operation на исполненную, вносим изменения в значение регистра hold в account_balance

Обновляем запись операции корректировки для учета комиссии

Атрибут	Логика заполнения
account_operation_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	= account_balance.version (новое значение)
operation_source	не изменяется

operation_direction	не изменяется
register_type	не изменяется
amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется
r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	не изменяется
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	не изменяется
auth_ref_match_id	не изменяется

Обновление записи в account_balance и создание в account_balance_hist

1. Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется сервисом
account_id	= accountId из параметров запроса

created_at	= now() at time zone 'utc'
base_changed_at	= account_balance. base_changed_at
hold_changed_at	= changedAt из параметров запроса
block_changed_at	=account_balance. block_changed_at
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold (новое значение)
block_old	=account_balance.block
block_new	=account_balance.block
hold_change_status	=account_balance. hold_change_status
overdraft_flg	=account_balance.overdraft_flg

2. Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'

base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= version+ 1
base	не изменяется
hold	Если у корректирующей операции account_operation. operation_direction == INCREASE, то = account_balance.hold (текущее значение до операции) + account_operation.amount (корректирующей операции), иначе = account_balance.hold (текущее значение до операции) - account_operation.amount (корректирующей операции)
block	не изменяется
hold_change_status	не изменяется
initialized_flg	не изменяется
overdraft_flg	не изменяется
last_effective_date	не изменяется

- ii. Иначе, если корректировка по комиссии уводит значение регистра в <0 (account_balance.hold + сумма корректировки* < 0) выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов):
- Если пришла из потока балансов (НЕ из retry топика), то сохраняем корректировочную операцию и операцию, полученную из way4, как невыполненные (сохраняем в account_operation без изменения account_balance), обновляем auth_ref_match_id у первоначальной операции, логируем ошибку с текстом "Корректировка на сумму комиссии не применима по счету {accountId}, так как уводит регистр hold в отрицательное значение. requestId:{requestId} operationId:{operationId}", level=WARN, увеличиваем метрику **message_negative_hold_total** и отправляем сообщение в ретрай-топик. Переходим к Шаг 5

Создание записи для учета корректировочной операции

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M041
operation_direction	Если разница между balanceUpdates/amount (из сообщения из way4) и account_operation.amount (с учетом operation_direction) (первоначальная операция) >0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= модуль разницы между balanceUpdates/amount (из сообщения из way4) и account_operation.amount (первоначальная операция)
way4_register_balance_new	= NULL
register_balance_old	= account_balance.hold (текущее значение до операции)

register_balance_new	Если operation_direction == INCREASE, тогда = account_balance.hold (текущее значение до операции) + account_operation.amount сумма корректировки amount из этого же объекта (таблица "Создание записи для учета корректировочной операции"), Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount сумма корректировки amount из этого же объекта (таблица "Создание записи для учета корректировочной операции")
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= TRUE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Создание записи по операции из way4

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'

updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance
register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	Если operation_direction == INCREASE, тогда = account_balance.hold (текущее значение до операции) + account_operation.amount, Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount,
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса

effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= FALSE
auth_ref_id	= account_operation_w4_auth_ref. auth_ref_id
auth_ref_match_id	= account_operation.auth_ref_id найденной пары

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

- Иначе (сообщение пришло из ретрай-топика), то повторно направляем его в ретрай-топик. Логирование не производим, метрику не увеличиваем, обновление записи не производим.
Переходим к Шаг 5

- b. Иначе операция была **НЕ по кэшу** (account_operation_auth_ref.is_cash_flg = FALSE) выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов):
- Если пришла из потока балансов (НЕ из retry топика), то сохраняем операцию без исполнения (сохраняем в account_operation без изменения account_balance) с ошибкой и выполняем матчинг. Дополнительно увеличиваем счетчик **message_amount_hold_unmatch_total** и логируем с текстом "Операция {operationId} по счету {accountId} не по кэшу. Значения регистра hold и authBlocked не совпадают при одинаковых значениях полей блоков authRef.requestId:{requestId}", level=ERROR. Переходим к **Шаг 5**

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'

register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M071
operation_direction	Если balanceUpdates/amount > 0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance
register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	Если operation_direction == INCREASE, тогда = account_balance. hold (текущее значение до операции) + account_operation. amount, Иначе = account_balance.hold (текущее значение до операции) - account_operation.amount,
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= TRUE

auth_ref_id	= account_operation_w4_auth_ref. auth_ref_id
auth_ref_match_id	= account_operation_auth_ref. auth_ref_id найденной пары

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

- Иначе (сообщение пришло из ретрай-топика) выполняется обновление запись и выполняем матчинг. Переходим к **Шаг 5**

Обновление записи в account_operation

Атрибут	Логика заполнения
account_operation_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	не изменяется
operation_source	не изменяется
operation_direction	не изменяется
register_type	не изменяется
amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется

r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	не изменяется
executed_flg	не изменяется
error_flg	не изменяется
auth_ref_id	не изменяется
auth_ref_match_id	= account_operation.auth_ref_id найденной пары

Проставляем у первоначальной операции значение соответствующего account_operation_w4_auth_ref.auth_ref_id (account_operation.auth_ref_match_id = account_operation_w4_auth_ref.auth_ref_id) и обновляем updated_at

4. Если находится **несколько записей**, выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов):
- Если сообщение пришло из потока балансов (НЕ из retry топика) и опердень открыт, то сохраняем операцию без исполнения (сохраняем в account_operation без изменения account_balance), логируем ошибку с текстом "Матчинг не будет осуществлен по счету {accountId}, так как найдено несколько записей с одинаковым блоком authRef.requestId: {requestId} operationId:{operationId}", увеличиваем метрику **message_repeated_records_total**. Переходим к **Шаг 5**

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса

version	= account_balance.version
operation_source	= M071
operation_direction	Если balanceUpdates/amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	Если balanceUpdates/amount положительное, тогда = account_balance.hold (текущее значение до операции) + balanceUpdates/amount, Иначе = account_balance.hold (текущее значение до операции) - balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= TRUE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id
auth_ref_match_id	= NULL

б. Иначе для остальных трех кейсов (сообщение пришло (НЕ из retry топика ИЛИ из retry топика) и ОД закрыт) ИЛИ (из retry топика и ОД открыт), _логируем ошибку с текстом "Матчинг не будет осуществлен по счету {accountId}, так как найдено несколько записей с одинаковым блоком authRef.requestId:{requestId} operationId:{operationId}", увеличиваем метрику **message_repeated_records_total**. Переходим к **Шаг 5**

5. Если запись не найдена то проверяем возможность применения изменений из сообщения к регистру hold ($\text{account_balance.hold} + \text{balanceUpdates/amount} \geq 0$).

- а. Если условие выполняется, то выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов И открыт или закрыт ОД):
- і. ОД открыт и сообщение НЕ из retry топика: создаем операцию и применяем ее. Переходим к **Шаг 5**

Создание операции в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.hold
register_balance_new	= account_balance.hold + balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= balanceUpdates/operationId из параметров запроса

request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id
auth_ref_match_id	= NULL

Создание записи в account_balance_hist и обновление в account_balance

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= changed_at из сообщения
block_changed_at	=account_balance.block_changed_at
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	= account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold + balanceUpdates/amount
block_old	=account_balance.block
block_new	=account_balance.block

hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= version + 1
base	не изменяется
hold	=account_balance.hold + balanceUpdates/amount
block	не изменяется
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

ii. **ОД открыт** и сообщение из ретрай-топика: обновляем операцию и применяем ее.
Переходим к **Шаг 5**

Обновление записи в account_operation

Атрибут	Логика заполнения
account_operation_id	не изменяется
created_at	не изменяется
updated_at	= now() at time zone 'utc'
register_changed_at	не изменяется
account_id	не изменяется
version	= account_balance.version (новое значение)
operation_source	не изменяется
operation_direction	не изменяется
register_type	не изменяется
amount	не изменяется
way4_register_balance_new	не изменяется
register_balance_old	не изменяется
register_balance_new	не изменяется
r_id	не изменяется
span_id	не изменяется
operation_id	не изменяется
request_id	не изменяется
effective_date	не изменяется
executed_flg	= TRUE
error_flg	= FALSE

auth_ref_id	не изменяется
auth_ref_match_id	не изменяется

Создание записи в account_balance_hist и обновление в account_balance

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= changed_at из сообщения
block_changed_at	=account_balance.block_changed_at
version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	= account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold + balanceUpdates/amount
block_old	=account_balance.block
block_new	=account_balance.block
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= version + 1
base	не изменяется
hold	=account_balance.hold + balanceUpdates/amount
block	не изменяется
hold_change_status	не изменяется
overdraft_flg	не изменяется
initialized_flg	не изменяется
last_effective_date	не изменяется

iii. **ОД закрыт:** отфильтровываем сообщение. Переходим к **Шаг 5**

- b. Если условие НЕ выполняется ($\text{account_balance.hold} + \text{balanceUpdates/amount} < 0$), то выполняется логика в зависимости откуда пришло сообщение (retry-топик или поток изменений балансов И открыт или закрыт ОД):
- **ОД открыт** И сообщение НЕ из retry топика: создаем операцию и НЕ применяем ее, логируем ошибку с текстом "Операция не применима по счету {accountId}, так как уводит регистр hold в отрицательное значение.requestId:{requestId} operationId:{operationId}", увеличиваем метрику **message_negative_hold_total** и отправляем сообщение в ретрай-топик. Переходим к **Шаг 5**.

Создание записи в account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.hold
register_balance_new	= account_balance.hold + balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= balanceUpdates/operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE

error_flg	= TRUE
auth_ref_id	= account_operation_w4_auth_ref.auth_ref_id
auth_ref_match_id	= NULL

- **ОД открыт** И сообщение из retry топика: Сервис повторно направляет сообщение в ретрай-топик.Переходим к **Шаг 5**

- **ОД закрыт**: отфильтровываем сообщение.Переходим к **Шаг 5**

Шаг 4.b.2 – Изменение регистра AUTH_BLOCKED при НЕавторизационной операции (отсутствует блок authRef)

В сообщении от синхронизатора балансов (m071) НЕТ объекта balanceUpdates.authRef.
Выполняем алгоритм из шага **4.b.1. пункт 5** ("если запись не найдена...") и переходим в **шаг 5**.

1. ~~Смотрим на значение balanceUpdates/amount.~~

а. ~~Если balanceUpdates/amount > 0 (операция по увеличению суммы холдов), то выполняем операцию над регистром hold.~~

Создание записи в account_operation

Выполняется создание записи

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	= INCREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса

way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Обновление account_balance и account_balance_hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= changedAt из параметров запроса
block_changed_at	= account_balance.block_changed_at

version	= account_balance.version (новое значение)
base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount
block_old	=account_balance.block
block_new	=account_balance.block
hold_change_status	=account_balance.hold_change_status
overdraft_flg	=account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= текущее значение для этого счета + 1
base	не изменяется
hold	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount

block	не изменяется
hold_change_status	не изменяется
initialized_flg	не изменяется
overdraft_flg	не изменяется
last_effective_date	не изменяется

- b. ~~Иначе $\text{balanceUpdates/amount} < 0$ (операция по уменьшению суммы холдов) делаем проверку $\text{account_balance.hold} + \text{balanceUpdates/amount} \geq 0$.~~
- i. ~~Если результат применения изменения к регистру $\text{hold} \geq 0$, то меняем значение hold .~~

Обновление account_operation

Выполняется создание записи

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	= account_balance.version (новое значение)
operation_source	= M071
operation_direction	= DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса

register_balance_old	= account_balance.hold (текущее значение до операции)
register_balance_new	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= TRUE
error_flg	= FALSE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Обновление account_balance и account_balance hist

Создание записи в account_balance_hist

Атрибут	Логика заполнения
account_balance_hist_id	= генерируется Сервисом
account_id	= accountId из параметров запроса
created_at	= now() at time zone 'utc'
base_changed_at	= account_balance.base_changed_at
hold_changed_at	= changedAt из параметров запроса
block_changed_at	= account_balance.block_changed_at
version	= account_balance.version (новое значение)

base_old	=account_balance.base
base_new	=account_balance.base
hold_old	= account_balance.hold
hold_new	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount
block_old	= account_balance.block
block_new	= account_balance.block
hold_change_status	= account_balance.hold_change_status
overdraft_flg	= account_balance.overdraft_flg

Обновление записи в account_balance

Атрибут	Логика заполнения
account_id	= accountId из параметров запроса
created_at	не изменяется
updated_at	= now() at time zone 'utc'
base_changed_at	не изменяется
hold_changed_at	= changedAt из параметров запроса
block_changed_at	не изменяется
version	= текущее значение для этого счета + 1
base	не изменяется
hold	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount

block	не изменяется
hold_change_status	не изменяется
initialized_flg	не изменяется
overdraft_flg	не изменяется
last_effective_date	не изменяется

- ii. ~~Иначе (результат применения изменения к регистру hold<0) сохраняем операцию без исполнения с ошибкой. Дополнительно увеличиваем счетчик message_negative_hold_total~~

Обновление account_operation

Атрибут	Логика заполнения
account_operation_id	= генерируется сервисом
created_at	= now() at time zone 'utc'
updated_at	= now() at time zone 'utc'
register_changed_at	= changedAt из параметров запроса
account_id	= accountId из параметров запроса
version	=account.balance.version
operation_source	= M071
operation_direction	Если amount>0, тогда = INCREASE, иначе = DECREASE
register_type	= HOLD
amount	= balanceUpdates/amount без знака из параметров запроса
way4_register_balance_new	= balanceUpdates/newBalance из параметров запроса
register_balance_old	= account_balance.hold

register_balance_new	= account_balance.hold (текущее значение до операции) + balanceUpdates/amount
r_id	= NULL
span_id	= NULL
operation_id	= balanceUpdates/operationId из параметров запроса
request_id	= requestId из параметров запроса
effective_date	= effectiveDate из параметров запроса
executed_flg	= FALSE
error_flg	= TRUE
auth_ref_id	= NULL
auth_ref_match_id	= NULL

Шаг 5 – Изменение offset

- 1. Сервис выполняет commit для изменения параметра offset и алгоритм завершает работу.

Аудит

Шаг 1 - Сообщение о начале операции

При попытке валидации сообщения, полученного из топика ~~ccop.t003.m071~~
~~account_balance_event_v1~~ ccop.t002.m071.account_balance_verification_event_v1 (шаг 2 спецификации) осуществляем проверку на наличие счета в БД (по accountId из запроса == account_info.account_id в БД). Далее Сервис формирует сообщение о начале операции и подготавливает его для публикации в топик СС Аудит как событие информационной безопасности (переходит к шагу 3 блока "Отправка сообщений в СС Аудит). Для этого Сервис приводит сообщение к формату ниже:

Формат исходящего сообщения

Параметр сообщения публикуемого в Исходящем топике	Значение	Обязательность параметра в исходящем сообщении	Описание
----------------------------------------------------	----------	------------------------------------------------	----------

event_type	= "Balance"	да	Тип события
event_code	= "BALANCE_OPERATION"	да	Код типа события
business_object	= "Запрос и ответ на операцию по балансу счета"	да	Объект операции
business_operation	= "Операция по балансу счета"	да	Наименования операций аудита
initiator{ }			
initiator/sub	= "ccop-m041-ms-acc-balance"	да	Инициатор запроса
initiator/channel	= "internal"	да	Канал инициатора запроса
business_object_expr { }			
business_object_expr /access_type	= "Изменение"	да	Тип операции
business_object_expr /dataCategory	= "Банковская тайна"	да	Категория данных, к которым предоставляется доступ
business_object_expr /accountNumber	= account_info.number	да, если account_info. account_id найден	Номера счета
business_object_expr /clientId	= account_info.client_id	да, если account_info. account_id найден	Внутренний идентификатор клиента в WAY4x24
business_object_expr /infoSystemCode	= "CCOP"	да	Идентификатор ОПС из РИС
business_object_expr /infoSystemId	= "2218"	да	Идентификационный номер ОПС из РИС
business_object_expr /accountId	= accountId из запроса	да	Идентификатор счета
business_object_expr /changedAt	= changedAt из запроса	да	Дата и время изменения статуса в Way4
business_object_expr /effectiveDate	= effectiveDate из запроса	да	Операционный день

business_object_expr /requestId	= requestId из запроса	да	Уникальный идентификатор запроса
business_object_expr /balanceUpdates[]		да	Массив изменений регистров баланса
business_object_expr /balanceUpdates[]/{ } /balanceType	= balanceUpdates[]/{ } /balanceType из запроса	да	Тип регистра, уникален в рамках массива
business_object_expr /balanceUpdates[]/{ } /amount	= balanceUpdates[]/{ } /amount из запроса	да	Сумма изменений по регистру (со знаком)
business_object_expr /balanceUpdates[]/{ } /newBalance	= balanceUpdates[]/{ } /newBalance из запроса	да	Значение баланса после изменения (со знаком)
business_object_expr /balanceUpdates[]/{ } /authRef{ }		нет	Объект обязательный, если изменение вызвано авторизацией. Может приходить только для balanceType: AUTH_BLOCKED
business_object_expr /balanceUpdates[]/{ } /authRef{ }/cardIdW4	= balanceUpdates[]/{ } /authRef{ } /cardIdW4 из запроса	да	Внутренний идентификатор карты Way4
business_object_expr /balanceUpdates[]/{ } /authRef{ } /authDateTime	= balanceUpdates[]/{ } /authRef{ } /authDateTime из запроса	да	Дата-время в сообщении
business_object_expr /balanceUpdates[]/{ } /authRef{ }/rrn	= balanceUpdates[]/{ } /authRef{ }/rrn из запроса	да	Ссылочный номер операции
business_object_expr /balanceUpdates[]/{ } /authRef{ }/acquirerId	= balanceUpdates[]/{ } /authRef{ } /acquirerId из запроса	да	Идентификатор эквайера
business_object_expr /balanceUpdates[]/{ } /authRef{ } /approvalCode	= balanceUpdates[]/{ } /authRef{ } /approvalCode из запроса	да	Идентификатор успешной операции авторизации
audit_event_start{ }			
audit_event_start/title	= "Операция по балансу счета"	да	Наименование события
audit_event_start /messageTemplate	= "Начало операции"	да	Сообщения по операции

```

{
  "event_type": "Balance",
  "event_code": "BALANCE_OPERATION",
  "business_object": "Запрос и ответ на операцию по балансу счета" ,
  "business_operation": "Операция по балансу счета",
  "initiator": {
    "sub": "ccop-m041-ms-acc-balance",
    "channel": "internal"
  },
  "business_object_expr": {
    "access_type": "Изменение",
    "dataCategory": "Банковская тайна",
    "accountNumber": "1652789345",
    "clientId": "7392017638940",
    "InfoSystemCode": "CCOP",
    "InfoSystemId": "2218",
    "accountId": 334080100,
    "changedAt": "2024-03-31T17:00:008.1234567+03:00",
    "effectiveDate": "2000-10-31",
    "requestId": "26691BBA-2ECC-8E34-4E06-708B40A39971",
    "operationId": 240169010420,
    "balanceUpdates": [
      {
        "balanceType": "OWN_FUNDS",
        "amount": "-90091",
        "newBalance": "50762"
      },
      {
        "balanceType": "AUTH_BLOCKED",
        "amount": "-80000",
        "newBalance": "10345",
        "authRef": {
          "cardIdW4": "1234567890125678",
          "authDateTime": "2024-03-31T17:00:008.1234567+03:00",
          "rrn": "rrn...",
          "acquirerId": "acquirerId...",
          "approvalCode": "1g3kt6"
        }
      }
    ]
  },
  "audit_event_success": {
    "title": "Операция по балансу счета",
    "messageTemplate": "Успешное обновление баланса"
  }
}

```

Шаг 2 - Сообщение о завершении операции

1. После завершения работы Сервис формирует сообщение об окончании операции и подготавливает его для публикации в топик СС Аудит (переходит к шагу 3 блока "Отправка сообщений в СС Аудит). Для этого Сервис приводит сообщение к формату ниже:

Формат исходящего сообщения

Параметр сообщения публикуемого в Исходящем топике	Значение	Обязательность параметра в исходящем сообщении	Описание
event_type	= "Balance"	да	Тип события
event_code	= "BALANCE_OPERATION"	да	Код типа события
business_object	= "Запрос и ответ на операцию по балансу счета"	да	Объект операции
business_operation	= "Операция по балансу счета"	да	Наименования операций аудита
initiator{ }			
initiator/sub	= "ccop-m041-ms-acc-balance"	да	Инициатор запроса
initiator/channel	= "internal"	да	Канал инициатора запроса
business_object_expr{ }			
business_object_expr /access_type	= "Изменение"	да	Тип доступа
business_object_expr /dataCategory	= "Банковская тайна"	да	Категория данных, к которым предоставляется доступ
business_object_expr /accountNumber	= account_info.number	да, если account_info.account_id найден	Номер счета
business_object_expr /clientId	= account_info.client_id	да, если account_info.account_id найден	Внутренний идентификатор клиента в WAY4x24
business_object_expr /infoSystemCode	= "CCOP"	да	Идентификатор ОПС из РИС
business_object_expr /infoSystemId	= "2218"	да	Идентификационный номер ОПС из РИС

business_object_expr /accountId	= accountId из запроса	да	Идентификатор счета
business_object_expr /changedAt	= changedAt из запроса	да	Дата и время изменения статуса в Way4
business_object_expr /effectiveDate	= effectiveDate из запроса	да	Операционный день
business_object_expr /requestId	= requestId из запроса	да	Уникальный идентификатор запроса
business_object_expr /balanceUpdates[]		да	Массив изменений регистров баланса
business_object_expr /balanceUpdates[]/{ } /balanceType	= balanceUpdates[]/{ } /balanceType из запроса	да	Тип регистра, уникален в рамках массива
business_object_expr /balanceUpdates[]/{ } /amount	= balanceUpdates[]/{ } /amount из запроса	да	Сумма изменений по регистру (со знаком)
business_object_expr /balanceUpdates[]/{ } /newBalance	= balanceUpdates[]/{ } /newBalance из запроса	да	Значение баланса после изменения (со знаком)
business_object_expr /balanceUpdates[]/{ } /authRef{ }		нет	Объект обязательный, если изменение вызвано авторизацией. Может приходить только для balanceType: AUTH_BLOCKED
business_object_expr /balanceUpdates[]/{ } /authRef{ }/cardIdW4	= balanceUpdates[]/{ } /authRef{ } /cardIdW4 из запроса	да	Внутренний идентификатор карты Way4
business_object_expr /balanceUpdates[]/{ } /authRef{ } /authDateTime	= balanceUpdates[]/{ } /authRef{ } /authDateTime из запроса	да	Дата-время в сообщении
business_object_expr /balanceUpdates[]/{ } /authRef{ }/rrn	= balanceUpdates[]/{ } /authRef{ }/rrn из запроса	да	Ссылочный номер операции
business_object_expr /balanceUpdates[]/{ } /authRef{ }/acquirerId	= balanceUpdates[]/{ } /authRef{ } /acquirerId из запроса	да	Идентификатор эквайера

business_object_expr /balanceUpdates[]/{} /authRef{} /approvalCode	= balanceUpdates[]/{} /authRef{} /approvalCode из запроса	да	Идентификатор успешной операции авторизации
audit_event_success {}			
audit_event_success /title	= "Операция по балансу счета"	да, при успешном завершении работы	Наименование события
audit_event_success /messageTemplate	= "Успешное обновление баланса"	да, при успешном завершении работы	Сообщения по операции
audit_event_failure{}			
audit_event_failure /title	= "Операция по балансу счета"	да, при неуспешном завершении работы	Наименование события
audit_event_failure /messageTemplate	= "Произошла ошибка"	да, при неуспешном завершении работы	Сообщения по операции

```
{
  "event_type": "Balance",
  "event_code": "BALANCE_OPERATION",
  "business_object": "Запрос и ответ на операцию по балансу счета" ,
  "business_operation": "Операция по балансу счета",
  "initiator": {
    "sub": "ccop-m041-ms-acc-balance",
    "channel": "internal"
  },
  "business_object_expr": {
    "access_type": "Изменение",
    "dataCategory": "Банковская тайна",
    "accountNumber": "1652789345",
    "clientId": "7392017638940",
    "InfoSystemCode": "CCOP",
    "InfoSystemId": "2218",
    "accountId": 334080100,
    "changedAt": "2024-03-31T17:00:008.1234567+03:00",
    "effectiveDate": "2000-10-31",
    "requestId": "26691BBA-2ECC-8E34-4E06-708B40A39971",
    "operationId": 240169010420,
    "balanceUpdates": [
      {
        "balanceType": "OWN_FUNDS",
        "amount": "-90091",
```

```
    "newBalance": "50762"
  },
  {
    "balanceType": "AUTH_BLOCKED",
    "amount": "-80000",
    "newBalance": "10345",
    "authRef": {
      "cardIdW4": "1234567890125678",
      "authDateTime": "2024-03-31T17:00:008.1234567+03:00",
      "rrn": "rrn...",
      "acquirerId": "acquirerId...",
      "approvalCode": "1g3kt6"
    },
  },
}
]
"audit_event_success": {
  "title": "Операция по балансу счета",
  "messageTemplate": "Успешное обновление баланса"
}
}
}

{
  "event_type": "Balance",
  "event_code": "BALANCE_OPERATION",
  "business_object": "Запрос и ответ на операцию по балансу счета" ,
  "business_operation": "Операция по балансу счета",
  "initiator": {
    "sub": "ccop-m041-ms-acc-balance",
    "channel": "internal"
  },
  "business_object_expr": {
    "access_type": "Изменение",
    "dataCategory": "Банковская тайна",
    "accountNumber": "1652789345",
    "clientId": "7392017638940",
    "InfoSystemCode": "CCOP",
    "InfoSystemId": "2218",
    "accountId": 334080100,
    "changedAt": "2024-03-31T17:00:008.1234567+03:00",
    "effectiveDate": "2000-10-31",
    "requestId": "26691BBA-2ECC-8E34-4E06-708B40A39971",
    "operationId": 240169010420,
    "balanceUpdates": [
    {
      "balanceType": "OWN_FUNDS",
      "amount": "-90091",
      "newBalance": "50762"
    },
    {
      "balanceType": "AUTH_BLOCKED",
      "amount": "-80000",
```

```
    "newBalance": "10345",
    "authRef": {
      "cardIdW4": "1234567890125678",
      "authDateTime": "2024-03-31T17:00:00.1234567+03:00",
      "rrn": "rrn...",
      "acquirerId": "acquirerId...",
      "approvalCode": "1g3kt6"
    },
  }
]
"audit_event_failure": {
  "title": "Операция по балансу счета",
  "messageTemplate": "Произошла ошибка"
}
}
```

Шаг 3 - Публикация сообщения

1. Сервис публикует сообщение в Исходящем топике.

Параметр	Значение
Тип взаимодействия	kafka
Протокол	TCP
Формат данных	JSON
Топик	initial-audit
Имя БД	ccop-m041-ms-acc-balance
Источники запроса	КОЮЛ Остаток (ccop-m041-ms-acc-balance)
Вид события	Информационная безопасность
Потребитель запроса	
Интеграция	

2. Если возникает ошибка, то Сервис логирует ошибку и завершает работу.
3. Если сообщение успешно направлено в топик kafka, Сервис завершает работу.