***Сервис работы с блокировками и движениями (адаптер MIDAS)***

***Функциональная спецификация***

***Версия 4.2.0***

Лист изменений

|  |  |  |  |
| --- | --- | --- | --- |
| Версия | Дата | Автор | Описание |
| 1.0.0 | 18.11.2014 | Е. Поршнев | Первоначальная версия |
| 1.1.0 | 08.12.2014 | Е. Поршнев | Внесены изменения по замечаниям Т.Николаевой. |
| 2.0.0 | 16.12.2014 | Е. Поршнев | Отказ от использования модуля IFXBM для реализации подтвержёднных блокировок.  Уточнена логика русификации проводок и подготовки расширенной выписки (внутридневной и окончательной).  Уточнены типы данных некоторых полей в БД.  Уточнена логика работы потока MA в режиме "COB".  Внесены изменения в форматы входящих сообщений. |
| 2.1.0 | 19.12.2014 | Е. Поршнев | Для связи проджекта с движением используется пара полей TransactionNumber + ChequeNumber вместо Narrative. |
| 2.2.0 | 22.12.2014 | Е. Поршнев | Для связи проджекта с движением используется RRN проджекта.  Параметры Narrative, Transaction Type и Transaction Number доступны как входные параметры адаптера. |
| 2.3.0 | 25.12.2014 | Е. Поршнев | Внесён ряд технических уточнений.  Запрещена обработка блокировок/движений по неритейловым счетам.  Более подробно описаны правила определения баланса счёта. |
| 2.4.0 | 25.12.2014 | Е. Поршнев | Уточнена логика работы с проджектами.  Уточнена логика русификации проводок.  Исправлена обработка ситуации, когда одна и та же блокировка несколько раз авторизовывается и разавторизовывается. |
| 2.5.0 | 12.01.2015 | Е. Поршнев | Если при создании подтверждённой блокировки возникает ошибка недостаточности денег на счёте, необходимо вернуть её с отдельным кодом ошибки. |
| 3.0.0 | 25.12.2014 | Е. Поршнев | Описано создание подтвержденных блокировок на нефиксированные суммы, рассчитываемые исходя из остатка на счёте.  Описана обработка запросов на частичное подтверждение блокировок (QueryPreBlockPartialAuthorize). |
| 3.0.1 | 15.01.2015 | Е. Поршнев | Незначительные правки |
| 3.0.3 | 17.04.2015 | C. Шашков | Незначительные правки, дополнительные требования |
| 4.0.0 | 15.06.2015 | С. Шашков, А. Зимин | Незначительные правки, добавлены новые поля в файлы, добавлен новый формат сообщений для сервиса AEMovementCreate;  Описан механизм удаления сторнированных проводок |
| 4.1.0 | 17.12.2015 | C. Шашков | При определении счёта MIDAS по счёту ЦБ добавлен предварительный поиск ЦБ счёта среди субсчетов |
| 4.2.0 | 29.02.2016 | Р. Исмагилов | Доработка формата QueryMovementCreate:  1) разрешаются движения по неритейловым счетам,  2) бранч технического счёта берётся не из счёта основной полупроводки, а из головного бранча для этого счёта.  3) если технический счёт не существует, он создаётся автоматически. Для этого расширена структура таблицы IFXPPTAP, добавлено наименование счёта. |

Оглавление

[1 Общие положения 5](#_Toc422234695)

[1.1 Условные наименования, сокращения и термины 5](#_Toc422234696)

[1.2 Назначение 5](#_Toc422234697)

[1.3 Ссылки 5](#_Toc422234698)

[2 Описание программного средства 6](#_Toc422234699)

[2.1 Используемые понятия 6](#_Toc422234700)

[2.2 Жизненый цикл блокировок и движений 6](#_Toc422234701)

[2.3 Обзор доработок 7](#_Toc422234703)

[2.4 Режим работы 7](#_Toc422234704)

[3 Файлы ПС 9](#_Toc422234705)

[3.1 IFXPPLHP — PF — Запросы в MIDAS: заголовки 9](#_Toc422234706)

[3.2 IFXPPLBP — PF — Запросы в MIDAS: детали по блокировкам 9](#_Toc422234707)

[3.3 IFXPPLMP — PF — Запросы в MIDAS: детали по движениям 10](#_Toc422234708)

[3.4 IFXPPABP — PF — Блокировки (подтверждённые и неподтверждённые) 12](#_Toc422234709)

[3.5 IFXPPABL — LF — Блокировки (by BLKSRC, BLKID) 13](#_Toc422234710)

[3.6 IFXPPABM — LF — Все блокировки по счёту 14](#_Toc422234711)

[3.7 IFXPPABK1 — LF — Блокировки (by RRN) 14](#_Toc422234713)

[3.8 IFXPPABK2 — LF — Блокировки (by STNRRN) 14](#_Toc422234714)

[3.9 IFXPPAMP — PF — Движения Payment hub 14](#_Toc422234715)

[3.10 IFXPPAML — LF — Движения Payment hub (by MVMSRC, MVMID) 16](#_Toc422234716)

[3.11 IFXPPAMN — LF — Движения Payment hub (по дате, счёту) 16](#_Toc422234717)

[3.12 IFXPPAMK1 — LF — Движения Payment hub (by PRJRRN) 16](#_Toc422234719)

[3.13 IFXPPTAP — PF — Технические счета 16](#_Toc422234720)

[3.14 IFXPPJRP — PF — Лог создания проджектов по блокировкам 16](#_Toc422234721)

[4 MQ-адаптер 18](#_Toc422234722)

[4.1 Общее описание 18](#_Toc422234723)

[4.2 Алгоритм работы (обработка одного сообщения) 20](#_Toc422234724)

[4.3 Валидация сообщения в целом 24](#_Toc422234725)

[4.4 Валидация одной записи 24](#_Toc422234726)

[4.4.1 Для формата QueryAmountPreBlockCreate 24](#_Toc422234727)

[4.4.2 Для формата QueryAmountBlockCreate 25](#_Toc422234728)

[4.4.3 Для формата QueryPreBlockPartialAuthorize 26](#_Toc422234729)

[4.4.4 Для формата QueryAmountBlockCancel 26](#_Toc422234730)

[4.4.5 Для формата QueryAmountBlockReverse 26](#_Toc422234731)

[4.4.6 Для формата QueryMovementCreate 26](#_Toc422234732)

[4.4.7 Для формата AEMovementCreate 28](#_Toc422234733)

[4.5 Дальнейшая обработка записи 28](#_Toc422234735)

[4.5.1 Для формата QueryAmountPreBlockCreate 28](#_Toc422234736)

[4.5.2 Для формата QueryAmountBlockCreate 30](#_Toc422234737)

[4.5.3 Для формата QueryPreBlockPartialAuthorize 32](#_Toc422234738)

[4.5.4 Для формата QueryAmountBlockCancel 34](#_Toc422234739)

[4.5.5 Для формата QueryAmountBlockReverse 35](#_Toc422234740)

[4.5.6 Для формата QueryMovementCreate 36](#_Toc422234741)

[4.5.7 Для формата AEMovementCreate 41](#_Toc422234742)

[4.6 Технические подробности 42](#_Toc422234743)

[5 Прочие изменения 43](#_Toc422234744)

[5.1 IFXPPSTR — RPGLE — Start / stop requests processing 43](#_Toc422234745)

[5.2 IFXPPMPR — RPGLE — Make project for block 43](#_Toc422234746)

[5.3 MMBGLBLR — RPGLE — Определение баланса по счёту 46](#_Toc422234747)

[5.4 MMBGLAPR — RPGLE — Создание проводок межбранча и конверсии 47](#_Toc422234748)

[5.5 MMBGLSU0 — RPGLE — Make project and return its RRN 51](#_Toc422234749)

[5.6 IFXPPRPR — RPGLE — Recreate projects (in the morning) 53](#_Toc422234750)

[5.7 IFXSTPRR — RPGLE — Process postings 54](#_Toc422234751)

[5.8 IFXPPDDR — RPGLE — Удал. сокращ. проводки AE из GLPOSTPA COB Before 54](#_Toc422234752)

[5.9 IFXSTPNR — RPGLE — Process internal postings 56](#_Toc422234753)

[5.10 IFXST00R — RPGLE — Process postings DEFAULT (other modules) 56](#_Toc422234754)

[5.11 MT940EXTM — PF — Формат окончательной выписки для STAMT (детали операции) 56](#_Toc422234755)

[5.12 MT940SMG — RPGLE — Выгрузка окончательной выписки в STAMT 57](#_Toc422234756)

[5.13 MT942EXTM — PF — Формат внутридневной выписки для STAMT (детали операции) 57](#_Toc422234757)

[5.14 MT942SMG — RPGLE — Выгрузка внутридневной выписки в STAMT 57](#_Toc422234758)

[5.15 Изменения в настройках 58](#_Toc422234759)

[5.15.1 Настройки выгрузки DX'ов — IFXDXGSP 58](#_Toc422234760)

[5.15.2 Настройки автоматической очистки файлов — IMBCLSTP 58](#_Toc422234761)

[6 Форматы сообщений 59](#_Toc422234762)

[6.1 Входящий — QueryAmountPreBlockCreate — создание неподтверждённых блокировок 59](#_Toc422234763)

[6.2 Входящий — QueryAmountBlockCreate — создание подтверждённых блокировок 59](#_Toc422234764)

[6.3 Входящий — QueryPreBlockPartialAuthorize — частичная авторизация неподтверждённых блокировок 61](#_Toc422234765)

[6.4 Входящий — QueryAmountBlockCancel — удаление блокировок 61](#_Toc422234766)

[6.5 Входящий — QueryAmountBlockReverse — перевод подтверждённых блокировок в неподтверждённые 61](#_Toc422234767)

[6.6 Входящий — QueryMovementCreate — cоздание движений 62](#_Toc422234768)

[6.7 Входящий — AEMovementCreate — cоздание движений 63](#_Toc422234769)

[6.8 Входящий — ControlCommand — внутренняя команда 65](#_Toc422234770)

[6.9 Исходящий — StatusAmountBlock — результат выполнения запроса по работе с блокировками 65](#_Toc422234771)

[6.10 Исходящий — StatusMovement — результат выполнения запроса по работе с движениями 65](#_Toc422234772)

[6.11 Исходящий — StatusUnknownFormat — ответ на неизвестный запрос 66](#_Toc422234773)

[7 Используемые MQ-очереди 67](#_Toc422234774)

[Приложение I. Табличные данные 68](#_Toc422234796)

[I.I Глобальные статусы 68](#_Toc422234797)

[I.II Статусы обработки блокировок / движений 68](#_Toc422234798)

[I.III Соответствие форматов запросов и ответов 72](#_Toc422234799)

[I.IV Возможные значения поля IFXPPABP.ABTYPE 72](#_Toc422234800)

[I.V Возможные значения поля IFXPPABP.ABSTS 72](#_Toc422234801)

[I.VI Сообщения об ошибках при создании проджекта 72](#_Toc422234802)

# Общие положения

## Условные наименования, сокращения и термины

|  |  |
| --- | --- |
| Название термина или сокращения | Определение термина или расшифровка сокращения |
| MA | Message Adapter |
| API | Application Program Interface |
| АБС | Автоматизированная банковская система |
| MIDAS | АБС Банка |
| COB | Closure Of Business, процесс закрытия дня в MIDAS. |

## Назначение

Данный документ содержит требования на разработку адаптера MIDAS, необходимого для обеспечения работы атомарных сервисов **AMDSBlockAmountPreBlockCreate**, **AMDSBlockAmountBlockCreate**, **AMDSBlockAmountPreBlockAuthorize**, **AMDSBlockAmountPreBlockPartialAuthorize**, **AMDSBlockAmountBlockDeauthorize**, **AMDSBlockAmountRemove**, **AMDSMovementCreate**, **ADPAEPosting** проводимым согласно новым требованиям.

Документ используется в качестве руководства для разработчиков системы.

## Ссылки

Данный документ основывается на следующих исходных документах, определяющих функциональные требования к адаптеру:

* «Требования на доработку сервисного ландшафта и интеграционных подсистем для проекта “PaymentHub”». (XSRS\_PaymentHub\_Phase1\_Blocks&Movements\_v0.7.docx)
* «Блокировки сумм на счетах клиентов для проведения платежей». (SERVICES\_DG\_Z325\_Блокировки\_v.0.03.docx)
* Список вопросов и ответов (Вопросы PH\_NTV27102014.xlsx).

# Описание программного средства

## Используемые понятия

В PaymentHub при работе со счётом клиента используется три вида сущностей: неподтверждённые блокировки (в MIDAS аналога нет), подтверждённые блокировки (аналог проджекта с возможностью ежедневного пересоздания) и движения (аналог проджекта и проводки).

Неподтверждённые блокировки

Не должны влиять на доступный остаток по счёту и не должны попадать в выписку.

В MIDAS будут реализованы как записи в таблице IFXPPABP (пункт 3.4) с условием ABTYPE = 'P'.

Подтверждённые блокировки

Должны влиять на доступный остаток по счёту, должны попадать в выписку.

В MIDAS будут реализованы как записи в таблице IFXPPABP (пункт 3.4) с условием ABTYPE = 'A'. Дополнительно по таким блокировкам создаются проджекты (при необходимости).

Движения

Должны влиять на доступный остаток по счёту и отражаться в выписке.

В MIDAS будут реализованы как сочетание проджекта (для изменения остатка в режиме реального времени) и проводки (для того, чтобы баланс остался изменённым и после COB). Также добавляются записи в таблицу IFXPPAMP (пункт 3.7).

Поскольку учёт в MIDAS производиться не будет, все проводки будут корреспондироваться с техническими счетами (пункт 3.10).

Движения AE

Должны влиять на доступный остаток по счёту и отражаться в выписке.

В MIDAS будут реализованы как сочетание проджекта для ритейловых счетов (для изменения остатка в режиме реального времени) и проводки (для того, чтобы баланс остался изменённым и после COB). Также добавляются записи в таблицу IFXPPAMP (пункт 3.7). При создании движения AE при необходимости создаются дополнительные проводки для межфилиальных расчётов, а также при необходимости конверсии проводки по счетам BCE, и по счетам позиции.

## Жизненый цикл блокировок и движений

Неподтверждённые блокировки

* Блокировка может быть создана.
* Блокировка может быть удалена.
* Блокировка может быть авторизована (при этом происходит её превращение в подтверждённую).
* Блокировка может быть частично авторизована (на неполную сумму). В этом случае сумма исходной блокировки уменьшается, и дополнительно создаётся подтверждённая блокировка.

Подтверждённые блокировки

* Блокировка может быть создана с нуля.
* Блокировка может быть создана на основе неподтверждённой (в том числе в результате частичной авторизации).
* Блокировка может быть разавторизована — превращена в неподтверждённую.
* Блокировка может быть удалена.

Движения

* Движение может быть создано с нуля.
* Дебетовое движение может быть создано на основе блокировки (как подтверждённой так и неподтверждённой).

Движения AE

* Движение AE может быть создано с нуля.
* Дебетовое движение AE может быть создано на основе блокировки (как подтверждённой так и неподтверждённой).

## Обзор доработок

Основная часть нового функционала реализуется в виде обработчика MA — см. раздел 4.

Создаётся компонента по ежедневному пересозданию проджектов для подтверждённых блокировок (пункт 5.3).

Производится доработка существующего ПО в части русификации проводок (пункт 5.7), а также заполнения деталей внутридневной и окончательной выписки для STAMT (пункты 5.11–5.14). По проджектам, которые возникают в связи с блокировками и движениями, необходимо передавать в STAMT источник операции и идентификатор исходного документа.

## Режим работы

Предполагается, что поток адаптера работает всё время. Если в АБС идёт COB, он должен возвращать ошибку "В АБС идёт COB".

В COB Before добавляется компонента, отсылающая техническое MQ-сообщение адаптеру о том что COB начался, пора закрыть все файлы и перестать отвечать на запросы.

В COB Before добавляется компонента, которая удаляет сокращающиеся основные и сторнирующие их проводки АЕ.

В COB After добавляется компонента, проверяющая, что поток запущен (если по какой-либо причине поток в системе отсутствует, она его запустит), и посылающая ему сообщение, что можно начинать отвечать на запросы.

# Файлы ПС

## IFXPPLHP — PF — Запросы в MIDAS: заголовки

Файл содержит лог обработки запросов. Для каждого входящего сообщения Query\*\*\* в лог добавляется одна запись.

Запросы-дубликаты также логируются.

Очистка файла производится компонентой 9280 COB After (IMBCLEAR) на основе поля LHTS. Срок гарантированного хранения записей составляет 90 дней.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| LHID | Char(64) | Идентификатор запроса |
| LHTS | Timestamp | Дата-время запроса |
| LHRUND | Decimal(5 0) | Дата MIDAS запроса |
| LHIMSGFMT | Char(32) | Формат входящего сообщения |
| LHOMSGFMT | Char(32) | Формат ответа |
| LHRTCD | Char(1) | Код результата обработки. Список значений приведён в приложении I.I. |
| LHERRM | VarChar(80) | Сообщение об ошибке (глобальной) |

В файл должен быть встроен ключ по полям {LHID, LHTS}.

## IFXPPLBP — PF — Запросы в MIDAS: детали по блокировкам

Файл содержит лог обработки блокировок. Для каждого входящего сообщения одного из форматов QueryAmountPreBlockCreate, QueryAmountBlockCreate, QueryPreBlockPartialAuthorize, QueryAmountBlockCancel, QueryAmountBlockReverse в лог добавляется по одной записи на каждую блокировку.

Запросы-дубликаты в этом файле не логируются.

Очистка файла производится компонентой 9280 COBAfter (IMBCLEAR) на основе поля LBTS. Срок гарантированного хранения записей составляет 90 дней.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| LBID | Char(64) | Идентификатор запроса |
| LBTS | Timestamp | Дата-время запроса |
| LBROWNUM | Binary(9 0) | Номер блокировки в списке |
| LBBLKSRC | Char(32) | Код системы-инициатора блокировки |
| LBBLKID | Char(32) | Идентификатор блокировки во внешней системе |
| LBPRESRC | Char(32) | Код системы-инициатора авторизуемой блокировки |
| LBPREID | Char(32) | Идентификатор авторизуемой блокировки во внешней системе |
| LBOBJREF | Char(32) | Референс объекта системы-инициатора |
| LBCBAC | Char(20) | Счёт в формате ЦБ |
| LBAMTTYPE | Char(10) | Тип суммы блокировки |
| LBAMT | Char(23) | Сумма блокировки в валюте счёта |
| LBPERCENT | Char(8) | Процент от суммы остатка |
| LBSTRD | Char(8) | Дата начала действия блокировки |
| LBENDD | Char(8) | Дата конца действия блокировки |
| LBTRYP | Char(2) | Transaction Type для проджекта |
| LBTNMR | Char(6) | Transaction Number для проджекта |
| LBNRTD | Char(30) | Комментарий к проджекту |
| LBUSEOVD | Char(1) | Учитывать лимит овердрафта |
| LBIGNBLK | Char(1) | Флаг обхода блокировки all/credit/debit |
| LBIGNBAL | Char(1) | Флаг игнорирования доступного остатка |
| LBNARR | VarChar(300) | Комментарий к блокировке |
| LBPARSRC | Char(32) | Родительская блокировка |
| LBPARID | Char(32) |
| LBABSREF | Char(32) | Референс созданной блокировки в АБС |
| LBBLKAMT | Char(23) | Величина заблокированной суммы |
| LBTYPE | Char(1) | Статус блокировки |
| LBRTCD | Char(2) | Код результата обработки. Список значений приведён в приложении I.II. |
| LBERRM | VarChar(80) | Сообщение об ошибке |

В файл должен быть встроен ключ по полям {LBID, LBROWNUM}.

## IFXPPLMP — PF — Запросы в MIDAS: детали по движениям

Файл содержит лог обработки движений. Для каждого входящего сообщения формата QueryMovementCreate, AEMovementCreate в лог добавляется по одной записи на каждое движение.

Запросы-дубликаты в этом файле не логируются.

Очистка файла производится компонентой 9280 COB After (IMBCLEAR) на основе поля LMTS. Срок гарантированного хранения записей составляет 90 дней.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| LMID | Char(64) | Идентификатор запроса |
| LMTS | Timestamp | Дата-время запроса |
| LMROWNUM | Binary(9 0) | Номер движения в списке |
| LMMVMSRC | Char(32) | Код системы-инициатора движения |
| LMMVMID | Char(32) | Идентификатор движения во внешней системе |
| LMBLKSRC | Char(32) | Идентификатор блокировки |
| LMBLKID | Char(32) |
| LMOBJREF | Char(32) | Референс объекта системы-инициатора |
| LMADDREF | Char(32) | Дополнительный идентификатор (комиссии и т.п.) |
| LMCBAC | Char(20) | Счёт в формате ЦБ |
| LMAMT | Char(23) | Сумма движения в валюте счёта |
| LMDRCR | Char(1) | Признак дебет/кредит |
| LMSPOS | Char(7) | Источник проводок |
| LMOTRF | Char(15) | Идентификатор проводки |
| LMDPMT | Char(3) | Департамент |
| LMPRFC | Char(4) | Центр прибыли |
| LMBOKC | Char(2) | Book Code |
| LMTRAT | Char(5) | Тип транзакции |
| LMPNAR | Char(30) | Английский комментарий к проводке |
| LMTRYP | Char(2) | TransactionType для проджекта |
| LMTNMR | Char(6) | TransactionNumber для проджекта |
| LMNRTD | Char(30) | Комментарий к проджекту |
| LMOED | Char(8) | Дата и время поступления требования к счёту |
| LMVALD | Char(8) | Дата валютирования |
| LMRNAR | VarChar(300) | Русский комментарий к проводке |
| LMSRNAR | VarChar(100) | Короткий русский комментарий к проводке |
| LMUSEOVD | Char(1) | Учитывать лимит овердрафта |
| LMIGNBLK | Char(1) | Флаг обхода блокировки all/credit/debit |
| LMIGNBAL | Char(1) | Флаг игнорирования доступного остатка |
| LMSTORNO | Char(1) | Движение является СТОРНО |
| LMPRIMEID | Char(6) | Идентификатор блокировки средств на карте в системе Prime |
| LMABSREF | Char(32) | Референс созданного движения в АБС |
| LMFAN | Char(1) | Признак необходимости создания веерной проводки (0/1) |
| LMRTCD | Char(2) | Код результата обработки. Список значений приведён в приложении I.II. |
| LMERRM | VarChar(80) | Сообщение об ошибке |

В файл должен быть встроен ключ по полям {LMID, LMROWNUM}.

## IFXPPABP — PF — Блокировки (подтверждённые и неподтверждённые)

Файл содержит блокировки.

Очистка файла производится компонентой 9280 COB After (IMBCLEAR) на основе поля ABDLTD. Срок гарантированного хранения записей составляет 90 дней. Поскольку поле ABDLTD заполняется только у удалённых блокировок, активные блокировки удалены при очистке быть не могут.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| ABID | Char(12) | Внутренний индентификатор блокировки |
| ABSTS | Char(1) | Статус блокировки (активна или удалена) |
| ABTYPE | Char(1) | Тип блокировки (подтвержена или нет) |
| ABLHID | Char(64) | Идентификатор запроса |
| ABBLKSRC | Char(32) | Код системы-инициатора блокировки |
| ABBLKID | Char(32) | Идентификатор блокировки во внешней системе |
| ABPRESRC | Char(32) | Код системы-инициатора авторизуемой блокировки |
| ABPREID | Char(32) | Идентификатор авторизуемой блокировки во внешней системе |
| ABOBJREF | Char(32) | Референс объекта системы-инициатора |
| ABCBAC | Char(20) | Счёт в формате ЦБ |
| ABBRCA | Char(3) | Счёт в формате MIDAS |
| ABCNUM | Numeric(6 0) |
| ABCCY | Char(3) |
| ABACOD | Numeric(4 0) |
| ABACSQ | Numeric(2 0) |
| ABAMTTYPE | Char(10) | Тип суммы блокировки |
| ABEXCLIM | Decimal(13 0) | Сумма, которая должна остаться на счёте |
| ABPERCENT | Decimal(7 4) | Процент суммы остатка |
| ABAMT | Decimal(13 0) | Сумма блокировки в валюте счёта |
| ABSTRD | Decimal(5 0) | Дата начала действия блокировки |
| ABENDD | Decimal(5 0) | Дата конца действия блокировки |
| ABVALTYP | Char(10) | Тип валютирования (\*CURDATE / \*ENDDATE) |
| ABTRYP | Char(2) | Transaction Type для проджекта |
| ABTNMR | Char(6) | Transaction Number для проджекта |
| ABNRTD | Char(30) | Комментарий к проджекту |
| ABUSEOVD | Char(1) | Учитывать лимит овердрафта |
| ABIGNBLK | Char(1) | Флаг обхода блокировки all/credit/debit |
| ABIGNBAL | Char(1) | Флаг игнорирования доступного остатка |
| ABPARSRC | Char(32) | Родительская блокировка |
| ABPARID | Char(32) |
| ABNARR | VarChar(300) | Комментарий к блокировке |
| ABLPD | Decimal(5 0) | Дата создания проджекта |
| ABDLTD | Decimal(5 0) | Дата удаления блокировки |
| ABRRN | Binary(9 0) | RRN проджекта в RSACMTPD |
| ABSTNRRN | Binary(9 0) | RRN сторнирующего проджекта в RSACMTPD |

В файл должен быть встроен ключ по полю ABID.

## IFXPPABL — LF — Блокировки (by BLKSRC, BLKID)

Логический файл на файл IFXPPABP с ключом по полям {ABBLKSRC, ABBLKID} и отбором записей по условию ABSTS = 'A'.

## IFXPPABM — LF — Все блокировки по счёту

Логический файл на файл IFXPPABP с ключом по полям {ABCNUM, ABCCY, ABACOD, ABACSQ, ABBRCA, ABTYPE}.

## IFXPPABK1 — LF — Блокировки (by RRN)

Логический файл на файл IFXPPABP с ключом по полям {ABRRN, ABLPD desc} и отбором записей по условию ABRRN ≠ 0.

## IFXPPABK2 — LF — Блокировки (by STNRRN)

Логический файл на файл IFXPPABP с ключом по полям {ABSTNRRN, ABLPD desc} и отбором записей по условию ABSTNRRN ≠ 0.

## IFXPPAMP — PF — Движения Payment hub

Файл содержит движения.

Очистка файла производится компонентой 9280 COB After (IMBCLEAR) на основе поля AMPSTD. Срок гарантированного хранения записей составляет 90 дней.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| AMID | Char(12) | Внутренний индентификатор движения |
| AMLHID | Char(64) | Идентификатор запроса |
| AMMVMSRC | Char(32) | Код системы-инициатора движения |
| AMMVMID | Char(32) | Идентификатор движения во внешней системе |
| AMBLKSRC | Char(32) | Идентификатор блокировки |
| AMBLKID | Char(32) |
| AMOBJREF | Char(32) | Референс объекта системы-инициатора |
| AMADDREF | Char(32) | Дополнительный идентификатор (комиссии и т.п.) |
| AMCBAC | Char(20) | Счёт в формате ЦБ |
| AMBRCA | Char(3) | Счёт в формате MIDAS |
| AMCNUM | Numeric(6 0) |
| AMCCY | Char(3) |
| AMACOD | Numeric(4 0) |
| AMACSQ | Numeric(2 0) |
| AMAMT | Decimal(13 0) | Сумма движения в валюте счёта |
| AMPSTD | Decimal(5 0) | Дата проводки |
| AMVALD | Decimal(5 0) | Дата валютирования |
| AMDRCR | Decimal(1 0) | Признак дебет/кредит |
| AMSPOS | Char(7) | Источник проводок |
| AMOTRF | Char(15) | Идентификатор проводки |
| AMDPMT | Char(3) | Департамент |
| AMPRFC | Char(4) | Центр прибыли |
| AMBOKC | Char(2) | Book Code |
| AMTRAT | Char(5) | Тип транзакции |
| AMPNAR | Char(30) | Английский комментарий к проводке |
| AMTRYP | Char(2) | Transaction Type для проджекта |
| AMTNMR | Char(6) | Transaction Number для проджекта |
| AMNRTD | Char(30) | Комментарий к проджекту |
| AMOED | Char(8) | Дата и время поступления требования к счёту |
| AMRNAR | VarChar(300) | Русский комментарий к проводке |
| AMSRNAR | VarChar(100) | Короткий русский комментарий к проводке |
| AMUSEOVD | Char(1) | Учитывать лимит овердрафта |
| AMIGNBLK | Char(1) | Флаг обхода блокировки all/credit/debit |
| AMIGNBAL | Char(1) | Флаг игнорирования доступного остатка |
| AMSTORNO | Char(1) | Движение является СТОРНО |
| AMPRIMEID | Char(6) | Идентификатор блокировки средств на карте в системе Prime |
| AMFAN | Char(1) | Признак необходимости создавать веерную проводку (0/1) |
| AMPRJRRN | Binary(9 0) | RRN проджекта в RSACMTPD |
| AMPSTRRN | Binary(9 0) | RRN проводки в GLPOSTPA |

В файл должен быть встроен ключ по полю AMID.

## IFXPPAML — LF — Движения Payment hub (by MVMSRC, MVMID)

Логический файл на файл IFXPPAMP с ключом по полям {AMMVMSRC, AMMVMID}.

## IFXPPAMN — LF — Движения Payment hub (по дате, счёту)

Логический файл на файл IFXPPAMP с ключом по полям {AMPSTD, AMCNUM, AMCCY, AMACOD, AMACSQ, AMBRCA, AMAMT, AMOTRF, AMDRCR}.

## IFXPPAMK1 — LF — Движения Payment hub (by PRJRRN)

Логический файл на файл IFXPPAMP с ключом по полю AMPRJRRN и отбором записей по условию AMPRJRRN ≠ 0.

## IFXPPTAP — PF — Технические счета

Файл содержит список технических счетов, используемых для корреспонденции с клиентскими. Счета открываются в разрезе валюты и филиала.

Опция ведения файла не предусмотрена. Настройки в файл добавляются при установке ПС.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| TABRCA | Char(3) | Отделение счёта |
| TAACOD | Numeric (4, 0) | Account Code |
| TAACSQ | Numeric (2, 0) | Account Sequence |
| TAACNM | Char(20) | Short Acc.Name |

Структура файла пока не окончательная, так как нет решения бухгалтерии о том, какие именно счета будут использоваться.

В файл должен быть встроен уникальный ключ по полю TABRCA.

Текущая версия заполнения данной таблицы:

|  |  |  |  |
| --- | --- | --- | --- |
| Отделение счёта | Account Code | Account Sequence | Acc.Name |
| 001 | 1314 | 01 | AO UniCredit Bank |
| 002 | 1314 | 01 | UCB St.Petersburg |
| 009 | 1314 | 01 | UCB Perm |
| 015 | 1314 | 01 | UCB Rostov-on-Don |
| 016 | 1314 | 01 | UCB Chelyabinsk |
| 023 | 1314 | 01 | UCB Krasnodar |
| 030 | 1314 | 01 | UCB Voronezh |
| 033 | 1314 | 01 | UCB Samara |
| 040 | 1314 | 01 | UCB Ekaterinburg |
| 045 | 1314 | 01 | UCB Nizhny Novgorod |
| 050 | 1314 | 01 | UCB Novosibirsk |
| 066 | 1314 | 01 | UCB Ufa |
| 093 | 1314 | 01 | UCB Volgograd |
| 101 | 1314 | 01 | UCB Stavropol |

## IFXPPJRP — PF — Лог создания проджектов по блокировкам

В лог записываются ошибки утреннего пересоздания проджектов по подтверждённым блокировкам.

Очистка файла производится компонентой 9280 COB After (IMBCLEAR) на основе поля JRDATE. Срок гарантированного хранения записей составляет 90 дней.

Структура файла

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип | Описание |
| JRABID | Char(12) | Внутренний индентификатор блокировки |
| JRBLKSRC | Char(32) | Код системы-инициатора блокировки |
| JRBLKID | Char(32) | Идентификатор блокировки во внешней системе |
| JROBJREF | Char(32) | Референс объекта системы-инициатора |
| JRBRCA | Char(3) | Счёт в формате MIDAS |
| JRCNUM | Numeric(6 0) |
| JRCCY | Char(3) |
| JRACOD | Numeric(4 0) |
| JRACSQ | Numeric(2 0) |
| JRDATE | Decimal(5 0) | Операционная дата |
| JRTS | TimeStamp | Время |
| JRRTCD | Char(1) | Код ошибки |
| JRERRM | Varchar(80) | Текст ошибки |

# MQ-адаптер

## Общее описание

Обработка сообщений осуществляется программой IFXPPMH2R в 75-м потоке Message Adapter и в 76-м потоке Message Adapter (только запросы AEMovementCreate)

Поток обслуживает следующие сервисы:

|  |  |  |
| --- | --- | --- |
| Сервис | Формат сообщения | Описание |
| SCASABlockAmountPreBlockCreate | QueryAmountPreBlockCreate | Создание неподтверждённых блокировок |
| SCASABlockAmountBlockCreate | QueryAmountBlockCreate | Создание подтверждённых блокировок |
| SCASABlockAmountPreBlockAuthorize | QueryAmountBlockCreate | Создание подтверждённых блокировок на основе неподтверждённых (авторизация) |
| SCASABlockAmountPreBlockPartialAuthorize | QueryPreBlockPartialAuthorize | Частичное подтверждение (авторизация) блокировок по списку |
| SCASABlockAmountBlockDeauthorize | QueryAmountBlockReverse | Разавторизация подтверждённых блокировок (превращение в неподтверждённые) |
| SCASABlockAmountRemove | QueryAmountBlockCancel | Удаление блокировок (подтверждённых или неподтверждённых) |
| SCASAMovementCreate | QueryMovementCreate | Создание движений |
| AdpAEPosting | AEMovementCreate | Создание движений AE |

Адаптер имеет два режима работы:

* "день" — нормальная работа.
* "COB" — на все запросы возвращается ответ "В АБС идёт COB". Валидация и логирование запросов не производятся.

Переключение в режим "COB" производится на время закрытия дня в MIDAS.

Адаптер контролирует уникальность поступающих запросов. Повторная валидация/обработка запроса-дубликата не производится.

* Если дублирующий запрос приходит по успешно обработанной операции, то адаптер отвечает точно так же, как по изначальной операции, но указывает Status = '1' (вместо '0').
* Если дублирующий запрос приходит по операции, обработанной ранее с ошибкой, то адаптер отвечает точно так же, как по изначальной операции, не изменяя Status.

Каждый из форматов сообщений допускает передачу нескольких[[1]](#footnote-1) записей (блокировок или движений) в одном запросе. В этом случае запрошенное действие должно быть произведено в одной транзакции: либо для всех переданных блокировок (движений), либо ни для одной.

Если на каком-либо из этапов обработки сообщения обнаружена ошибка по одной из блокировок (движений), следует продолжать работу и пытаться обработать остальные блокировки (движения); после этого откатить всю транзакцию. Таким образом при наличии нескольких ошибочных блокировок (движений) в одном сообщений должны быть найдены все ошибки (по возможности). Обработка каждой конкретной блокировки (движения) производится до первой ошибки.

Поддерживается частичное подтвеждение блокировки. В этом случае создаётся новая подтверждённая блокировка, а сумма неподтверждённой блокировки уменьшается на сумму частичного подтверждения.

В случае, если производится авторизация блокировки (ввод подтверждённой блокировки на основе неподтверждённой), либо ввод движения на основе блокировки, проверяется совпадение атрибутов у новой сущности (блокировки или движения) и у исходной (блокировки): они должны быть либо одновременно не заполнены, либо одновременно заполнены и иметь одинаковые значения. Если это условие не выполнено, возвращается ошибка. Список контролируемых атрибутов:

* Авторизация блокировки: ObjectReference, Account, Amount, StartDate, EndDate, UseOverdraft, Comment, ParentID.
* Частичное подтверждение блокировки: ObjectReference, Account, StartDate, EndDate, UseOverdraft, Comment, ParentID. Дополнительно проверяется, что сумма частичного подтверждения не превышает сумму неподтверждённой блокировки.
* Ввод [дебетового] движения на основе неподтверждённой блокировки: ObjectReference, Account, Amount, UseOverdraft, Comment.
* Ввод [дебетового] движения на основе подтверждённой блокировки: ObjectReference, Account, Amount, UseOverdraft, IgnoreBlockFlag, IgnoreBalance, Comment.

При получении сообщения добавляется запись в лог-файл IFXPPLHP. После того, как сообщение разобрано на группы Block (Movement), для каждой такой группы добавляется по записи в файл IFXPPLBP (IFXPPLMP для Movement).

Для запросов-дубликатов добавляется запись в файл IFXPPLHP, но не добавляются записи в файл IFXPPLBP (IFXPPLMP).

В случае глобальной ошибки (неверный MessageFormat, либо недопустимая длина сообщения) в ответном сообщении заполняются только поля MessageFormat, Status и Message; группы Block (Movement) не добавляются. То же самое происходит при работе в режиме "COB". В остальных случаях группа Block (Movement) добавляется в ответное сообщение в стольких же экземплярах, в скольких она присутствовала в запросе.

Если обработка запроса выполнена успешно (без ошибок), поле Status ответа заполняется константой «0». Если возникли ошибки при валидации/обработке отдельных блокировок (движений), поле Status заполняется константой «2», при этом код и текст каждой из обнаруженных ошибок записываются в поля ErrorCode и ErrorText соответствующей по номеру группы Block (Movement). Если возникла глобальная ошибка, поле Status заполняется константой «3», текст ошибки указывается в поле Message. Если поток находится в режиме "COB", поле Status заполняется константой «4».

Работа с блокировками и движениями по неритейловым счетам не производится. Такие события должны отсекаться на уровне сервиса.

## Алгоритм работы (обработка одного сообщения)

Произвести верхнеуровневую валидацию сообщения (пункт 4.3).

Если это сообщение формата ControlCommand и Action = \*START, перейти в режим "день" и завершить работу.

Если это сообщение формата ControlCommand и Action = \*STOP, перейти в режим "COB" и завершить работу.

Если поток находится в режиме "COB", но COB уже окончен (проверяется по Data Area COB02STAT), поток переходит в режим "день" автоматически.

Если поток находится в режиме "COB", отправить ответ с кодом '4' и сообщением "В АБС идёт COB". После этого завершить работу.

Добавить строку в лог-файл IFXPPLHP:

IFXPPLHP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| LHID | Идентификатор запроса | RequestID |
| LHTS | Дата-время запроса | %TimeStamp() |
| LHRUND | Дата MIDAS запроса | Текущая операционная дата |
| LHIMSGFMT | Формат входящего сообщения | MessageFormat |
| LHOMSGFMT | Формат ответа | Определяется по таблице в приложении I.III |
| LHRTCD | Код результата обработки | Пока не заполняется |
| LHERRM | Сообщение об ошибке (глобальной) | Пока не заполняется |

Если на этапе валидации обнаружена глобальная ошибка, дозаполнить поля лог-файла LHRTCD (константой '3') и LHERRM (сообщением об ошибке), отправить ответ в формате LHOMSGFMT и завершить работу.

Проверить по таблице IFXPPLHP, не приходил ли запрос с таким же идентификатором (RequestID) ранее.

Если приходил и был обработан успешно, дозаполнить поля лог-файла LHRTCD (константой '1') и LHERRM (сообщением о дублировании) и отослать тот же ответ[[2]](#footnote-2), что и на исходный запрос, с изменением поля Status на '1' (Дублирование запроса).

Если приходил и был обработан с ошибкой, скопировать поля LHRTCD и LHERRM из исходного запроса и отослать тот же ответ, что и на исходный запрос (не изменяя Status).

Начать новую транзакцию. Запись в логи производится вне контроля целостности.

Разобрать тело сообщения на записи (блокировки или движения). Записи обрабатываются последовательно в порядке следования в запросе. Для каждой записи:

Добавить строку в лог-файл IFXPPLBP (для блокировок) или IFXPPLMP (для движений):

IFXPPLBP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| LBID | Идентификатор запроса | IFXPPLHP.LHID |
| LBTS | Дата-время запроса | IFXPPLHP.LHTS |
| LBROWNUM | Номер блокировки в списке | Порядковый номер записи, считая с 1 |
| LBBLKSRC | Идентификатор блокировки | Block.ID.SystemCode |
| LBBLKID | Block.ID.Reference |
| LBPRESRC | Идентификатор авторизуемой блокировки | Block.PreBlockID.SystemCode |
| LBPREID | Block.PreBlockID.Reference |
| LBOBJREF | Референс объекта системы-инициатора | Block.ObjectReference |
| LBCBAC | Счёт в формате ЦБ | Block.Account |
| LBAMTTYPE | Тип суммы блокировки | Block.AmountType |
| LBAMT | Сумма блокировки в валюте счёта | Block.Amount |
| LBPERCENT | Процент от суммы остатка | Block.Percent |
| LBSTRD | Дата начала действия блокировки | Block.StartDate |
| LBENDD | Дата конца действия блокировки | Block.EndDate |
| LBTRYP | Transaction Type для проджекта | Block.ProjectTrnTyp |
| LBTNMR | Transaction Number для проджекта | Block.ProjectTrnNbr |
| LBNRTD | Комментарий к проджекту | Block.ProjectNarrative |
| LBUSEOVD | Учитывать лимит овердрафта | Block.UseOverdraft |
| LBIGNBLK | Флаг обхода блокировки all/credit/debit | Block.IgnoreBlockFlag |
| LBIGNBAL | Флаг игнорирования доступного остатка | Block.IgnoreBalance |
| LBNARR | Комментарий к блокировке | Block.Comment |
| LBPARSRC | Родительская блокировка | Block.ParentID.SystemCode |
| LBPARID | Block.ParentID.Reference |
| LBABSREF | Референс созданной блокировки в АБС | Пока не заполняется |
| LBBLKAMT | Величина заблокированной суммы | Пока не заполняется |
| LBTYPE | Статус блокировки | Пока не заполняется |
| LBRTCD | Код результата обработки | Пока не заполняется |
| LBERRM | Сообщение об ошибке | Пока не заполняется |

Не во всех форматах присутствуют все перечисленные поля. В случае, если поля в сообщении нет, в файле оно также не заполняется.

IFXPPLMP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| LMID | Идентификатор запроса | IFXPPLHP.LHID |
| LMTS | Дата-время запроса | IFXPPLHP.LHTS |
| LMROWNUM | Номер движения в списке | Порядковый номер записи, считая с 1 |
| LMMVMSRC | Идентификатор движения | Movement.ID.SystemCode |
| LMMVMID | Movement.ID.Reference |
| LMBLKSRC | Идентификатор блокировки | Movement.BlockID.SystemCode |
| LMBLKID | Movement.BlockID.Reference |
| LMOBJREF | Референс объекта системы-инициатора | Movement.ObjectReference |
| LMCBAC | Счёт в формате ЦБ | Movement.Account |
| LMAMT | Сумма движения в валюте счёта | Movement.Amount |
| LMDRCR | Признак дебет/кредит | Movement.DebitCredit |
| LMSPOS | Источник проводок | Movement.SPOS |
| LMOTRF | Идентификатор проводки | Movement.OTRF |
| LMDPMT | Департамент | Movement.Department |
| LMPRFC | Центр прибыли | Movement.ProfitCenter |
| LMBOKC | Book Code | Movement.BookCode |
| LMTRAT | Тип транзакции | Movement.PostingTrnTyp |
| LMPNAR | Английский комментарий к проводке | Movement.PostingNarrative |
| LMTRYP | TransactionType для проджекта | Movement.ProjectTrnTyp |
| LMTNMR | TransactionNumber для проджекта | Movement.ProjectTrnNbr |
| LMNRTD | Комментарий к проджекту | Movement.ProjectNarrative |
| LMOED | Дата и время поступления требования к счёту | Movement.EntryDate |
| LMVALD | Дата валютирования | Movement.ValueDate |
| LMRNAR | Русский комментарий к проводке | Movement.Comment |
| LMSRNAR | Короткий русский комментарий к проводке | Movement.ShortComment |
| LMUSEOVD | Учитывать лимит овердрафта | Movement.UseOverdraft |
| LMIGNBLK | Флаг обхода блокировки all/credit/debit | Movement.IgnoreBlockFlag |
| LMIGNBAL | Флаг игнорирования доступного остатка | Movement.IgnoreBalance |
| LMSTORNO | Движение является СТОРНО | Movement.Storno |
| LMPRIMEID | Идентификатор блокировки средств на карте в системе Prime | Movement.PrimeBlockID |
| LMABSREF | Референс созданного движения в АБС | Пока не заполняется |
| LMRTCD | Код результата обработки | Пока не заполняется |
| LMERRM | Сообщение об ошибке | Пока не заполняется |

Выполнить валидацию записи (пункт 4.4).

Если на этапе валидации не возникло ошибок, произвести дальнейшую обработку (пункт 4.5).

Дозаполнить оставшиеся поля:

|  |  |  |  |
| --- | --- | --- | --- |
| Сущность | Успешность обработки | Поле | Значение |
| Неподтверждённая блокировка | Ошибок не было | LBABSREF | IFXPPABP.ABID |
| LBBLKAMT | IFXPPABP.ABAMT |
| LBTYPE | IFXPPABP.ABTYPE |
| LBRTCD | '00' |
| Ошибки были | LBRTCD | Код ошибки |
| LBERRM | Текст ошибки |
| Подтверждённая блокировка | Ошибок не было | LBABSREF | IFXPPABP.ABID |
| LBBLKAMT | IFXPPABP.ABAMT |
| LBTYPE | IFXPPABP.ABTYPE |
| LBRTCD | '00' |
| Ошибки были | LBRTCD | Код ошибки |
| LBERRM | Текст ошибки |
| Движение | Ошибок не было | LMABSREF | IFXPPAMP.AMID |
| LMRTCD | '00' |
| Ошибки были | LMRTCD | Код ошибки |
| LMERRM | Текст ошибки |

Если по всем записям запроса обработка прошла без ошибок:

Подтвердить транзакцию (COMMIT).

Дозаполнить поле LHRTCD (константой '0'), сформировать и отослать ответ в формате LHOMSGFMT.

Если хотя бы по одной записи запроса была ошибка:

Отменить транзакцию (ROLLBACK).

Дозаполнить поле LHRTCD (константой '2'), сформировать и отослать ответ в формате LHOMSGFMT.

## Валидация сообщения в целом

Проверить формат сообщения. Поле MessageFormat должно содержать одну из констант QueryAmountPreBlockCreate, QueryAmountBlockCreate, QueryPreBlockPartialAuthorize, QueryAmountBlockCancel, QueryAmountBlockReverse, QueryMovementCreate, ControlCommand, AEMovementCreate.В противном случае зафиксировать глобальную ошибку.

Проверить длину сообщения.

Для формата ControlCommand она должна быть равна 42. В противном случае проигнорировать сообщение.

Для остальных форматов она должна иметь вид size(Header) + k\*size(Record), где k ≥ 1 — целое число; иначе зафиксировать глобальную ошибку.

Если формат сообщения равен ControlCommand, проверить, что поле Action имеет значение \*START, либо \*STOP. В противном случае проигнорировать сообщение.

## Валидация одной записи

### Для формата QueryAmountPreBlockCreate

Уникальный идентификатор блокировки (поля Block.ID.SystemCode и Block.ID.Reference) должен быть заполнен; иначе зафиксировать ошибку с кодом «01».

В системе не должно быть активной блокировки (подтверждённой или неподтверждённой) с таким же значением Block.ID[[3]](#footnote-3); иначе зафиксировать ошибку с кодом «02».

Референс объекта внешней системы (поле Block.ObjectReference) должен быть заполнен; иначе зафиксировать ошибку с кодом «04».

Счёт ЦБ (поле Block.Account) должен быть заполнен и привязан[[4]](#footnote-4) к открытому[[5]](#footnote-5) счёту или субсчёту MIDAS; иначе зафиксировать ошибку с кодом «05», «06» или «07».

Счёт должен быть ритейловый[[6]](#footnote-6); иначе зафиксировать ошибку с кодом «21».

Сумма блокировки (поле Block.Amount) должна содержать положительное число, записанное в стандартной десятичной нотации. Количество значащих (ненулевых) цифр после десятичной точки не должно превышать максимально допустимое для валюты счёта[[7]](#footnote-7). В пересчёте на минорные единицы валюты («копейки») сумма не должна превосходить . В противном случае зафиксировать ошибку с кодом «08», «09» или «10».

Даты начала и окончания действия блокировки (Block.StartDate и Block.EndDate) могут содержать корректные даты[[8]](#footnote-8), либо быть не заполнены. В противном случае зафиксировать ошибку с кодом «11» или «12». Если не заполнена дата начала, используется текущая операционная дата. Если не заполнена дата окончания, используется \*HIVAL (без ограничения срока).

Дата окончания действия блокировки должна быть больше или равна дате начала действия блокировки. В противном случае зафиксировать ошибку с кодом «13».

Дата начала действия блокировки должна быть меньше либо равна текущей операционной дате, либо не не заполнена. В противном случае зафиксировать ошибку с кодом «38».

Поле Block.UseOverdraft может быть незаполнено, либо должно содержать одну из констант «0» либо «1». В противном случае зафиксировать ошибку с кодом «14».

Идентификатор родительской блокировки (группа Block.ParentID) может быть либо целиком не заполнен, либо должны быть заполнены оба поля Block.ParentID.SystemCode и Block.ParentID.Reference. Иначе зафиксировать ошибку с кодом «18».

### Для формата QueryAmountBlockCreate

Выполняются все те же проверки, что и для QueryAmountPreBlockCreate, кроме 4.4.1.2. Также смотри пункт 4.4.2.5 по поводу проверки 4.4.1.6.

(вместо 4.4.1.2) В системе не должно быть активной подтверждённой блокировки с таким же значением Block.ID; иначе зафиксировать ошибку с кодом «02».

Группа Block.PreBlockID может быть не заполнена, либо должна совпадать с Block.ID; иначе зафиксировать ошибку с кодом «34».

Поле Block.AmountType должно быть заполнено и содержать одну из констант: «\*AMOUNT», «\*EXCESS» или «\*PERCENT»; иначе зафиксировать ошибку с кодом «31».

Проверка 4.4.1.6 выполняется только если тип суммы блокировки (поле Block.AmountType) равен «\*AMOUNT» или «\*EXCESS».

Если поле Block.AmountType равен «\*PERCENT», то поле Block.Percent должно содержать положительное число, не превосходящее 100. Количество знаков после десятичной точки не должно превышать 4. В противном случае зафиксировать ошибку с кодом «32».

Поле Block.UseOverdraft должно быть заполнено; иначе зафиксировать ошибку с кодом «14».

Поле Block.IgnoreBlockFlag может быть не заполнено, либо должно содержать одну из констант «0» либо «1»; иначе зафиксировать ошибку с кодом «15». Если оно не заполнено, использовать значение по умолчанию «0».

Поле Block.IgnoreBalance может быть не заполнено, либо должно содержать одну из констант «0» либо «1»; иначе зафиксировать ошибку с кодом «16». Если оно не заполнено, использовать значение по умолчанию «0».

Поле Block.ProjectTrnTyp должно быть заполнено; иначе зафиксировать ошибку с кодом «17».

Поле Block.ProjectNarrative должно быть заполнено; иначе зафиксировать ошибку с кодом «19».

### Для формата QueryPreBlockPartialAuthorize

Выполняются все те же проверки, что и для QueryAmountBlockCreate.

Группа Block.PreBlockID должна быть заполнена и не должна совпадать с Block.ID. В противном случае вернуть ошибку с кодом «33» или «34».

### Для формата QueryAmountBlockCancel

Уникальный идентификатор блокировки (поля Block.ID.SystemCode и Block.ID.Reference) должен быть заполнен; иначе зафиксировать ошибку с кодом «01».

Блокировка с указанным идентификатором (подтверждённая или неподтверждённая) должна присутствовать в системе и быть активна. Иначе зафиксировать ошибку с кодом «03».

### Для формата QueryAmountBlockReverse

Уникальный идентификатор блокировки (поля Block.ID.SystemCode и Block.ID.Reference) должен быть заполнен; иначе зафиксировать ошибку с кодом «01».

Подтверждённая блокировка с указанным идентификатором должна присутствовать в системе и быть активна. Иначе зафиксировать ошибку с кодом «03».

### Для формата QueryMovementCreate

Уникальный идентификатор движения (поля Movement.ID.SystemCode и Movement.ID.Reference) должен быть заполнен; иначе зафиксировать ошибку с кодом «51».

В системе не должно быть движения с таким же значением Movement.ID [[9]](#footnote-9); иначе зафиксировать ошибку с кодом «53».

Идентификатор блокировки (группа Movement.BlockID) может быть либо целиком не заполнен, либо должны быть заполнены оба поля Movement.BlockID.SystemCode и Movement.BlockID.Reference. Иначе зафиксировать ошибку с кодом «52».

Референс объекта внешней системы (поле Movement.ObjectReference) должен быть заполнен; иначе зафиксировать ошибку с кодом «54».

Счёт ЦБ (поле Movement.Account) должен быть заполнен и привязан к открытому счёту или субсчёту MIDAS; иначе зафиксировать ошибку с кодом «55», «56» или «57».

Сумма движения (поле Movement.Amount) должна содержать положительное число, записанное в стандартной десятичной нотации. Количество значащих (ненулевых) цифр после десятичной точки не должно превышать максимально допустимое для валюты счёта. В пересчёте на минорные единицы валюты («копейки») сумма не должна превосходить . В противном случае зафиксировать ошибку с кодом «58», «59» или «60».

Признак дебета/кредита (поле Movement.DebitCredit) должно содержать одну из констант «D» или «C». В противном случае зафиксировать ошибку с кодом «61».

Если движение кредитовое (Movement.DebitCredit = 'C'), группа Movement.BlockID не может быть заполнена. В противном случае зафиксировать ошибку с кодом «62».

Источник движения (поле Movement.SPOS) должен быть заполнен. В противном случае зафиксировать ошибку с кодом «63».

Референс проводки (поле Movement.OTRF) должен быть заполнен. В противном случае зафиксировать ошибку с кодом «64».

Департамент (поле Movement.Department) должен быть заполнен и содержать допустимый[[10]](#footnote-11) код департамента. В противном случае зафиксировать ошибку с кодом «65».

Код центра прибыли (поле Movement.ProfitCenter) может быть не заполнен, либо должен содержать допустимый[[11]](#footnote-12) центр прибыли. В противном случае зафиксировать ошибку с кодом «66».

Поле Movement.BookCode может быть не заполнено, либо должно содержать допустимый[[12]](#footnote-13) код. В противном случае зафиксировать ошибку с кодом «67».

Тип транзакции (Movement.PostingTrnTyp) должен быть заполнен допустимым[[13]](#footnote-14) значением, либо может быть не заполнен. Если он не заполнен, будет использоваться значение по умолчанию: 92000. В случае указания недопустимого значения зафиксировать ошибку с кодом «68».

Английский комментарий к проводке (Movement.PostingNarrative) должен быть заполнен. Иначе зафиксировать ошибку с кодом «69».

Тип проджекта (Movement.ProjectTrnTyp) должен быть заполнен. Иначе зафиксировать ошибку с кодом «70».

Если описание проджекта (Movement.ProjectNarrative) не заполнено, используется значение из поля Movement.PostingNarrative.

Дата поступления требования к счёту (Movement.EntryDate) может быть не заполнена, либо должна содержать корректную дату. В противном случаезафиксировать ошибку с кодом «71».

Дата валютирования (Movement.ValueDate) может быть не заполнена (в этом случае используется текущая операционная дата), либо должна содержать корректную дату. Дата валютирования не должна находиться слишком далеко в прошлом либо в будущем (конкретные ограничения задаются настройками системы[[14]](#footnote-15)). В противном случае зафиксировать ошибку с кодом «72».

Комментарий к проводке (Movement.Comment) должен быть заполнен. Иначе зафиксировать ошибку с кодом «73».

Короткий комментарий к проводке (Movement.ShortComment) должен быть заполнен. Иначе зафиксировать ошибку с кодом «74».

Если движение дебетовое, поле Movement.UseOverdraft должно содержать одно из значений «0», «1»;иначе зафиксировать ошибку с кодом «75». Для кредитовых движений проверка не производится.

Поле Movement.IgnoreBlockFlag должно содержать одно из значений «0», «1», либо быть не заполнено (в этом случае используется значение по умолчанию «0»). Иначе зафиксировать ошибку с кодом «76».

Поле Movement.IgnoreBalance должно содержать одно из значений «0», «1», либо быть не заполнено (в этом случае используется значение по умолчанию «0»). Иначе зафиксировать ошибку с кодом «77».

Поле Movement.Storno должно содержать одно из значений «0», «1», либо быть не заполнено (в этом случае используется значение по умолчанию «0»). Иначе зафиксировать ошибку с кодом «78».

### Для формата AEMovementCreate

Проверить, что в запросе ровно два движения: одно по дебету, другое по кредиту;

Произвести те же проверки, что и для формата QueryMovementCreate со следующими отличиями: разрешаются проводки по неритейловым счетам. В этом случае их баланс не контролируется;

## Дальнейшая обработка записи

### Для формата QueryAmountPreBlockCreate

Добавить запись в таблицу IFXPPABP:

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| ABID | Внутренний индентификатор блокировки | GenNewID('AB') |
| ABSTS | Статус блокировки (активна или удалена) | 'A' |
| ABTYPE | Тип блокировки (подтвержена или нет) | 'P' |
| ABBLKSRC | Идентификатор блокировки | Block.ID.SystemCode |
| ABBLKID | Block.ID.Reference |
| ABPRESRC | Идентификатор авторизуемой блокировки | Не заполняется |
| ABPREID | Не заполняется |
| ABOBJREF | Референс объекта системы-инициатора | Block.ObjectReference |
| ABCBAC | Счёт в формате ЦБ | Block.Account |
| ABBRCA | Счёт в формате MIDAS | Определяется по Block.Account (сначала по таблице субсчетов IMBCBHCP, затем по таблице IMBCBADP) |
| ABCNUM |
| ABCCY |
| ABACOD |
| ABACSQ |
| ABAMTTYPE | Тип суммы блокировки | Block.AmountType |
| ABEXCLIM | Сумма, которая должна остаться на счёте | 0 |
| ABPERCENT | Процент суммы остатка | 0 |
| ABAMT | Сумма блокировки в валюте счёта | Определяется по Block.Amount (с учётом числа десятичных знаков валюты ABCCY) |
| ABSTRD | Дата начала действия блокировки | Если поле Block.StartDate заполнено, то Block.StartDate; в противном случае — текущая рабочая дата. Дата указывается в формате MIDAS. |
| ABENDD | Дата конца действия блокировки | Если поле Block.EndDate заполнено, то Block.EndDate; в противном случае — 2245-10-14[[15]](#footnote-16). Дата указывается в формате MIDAS. |
| ABVALTYP | Тип валютирования (\*CURDATE / \*ENDDATE) | Не заполняется |
| ABTRYP | Transaction Type для проджекта | Не заполняется |
| ABTNMR | Transaction Number для проджекта | Не заполняется |
| ABNRTD | Комментарий к проджекту | Не заполняется |
| ABUSEOVD | Учитывать лимит овердрафта | Block.UseOverdraft |
| ABIGNBLK | Флаг обхода блокировки all/credit/debit | Не заполняется |
| ABIGNBAL | Флаг игнорирования доступного остатка | Не заполняется |
| ABPARSRC | Родительская блокировка | Block.ParentID.SystemCode |
| ABPARID | Block.ParentID.Reference |
| ABNARR | Комментарий | Block.Comment |
| ABLPD | Дата создания проджекта | 0 |
| ABDLTD | Дата удаления | = ABENDD |
| ABRRN | RRN проджекта в RSACMTPD | 0 |
| ABSTNRRN | RRN сторнирующего проджекта в RSACMTPD | 0 |

### Для формата QueryAmountBlockCreate

Проверить, существует ли неподтверждённая блокировка (запись в таблице IFXPPABP) с идентификатором Block.PreBlockID.

Если неподтверждённая блокировка существует, убедиться в совпадении её полей ObjectReference, Account, Amount, StartDate, EndDate, UseOverdraft, Comment, ParentID с соответствующими полями запроса. Каждое из перечисленных полей либо должно быть в обоих случаях незаполнено, либо заполнено одним и тем же значением.  
Если обнаружились расхождения, зафиксировать ошибку с кодом «20».

Если неподтверждённая блокировка существует, считать запись в память и внести следующие изменения (обновление записи в БД будет произведена позже — в пункте 4.5.2.8):

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABPRESRC | Идентификатор авторизуемой блокировки | = ABBLKSRC |
| ABPREID | = ABBLKID |
| ABTYPE | Тип блокировки (подтвержена или нет) | 'A' |
| ABVALTYP | Тип валютирования (\*CURDATE / \*ENDDATE) | '\*CURDATE' |
| ABTRYP | Transaction Type для проджекта | Block.ProjectTrnTyp |
| ABTNMR | Transaction Number для проджекта | Block.ProjectTrnNbr |
| ABNRTD | Комментарий к проджекту | Block.ProjectNarrative |
| ABIGNBLK | Флаг обхода блокировки all/credit/debit | Block.IgnoreBlockFlag |
| ABIGNBAL | Флаг игнорирования доступного остатка | Block.IgnoreBalance |

Если неподтверждённой блокировки нет, заполнить поля таблицы IFXPPABP (запись в БД будет произведена позже — в пункте 4.5.2.8):

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| ABID | Внутренний индентификатор блокировки | GenNewID('AB') |
| ABSTS | Статус блокировки (активна или удалена) | 'A' |
| ABTYPE | Тип блокировки (подтвержена или нет) | 'A' |
| ABBLKSRC | Идентификатор блокировки | Block.PreBlockID.SystemCode |
| ABBLKID | Block.PreBlockID.Reference |
| ABPRESRC | Идентификатор авторизуемой блокировки | Не заполняется |
| ABPREID | Не заполняется |
| ABOBJREF | Референс объекта системы-инициатора | Block.ObjectReference |
| ABCBAC | Счёт в формате ЦБ | Block.Account |
| ABBRCA | Счёт в формате MIDAS | Определяется по Block.Account (сначала по таблице субсчетов IMBCBHCP, затем по таблице IMBCBADP) |
| ABCNUM |
| ABCCY |
| ABACOD |
| ABACSQ |
| ABAMTTYPE | Тип суммы блокировки | Block.AmountType |
| ABEXCLIM | Сумма, которая должна остаться на счёте | Если Block.AmountType = '\*EXCESS', то определяется по Block.Amount (с учётом числа десятичных знаков валюты ABCCY).  Если Block.AmountType ≠ '\*EXCESS', то 0. |
| ABPERCENT | Процент суммы остатка | Если Block.AmountType = '\*PERCENT', то Block.Percent.  Если Block.AmountType ≠ '\*PERCENT', то 0. |
| ABAMT | Сумма блокировки в валюте счёта | Если Block.AmountType = '\*AMOUNT', то определяется по Block.Amount (с учётом числа десятичных знаков валюты ABCCY).  Если Block.AmountType ≠ '\*AMOUNT', то 0. |
| ABSTRD | Дата начала действия блокировки | Если поле Block.StartDate заполнено, то Block.StartDate; в противном случае — текущая рабочая дата. Дата указывается в формате MIDAS. |
| ABENDD | Дата конца действия блокировки | Если поле Block.EndDate заполнено, то Block.EndDate; в противном случае — 2245-10-14. Дата указывается в формате MIDAS. |
| ABVALTYP | Тип валютирования (\*CURDATE / \*ENDDATE) | '\*CURDATE' |
| ABTRYP | Transaction Type для проджекта | Block.ProjectTrnTyp |
| ABTNMR | Transaction Number для проджекта | Block.ProjectTrnNbr |
| ABNRTD | Комментарий к проджекту | Block.ProjectNarrative |
| ABUSEOVD | Учитывать лимит овердрафта | Block.UseOverdraft |
| ABIGNBLK | Флаг обхода блокировки all/credit/debit | Block.IgnoreBlockFlag |
| ABIGNBAL | Флаг игнорирования доступного остатка | Block.IgnoreBalance |
| ABPARSRC | Родительская блокировка | Block.ParentID.SystemCode |
| ABPARID | Block.ParentID.Reference |
| ABNARR | Комментарий | Block.Comment |
| ABLPD | Дата создания проджекта | 0 |
| ABDLTD | Дата удаления | = ABENDD |
| ABRRN | RRN проджекта в RSACMTPD | Пока не заполнять |
| ABSTNRRN | RRN сторнирующего проджекта в RSACMTPD | 0 |

Создать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*INPUT' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) равен '1', зафиксировать ошибку с кодом «22»; если код возврата равен '2', зафиксировать ошибку с кодом «25» и сообщением p@ErrM.

Обновить подготовленную запись файла IFXPPABP из параметра p@Block (программа IFXPPMPR могла изменить ряд полей).

Обновить/добавить запись в файле IFXPPABP.

### Для формата QueryPreBlockPartialAuthorize

Проверить, что существует активная неподтверждённая блокировка (запись в таблице IFXPPABP) с идентификатором Block.PreBlockID. В противном случае зафиксировать ошибку с кодом «35».

Убедиться в совпадении её полей ObjectReference, Account, StartDate, EndDate, UseOverdraft, Comment, ParentID с соответствующими полями запроса. Каждое из перечисленных полей либо должно быть в обоих случаях незаполнено, либо заполнено одним и тем же значением.  
Если обнаружились расхождения, зафиксировать ошибку с кодом «20».

Вычислить сумму частичного подтверждения.

Если Block.AmountType = '\*AMOUNT', то {сумма частичного подтверждения} = Block.Amount.

Если Block.AmountType = '\*EXCESS' или '\*PERCENT', необходимо определить доступный остаток по счёту с помощью программы MMBGLBLR.

Параметры запуска MMBGLBLR

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p1Mode | Режим работы | Если Block.UseOverdraft = '1', то '\*WITH\_OVD'.  Если Block.UseOverdraft = '0', то '\*NO\_OVD'. |
| p1MIAC | Счёт MIDAS (в формате ClientCcyAcodSqBrc) | Не заполняется |
| p1CBAC | Счёт ЦБ | Block.Account |
| p1Balance | Сумма |  |
| p1AP | Сторона баланса счёта (активный/пассивный) |  |
| p1RtCd | Код возврата |  |
| p1ErrM | Сообщение об ошибке |  |

Если работа программы MMBGLBLR завершилась с ошибкой (p1RtCd = '1'), зафиксировать ошибку с кодом «26».

Если счёт оказался активным (p1AP = '\*ACTIVE'), зафиксировать ошибку с кодом «36».

Если Block.AmountType = '\*EXCESS', то {сумма частичного подтверждения} = p1Balance – Block.Amount.

Если Block.AmountType = '\*PERCENT', то {сумма частичного подтверждения} = p1Balance \* Block.Percent / 100. Округление производится по математическим правилам.

Рассчитанная сумма должна быть положительной. В противном случае зафиксировать ошибку с кодом «22».

Сравнить полученную сумму частичного подтверждения с полем ABAMT неподтверждённой блокировки. Если она оказалась больше, зафиксировать ошибку с кодом «37».

Уменьшить сумму неподтверждённой блокировки на сумму частичного подтверждения (обновить запись в файле IFXPPABP). Если после этого сумма неподтверждённой блокировки стала равна нулю, удалить блокировку.

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABAMT | Сумма блокировки в валюте счёта | = ABAMT – {сумма частичного подтверждения} |
| ABDLTD | Дата удаления | Если ABAMT = 0, то текущая операционная дата.  Иначе не изменяется. |
| ABSTS | Статус блокировки (активна или удалена) | Если ABAMT = 0, то 'D'.  Иначе не изменяется. |

Заполнить поля таблицы IFXPPABP (запись в БД будет произведена позже — в пункте 4.5.3.11):

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABBLKSRC | Идентификатор авторизуемой блокировки | Block.ID.SystemCode |
| ABBLKID | Block.ID.Reference |
| ABPRESRC | Идентификатор авторизуемой блокировки | Block.PreBlockID.SystemCode |
| ABPREID | Block.PreBlockID.Reference |
| ABTYPE | Тип блокировки (подтвержена или нет) | 'A' |
| ABAMTTYPE | Тип суммы блокировки | '\*AMOUNT' |
| ABAMT | Сумма блокировки в валюте счёта | {сумма частичного подтверждения} (см. пункт 4.5.3.3) |
| ABVALTYP | Тип валютирования (\*CURDATE / \*ENDDATE) | '\*CURDATE' |
| ABTRYP | Transaction Type для проджекта | Block.ProjectTrnTyp |
| ABTNMR | Transaction Number для проджекта | Block.ProjectTrnNbr |
| ABNRTD | Комментарий к проджекту | Block.ProjectNarrative |
| ABIGNBLK | Флаг обхода блокировки all/credit/debit | Block.IgnoreBlockFlag |
| ABIGNBAL | Флаг игнорирования доступного остатка | Block.IgnoreBalance |

Остальные поля копируются из неподтверждённой блокировки.

Создать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*INPUT' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) равен '1', зафиксировать ошибку с кодом «22»; если код возврата равен '2', зафиксировать ошибку с кодом «25» и сообщением p@ErrM.

Обновить подготовленную запись файла IFXPPABP из параметра p@Block (программа IFXPPMPR могла изменить ряд полей).

Добавить запись в файл IFXPPABP.

### Для формата QueryAmountBlockCancel

Считать запись по блокировке в память и внести следующие изменения (обновление записи в БД будет произведена позже — в пункте 4.5.4.5):

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABSTS | Статус блокировки (активна или удалена) | 'D' |
| ABDLTD | Дата удаления | Текущая операционная дата |

Если блокировка подтверждённая (ABTYPE = 'A'), сторнировать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*STORNO' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) не равен '0', зафиксировать ошибку с кодом «25» и сообщением p@ErrM.

Обновить подготовленную запись файла IFXPPABP из параметра p@Block (программа IFXPPMPR могла изменить ряд полей). Для случая неподтверждённой блокировки этот пункт не требуется.

Обновить запись в файле IFXPPABP.

### Для формата QueryAmountBlockReverse

Сторнировать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*STORNO' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) не равен '0', зафиксировать ошибку с кодом «25» и сообщением p@ErrM.

Обновить подготовленную запись файла IFXPPABP из параметра p@Block (программа IFXPPMPR могла изменить ряд полей) и дополнительно обновить следующие поля:

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABPRESRC | Идентификатор авторизуемой блокировки | \*Blanks |
| ABPREID | \*Blanks |
| ABTYPE | Тип блокировки (подтвержена или нет) | 'P' |

Обновить запись по блокировке в таблице IFXPPABP.

### Для формата QueryMovementCreate

Заполнить поля таблицы IFXPPAMP (запись будет произведена позже — в пункте 4.5.6.24):

IFXPPAMP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| AMID | Внутренний индентификатор движения | GenNewID('AM') |
| AMMVMSRC | Код системы-инициатора движения | Movement.ID.SystemCode |
| AMMVMID | Идентификатор движения во внешней системе | Movement.ID.Reference |
| AMBLKSRC | Идентификатор блокировки | Movement.BlockID.SystemCode |
| AMBLKID | Movement.BlockID.Reference |
| AMOBJREF | Референс объекта системы-инициатора | Movement.ObjectReference |
| AMADDREF | Дополнительный референс | Movement.AdditionalReference |
| AMCBAC | Счёт в формате ЦБ | Movement.Account |
| AMBRCA | Счёт в формате MIDAS | Определяется по Movement.Account (сначала по таблице субсчетов IMBCBHCP, затем по таблице IMBCBADP). |
| AMCNUM |
| AMCCY |
| AMACOD |
| AMACSQ |
| AMAMT | Сумма движения в валюте счёта | Определяется по Movement.Amount (с учётом числа десятичных знаков валюты CCY). |
| AMPSTD | Дата проводки | Текущая операционная дата. Дата указывается в формате MIDAS. |
| AMVALD | Дата валютирования | Если поле Movement.ValueDate заполнено, то Movement.ValueDate; в противном случае — текущая операционная дата. Дата указывается в формате MIDAS. |
| AMDRCR | Признак дебет/кредит | ЕслиMovement.DebitCredit = 'D', то 0. ЕслиMovement.DebitCredit = 'C', то 1. |
| AMSPOS | Источник проводок | Movement.SPOS |
| AMOTRF | Идентификатор проводки | Movement.OTRF |
| AMDPMT | Департамент | Movement.Department |
| AMPRFC | Центр прибыли | Movement.ProfitCenter |
| AMBOKC | Book Code | Movement.BookCode |
| AMTRAT | Тип транзакции | Если поле Movement.PostingTrnTyp заполнено, то Movement.PostingTrnTyp; в противном случае 92000. |
| AMPNAR | Английский комментарий к проводке | Movement.PostingNarrative |
| AMTRYP | Transaction Type для проджекта | Movement.ProjectTrnTyp |
| AMTNMR | Transaction Number для проджекта | Movement.ProjectTrnNbr |
| AМNRTD | Комментарий к проджекту | Movement.ProjectNarrative |
| AMOED | Дата и время поступления требования к счёту | Movement.EntryDate |
| AMRNAR | Русский комментарий к проводке | Movement.Comment |
| AMSRNAR | Короткий русский комментарий к проводке | Movement.ShortComment |
| AMUSEOVD | Учитывать лимит овердрафта | Movement.UseOverdraft |
| AMIGNBLK | Флаг обхода блокировки all/credit/debit | Movement.IgnoreBlockFlag |
| AMIGNBAL | Флаг игнорирования доступного остатка | Movement.IgnoreBalance |
| AMSTORNO | Движение является СТОРНО | Movement.Storno |
| AMPRIMEID | Идентификатор блокировки средств на карте в системе Prime | Movement.PrimeBlockID |
| AMFAN | Флаг необходимости создавать веерные проводки | Movement.Fan |
| AMPSTRRN | RRN проводки в GLPOSTPA | Пока не заполнять |
| AMPRJRRN | RRN проджекта в RSACMTPD | Пока не заполнять |

Если IgnoreBlockFlag = '0', проверить, что по счёту не установлен запрет на осуществление движения (поле ACCNTAB.RETB).

ACCNTAB.RETB

|  |  |  |
| --- | --- | --- |
| Номер бита | Описание блока | Вывод |
| 0 | Refer all debits | Запрещены дебетовые движения |
| 1 | Refer all credits | Запрещены кредитовые движения |
| 2 | Block all debits | Запрещены дебетовые движения |
| 3 | Block all credits | Запрещены кредитовые движения |
| 4 | Inactive Account | Запрещены любые движения |
| 6 | Bankrupt/liquidated | Запрещены любые движения |
| 7 | Bad Debt | Запрещены любые движения |

Если запрет стоит, зафиксировать ошибку с кодом «79».

Если передан идентификатор блокировки (Movement.BlockID), проверить, существует ли блокировка (запись в таблице IFXPPABP) с таким идентификатором.

Если такая блокировка существует и она неподтверждённая, убедиться в совпадении её полей ObjectReference, Account, Amount (сравнивается с полем ABAMT), UseOverdraft, Comment с соответствующими полями запроса. Каждое из перечисленных полей либо должно быть в обоих случаях незаполнено, либо заполнено одним и тем же значением.  
Если обнаружились расхождения, зафиксировать ошибку с кодом «80».

Если такая блокировка существует и она подтверждённая, убедиться в совпадении её полей ObjectReference, Account, Amount (сравнивается с полем ABAMT), UseOverdraft, IgnoreBlockFlag, IgnoreBalance, Comment с соответствующими полями запроса. Каждое из перечисленных полей либо должно быть в обоих случаях незаполнено, либо заполнено одним и тем же значением.  
Если обнаружились расхождения, зафиксировать ошибку с кодом «81».

Если такая блокировка существует, считать запись по блокировке в память и внести следующие изменения (обновление записи в БД будет произведена позже — в пункте 4.5.6.10):

IFXPPABP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Новое значение |
| ABSTS | Статус блокировки (активна или удалена) | 'M' |
| ABDLTD | Дата удаления | Текущая операционная дата |

Если такая блокировка существует и она подтверждённая, сторнировать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*STORNO' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) не равен '0', зафиксировать ошибку с кодом «85» и сообщением p@ErrM.

Обновить подготовленную запись файла IFXPPABP из параметра p@Block (программа IFXPPMPR могла изменить ряд полей).

Обновить запись в файле IFXPPABP.

Если IgnoreBalance = '0', с помощью прогаммы MMBGLBLR определить остаток на счёте. В противном случае считать, что на счёте достаточно денег для любой операции (p1Balance = \*HIVAL).

Параметры запуска

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p1Mode | Режим работы | Если UseOverdraft = '1', то '\*WITH\_OVD'.  Если UseOverdraft = '0', то '\*NO\_OVD'. |
| p1MIAC | Счёт MIDAS (в формате ClientCcyAcodSqBrc) | CNUM + CCY + ACOD + ACSQ + BRCA |
| p1CBAC | Счёт ЦБ | Account |
| p1Balance | Сумма |  |
| p1AP | Сторона баланса счёта (активный/пассивный) |  |
| p1RtCd | Код возврата |  |
| p1ErrM | Сообщение об ошибке |  |

Если код возврата программы MMBGLBLR (p1RtCd) равен ‘4’ (что означает «Счёт не ритейловый»), тогда не считаем это ошибкой, но в дальнейшем (в п.4.5.6.21) не создаём по этому счёту проджект. Кроме того, поле «Transaction type» (IFXPPAMP.AMTRAT) в этом случае равно 00000.

Если движение дебетовое и счёт пассивный (p1AP = '\*PASSIVE'), проверить, что сумма предполагаемого движения не превышает доступный остаток по счёту (p1Balance). Если денег на счёте недостаточно, зафиксировать ошибку с кодом «82».

Аналогичная проверка баланса производится в случае кредитового движения и активного счёта.

Ввести основную полупроводку с помощью программы MMBGLGPI2:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| Mode | Режим работы | 'W' |
| BRCA | MIDAS account | AMBRCA |
| CNUM | AMCNUM |
| CCY | AMCCY |
| ACOD | AMACOD |
| ACSQ | AMACSQ |
| PSTD | Posting date | 0 |
| VALD | Value date | AMVALD |
| PSTA | Posting Amount | AMPSTA |
| DRCR | Debit/Credit indicator | AMDRCR |
| PNAR | Narrative | AMPNAR |
| OTRF | Original Transaction Reference | AMOTRF |
| ASOC | Associated customer | CNUM из контрпроводки |
| SPOS | Source of posting | AMSPOS |
| DPMT | Department | AMDPMT |
| SWCR | SWIFT common reference | AMID |
| PRFC | Profit center | AMPRFC |
| BOKC | Book code | AMBOKC |
| TRAT | Transaction type | AMTRAT |
| ErrN | Код ошибки |  |
| ErrM | Текст ошибки |  |
| RRN | RRN созданной записи (в GLPOSTPA) |  |

Проанализировать результат работы программы. Если код ошибки (ErrN) не равен 0, зафиксировать ошибку с кодом «86».

Записать RRN созданной записи в поле AMPSTRRN.

Проверить, что существует счёт парной полупроводки:

|  |  |  |
| --- | --- | --- |
| BRCA | MIDAS account | На основе бранча основной полупроводки (из п. 4.5.6.15) вычислить бранч головного офиса. Это и будет BRCA парной полупроводки.  Далее, проверить, что существует запись в таблице IFXPPTAP по ключу BRCA.  Если не существует, зафиксировать ошибку с кодом «98» |
| CNUM | Внутренний клиент отделения BRCA (SDBRCHPD.A8BICN) |
| CCY | CCY из основной полупроводки |
| ACOD | Определяется по файлу IFXPPTAP для отделения BRCA (IFXPPTAP. TAACOD) |
| ACSQ | Определяется по файлу IFXPPTAP для отделения BRCA (IFXPPTAP. TAACSQ) |

Если счёт не существует, создать его с помощью программы CBDRCOA1C.

Параметры вызова

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Описание | Размерность | Что передать |
| p6\_Account | Счёт MIDAS (в формате BrcClientCcyAcodSq) | Char(18) | BRCA + CNUM + CCY + ACOD + ACSQ |
| p6\_AccName | Short account name | Char(20) | IFXPPTAP.TAACNM |
| p6\_QTemp | Create in QTEMP work file flag | Char(1) | ‘Y’ |
| p6\_DACO | Дата открытия счёта | Decimal(5,0) | Текущая дата в формате Midas |
| p6\_RtCd | Код возврата | Char(1) | Выходной параметр:  ‘N’ = счёт успешно создан  ‘Y’ = ошибка при создании |
| p6\_ErrM | Сообщение об ошибке | Char(80) | Выходной параметр. Заполняется в случае ошибки, иначе пусто. |

Ввести парную полупроводку с помощью программы MMBGLGPI2:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| Mode | Режим работы | 'W' |
| BRCA | MIDAS account | из п. 4.5.6.18 |
| CNUM | из п. 4.5.6.18 |
| CCY | из п. 4.5.6.18 |
| ACOD | из п. 4.5.6.18 |
| ACSQ | из п. 4.5.6.18 |
| PSTD | Posting date | 0 |
| VALD | Value date | VALD из основной полупроводки. |
| DRCR | Debit/Credit indicator | Противоположный к DRCR из основной полупроводки. |
| PNAR | Narrative | PNAR из основной полупроводки. |
| OTRF | Original Transaction Reference | OTRF из основной полупроводки. |
| ASOC | Associated customer | CNUM из основной полупроводки |
| SPOS | Source of posting | SPOS из основной полупроводки. |
| DPMT | Department | DPMT из основной полупроводки. |
| SWCR | SWIFT common reference | SWCR из основной полупроводки. |
| PRFC | Profit center | PRFC из основной полупроводки. |
| BOKC | Book code | BOKC из основной полупроводки. |
| TRAT | Transaction type | 00000 |
| ErrN | Код ошибки |  |
| ErrM | Текст ошибки |  |
| RRN | RRN созданной записи (в GLPOSTPA) |  |

Проанализировать результат работы программы. Если код ошибки (ErrN) не равен 0, зафиксировать ошибку с кодом «86».

Ввести проджект с движением с помощью программы MMBGLSU0[[16]](#footnote-17):

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p0Mode | Режим работы | ' ' (Пробел) |
| p0BRCA | Midas account | AMBRCA |
| p0CNUM | AMCNUM |
| p0CCY | AMCCY |
| p0ACOD | AMACOD |
| p0ACSQ | AMACSQ |
| p0PSTA | Movement Amount | AMPSTA |
| p0DRCR | Debit or Credit | AMDRCR |
| p0VALD | Value Date | AMVALD |
| p0TRYP | Transaction Type | AMTRYP |
| p0TNMR | Transaction Number | AMTNMR |
| p0PNAR | Narrative Description | AMNRTD |
| p0CQNR | Cheque Number | 30 |
| p0RvIn | Признак сторнирования | ' ' (Пробел) |
| p0RtCd | Код возврата |  |
| p0ErrM | Сообщение об ошибке |  |
| p0RRN | RRN проджекта в RSACMTPD |  |

Проанализировать результат работы программы. Если код возврата (p0RtCd) не равен 0, зафиксировать ошибку с кодом «87».

Записать RRN созданной записи в поле AMPRJRRN.

Добавить запись, подготовленную в пунктах 4.5.6.1, 4.5.6.17 и 4.5.6.22, в файл IFXPPAMP.

### Для формата AEMovementCreate

#### Производятся те же действия, что и для формата QueryMovementCreate со следующими отличиями:

Вместо создания проводок по одной на каждое движения при помощи программы MMBGLGPI2, производится создание проводок сразу по паре движений по дебету и по кредиту при помощи программы MMBGLAPR:

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| Mode | Режим работы | '\*WORK' |
| Posting1 | Проводка по дебету | Структуру dsGLPOSTPA, заполненную аналогично заполнению полей для вызова MMBGLGPI2 при обработке формата QueryMovementCreate при создании движения по дебету |
| Posting2 | Проводка по кредиту | Структуру dsGLPOSTPA, заполненную аналогично заполнению полей для вызова MMBGLGPI2 при обработке формата QueryMovementCreate при создании движения по кредиту |
| CreateMainPostings | Создавать ли основые проводки | '\*CREATE' |
| ConversionInBranch | Конверсия в бранче по дебету или по кредиту | '\*DEBIT' |
| ErrN | Код ошибки |  |
| ErrM | Текст ошибки |  |
| NumOfCreatedPostings | Количество созданных полупроводок |  |
| CreatedPostings | Созданные полупроводки |  |
| CreatedPostingsRRNs | RRN созданных полупроводок |  |

В случае ошибки при вызове данной программы возвращается ошибка с кодом 88.

Балансы проверяются только по ритейловым счетам, аналогично проджекты создаются только по ритейловым счетам. Кроме того, по неритейловым счетам используется Transaction type 00000;

## Технические подробности

Программа AOMPOSU0 требует для своей работы Commitment Control и запускается в Default Activation Group. Хендлер адаптера должен запускаться в Activation Group \*CALLER и не может стартовать Commitment Control на уровне \*JOB из-за конфликтов с Commitment Control'ем самого адаптера (тот делает MQBEGIN).

При этом функциональность программ, работающих в Default Activation Group, существенно ограничена (например, они не могут иметь процедуры).

Для решения этой проблемы используется следующий workaround:

* В качестве хендлера MA указывается программа IFXPPMH1C, которая запускается в Activation Group \*CALLER и вызывает программу IFXPPMH2R.
* Программа IFXPPMH2R запускается в Default Activation Group, стартует Commitment Control на уровне \*ACTGRP (при первом запуске) и реализует всю логику, описанную в разделе 4.

# Прочие изменения

## IFXPPSTR — RPGLE — Start / stop requests processing

Новая программа, работающая в COB Before и COB After.

Параметры запуска

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Тип | Входной/выходной | Описание |
| p@Action | Char(10) | Входной | Режим работы |

Параметр p@Action имеет два допустимых значения: \*START и \*STOP.

Режим \*START

Проверить, что в системе уже запущен 75-й поток MA. Если нет, запустить его.

Проверить, что в системе уже запущен 76-й поток MA. Если нет, запустить его.

Отправить во входную очередь каждого адаптера (75 и 76 потоки) MQ-сообщение в формате ControlCommand с указанием Action = \*START (можно начинать отвечать на запросы).

Режим \*STOP

Отправить во входную очередь каждого адаптера (75 и 76 потоки) MQ-сообщение в формате ControlCommand с указанием Action = \*STOP (необходимо закрыть файлы и перестать отвечать на запросы).

## IFXPPMPR — RPGLE — Make project for block

Новая программа (API), осуществляющая ввод или сторнирование проджекта по подтверждённой блокировке.

Параметры запуска

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Тип | Входной/выходной | Описание |
| p@Action | Char(10) | Входной | Режим работы |
| p@Block | IFXPPABP % RowType | Входно-выходной | Блокировка |
| p@RtCd | Char(1) | Выходной | Код возврата |
| p@ErrM | Char(80) | Выходной | Текст ошибки |

Допустимые значения параметра p@Action

|  |  |
| --- | --- |
| Значение | Смысл |
| \*INPUT | Создание проджекта при первоначальном вводе блокировки |
| \*RECREATE | Утреннее пересоздание проджекта по уже существующей блокировке |
| \*STORNO | Сторнирование проджекта |
| \*TERMINATE | Освобожение ресурсов и выход из программы |

Параметр p@Block должен содержать прочитанную запись файла IFXPPABP (чтение самого файла не производится).

Параметр p@RtCd может принимать следующие значения:

|  |  |
| --- | --- |
| Значение | Смысл |
| '0' | Обработка прошла успешно |
| '1' | Недостаточно денег на счёте |
| '2' | Прочие ошибки |

В случае p@RtCd = '2' параметр p@ErrM будет содержать сообщение об ошибке.

Алгоритм работы

Если p@Action = '\*TERMINATE', вызвать программу MMBGLSU0 с кодом действия p@Mode = 'C', взвести индикатор LR и завершить работу.

Проверить, что проджект по блокировке ещё не создан (p@Block.ABLPD меньше текущей даты). В противном случае закончить работу с кодом '0'.

Проверить, что текущая операционная дата попадает в диапазон между p@Block.ABSTRD и p@Block.ABENDD. В противном случае закончить работу с кодом '0'.

Провести валидацию входных параметров. Если хотя бы одна не пройдена, завершить работу с ошибкой (и кодом '2').

Проверить, что блокировка подтверждённая (ABTYPE = 'A').

Проверить, что статус блокировки допускает запрошенное действие:

|  |  |
| --- | --- |
| Статус (ABSTS) | Допустимое действие (p@Action) |
| A | \*INPUT, \*RECREATE, \*STORNO |
| D, M | \*STORNO |

Проверить, что счёт присутствует в системе и не является закрытым.

(Только в режиме \*INPUT) Если p@Block.ABIGNBLK = '0', проверить, что по счёту не стоит запрета на дебетовые движения (биты номер 0, 2, 4, 6, 7 поля ACCNTAB.RETB).

(Только в режиме \*INPUT) Если p@Block.ABIGNBAL = '0', с помощью программы MMBGLBLR определить остаток на счёте. В противном случае считать, что на счёте достаточно денег для любой операции (остаток = \*HIVAL).

Параметры запуска

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p1Mode | Режим работы | Если p@Block.ABUSEOVD = '1', то '\*WITH\_OVD'.  Если p@Block.ABUSEOVD = '0', то '\*NO\_OVD'. |
| p1MIAC | Счёт MIDAS (в формате ClientCcyAcodSqBrc) | p@Block.ABCNUM + p@Block.ABCCY + p@Block.ABACOD + p@Block.ABACSQ + p@Block.ABBRCA |
| p1CBAC | Счёт ЦБ | p@Block.ABCBAC |
| p1Balance | Сумма |  |
| p1AP | Сторона баланса счёта (активный/пассивный) |  |
| p1RtCd | Код возврата |  |
| p1ErrM | Сообщение об ошибке |  |

Если работа программы завершилась с ошибкой (p1RtCd = '1'), также завершить работу с ошибкой (c кодом '2').

Если счёт активный (p1AP = '\*ACTIVE') и p@Block.ABAMTTYPE ≠ '\*AMOUNT', завершить работу с ошибкой "Для блокировки по активному счёту необходимо явное указание суммы" (c кодом '2').

(Только в режиме \*INPUT) Вычислить требуемую сумму блокировки.

Если p@Block.ABAMTTYPE = '\*AMOUNT', ничего делать не нужно, искомая сумма уже записана в p@Block.ABAMT.

Если p@Block.ABAMTTYPE = '\*EXCESS', присвоить p@Block.ABAMT = {остаток на счёте}[[17]](#footnote-18) – p@Block.ABEXCLIM.

Если p@Block.ABAMTTYPE = '\*PERCENT', присвоить p@Block.ABAMT = {остаток на счёте} \* p@Block.ABPERCENT / 100. Округление производится по математическим правилам.

Если полученная сумма (p@Block.ABAMT) меньше либо равна нулю, завершить работу с ошибкой "на счёте недостаточно средств".

Сравнить сумму блокировки (пункт 5.2.8) с остатком на счёте (пункт 5.2.5). Если денег на счёте недостаточно для установки запрошенной блокировки, завершить работу с кодом '1'.

Вычислить требуемую дату валютирования.

Если p@Block.ABVALTYP = '\*CURDATE', использовать текущую дату.

Если p@Block.ABVALTYP = '\*ENDDATE', использовать дату окончания срока блокировки (p@Block.ABENDD).

Ввести проджект с движением с помощью программы MMBGLSU0:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p0Mode | Режим работы | ' ' (Пробел) |
| p0BRCA | Midas account | p@Block.ABBRCA |
| p0CNUM | p@Block.ABCNUM |
| p0CCY | p@Block.ABCCY |
| p0ACOD | p@Block.ABACOD |
| p0ACSQ | p@Block.ABACSQ |
| p0PSTA | Movement Amount | p@Block.ABAMT |
| p0DRCR | Debit or Credit | Если p@Action in ('\*INPUT', '\*RECREATE'), то 0. Если p@Action = '\*STORNO', то 1. |
| p0VALD | Value Date | См. пункт 5.2.10 |
| p0TRYP | Transaction Type | p@Block.ABTRYP |
| p0TNMR | Transaction Number | p@Block.ABTNMR |
| p0PNAR | Narrative Description | p@Block.ABNRTD |
| p0CQNR | Cheque Number | Если p@Action in ('\*INPUT', '\*RECREATE'), то 10. Если p@Action = '\*STORNO', то 20. |
| p0RvIn | Признак сторнирования | ' ' (Пробел) |
| p0RtCd | Код возврата |  |
| p0ErrM | Сообщение об ошибке |  |
| p0RRN | RRN проджекта в RSACMTPD |  |

Проанализировать результат работы программы. Если код возврата (p0RtCd) не равен 0, присвоить p@ErrM = p0ErrM и завершить работу с ошибкой (и кодом '2').

Обновить параметр p@Block и завершить работу программы с кодом '0'.

p@Block

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Новое значение |
| ABAMT | Сумма блокировки | См. пункт 5.2.8 |
| ABLPD | Дата создания проджекта | Текущая операционная дата |
| ABRRN | RRN проджекта в RSACMTPD | Если p@Action in ('\*INPUT', '\*RECREATE'), то p0RRN.  Иначе не изменяется. |
| ABSTNRRN | RRN сторнирующего проджекта в RSACMTPD | Если p@Action = '\*STORNO', то p0RRN.  Иначе не изменяется. |

## MMBGLBLR — RPGLE — Определение баланса по счёту

Новая программа (API), позволяющая определить баланс клиентского (retail) счёта (как активного так и пассивного) с учётом проджектов и (опционально) лимита овердрафта. Используется cleared balance.

Параметры запуска

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Тип | Входной/выходной | Описание |
| p@Mode | Char(10) | Входной | Режим работы |
| p@MIAC | Char(18) | Входно-выходной | Счёт MIDAS (в формате ClientCcyAcodSqBrc) |
| p@CBAC | Char(20) | Входно-выходной | Счёт ЦБ |
| p@Balance | Decimal(15 0) | Выходной | Сумма |
| p@AP | Char(10) | Выходной | Сторона баланса счёта (активный/пассивный) |
| p@RtCd | Char(1) | Выходной | Код возврата |
| p@ErrM | Char(80) | Выходной | Сообщение об ошибке |

Программе можно передать только счёт MIDAS (в этом случае счёт ЦБ будет определён автоматически), только счёт ЦБ (в этом случае счёт MIDAS будет определён автоматически), или оба счёта (в этом случае соответствие переданных счетов друг другу не проверяется).

Допустимые значения параметра p@Mode

|  |  |
| --- | --- |
| Значение | Смысл |
| \*WITH\_OVD | Подсчёт баланса с учётом лимита овердрафта |
| \*NO\_OVD | Подсчёт баланса без учёта лимита овердрафта |
| \*TERMINATE | Освобожение ресурсов и выход из программы |

Алгоритм работы

Если p@Mode = '\*TERMINATE', закрыть все файлы и завершить работу.

Если заполнен ровно один из счетов MIDAS и ЦБ, определить второй и записать его в соответствующий параметр. Если счёт найти не удалось, завершить работу с ошибкой.

Определить сторону баланса счёта[[18]](#footnote-19) и записать её в параметр p@AP ('\*ACTIVE' — счёт активный; '\*PASSIVE' — счёт пассивный).

Определить тип счёта (ACCNTAB.ATYP). Если он отличен от 'R' (retail), завершить работу с ошибкой.

Определить текущий баланс счёта по полю MEMOS.CLBLN (Cleared Balance). Если p@Mode = '\*WITH\_OVD' и дата окончания срока действия овердрафта (ACCNTAB.ODED) не меньше текущей, к балансу счёта следует добавить сумму настроенного лимита (ACCNTAB.ODLN[[19]](#footnote-20)).

Для пассивных счетов следует изменить знак полученной суммы, чтобы всегда возвращалась величина доступного остатка по счёту.

Завершить работу программы с кодом '0'.

## MMBGLAPR — RPGLE — Создание проводок межбранча и конверсии

Новая программа (API), позволяющая при необходимости создать проводки для межбранча и конверсии. При этом конверсия возможна как в бранче по дебету, так и в бранче по кредиту (Параметр ConversionInBranch \*DEBIT/\*CREDIT)

На вход принимаются две записи dsGLPOSTPA - одна по дебету, вторая по кредиту. Эти основные проводки могут быть уже созданы снаружи (CreateMainPostings='\*ONLYTECH'), либо будут созданы данной API (CreateMainPostings='\*CREATE')

При необходимости возвращается массив созданных проводок и их RRN. Если не передан параметр p0\_ReturnCode, то в случае ошибок API завершает работу аварийно

Параметры запуска

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Тип | Входной/выходной | Описание |
| p@Mode | Char(10) | Входной | Режим работы  \*WORK/\*TERMINATE |
| p@dsPosting1 | ds GLPOSTPA | Входной | Первая полупроводка |
| p@dsPosting2 | ds GLPOSTPA | Входной | Вторая полупроводка |
| p@CreateMainPostings | Char(10) | Входной | Создавать ли основные проводки \*CREATE/\*ONLYTECH |
| p@ConversionInBranch | Char(10) | Входной | Бранч конверсии  \*DEBIT/\*CREDIT |
| p@ReturnCode | Char(1) | Выходной | Код возврата |
| p@ReturnMsg | Char(80) | Выходной | Текст ошибки |
| p@NumOfCreatedPostings | 3i 0, dim(8) | Выходной | Количество созданных полупроводок |
| p@CreatedPostings | ds GLPOSTPA, dim(8) | Выходной | Массив созданных полупроводок |
| p@CreatedPostingsRRN | 9B 0, dim(8) | Выходной | Массив RRN'ов созданных полупроводок |

Алгоритм работы

Если p@Mode = '\*TERMINATE', закрыть все файлы и завершить работу.

Проверить, что одна полупроводка по дебету, а вторая по кредиту. Если это не так, то вернуть ошибку «01 Одна полупроводка должна быть по дебету, вторая - по кредиту» с кодом «1»;

Проверить, что валюты и отделения проводок корректны. Если это не так, то вернуть ошибку «Отделение не найдено в SDBRCHPD:» или «Валюта не найдена в SDCURRPD или MMBGLBCP:»;

Если валюты по дебету и кредиту одинаковые, то проверить, что совпадают суммы. Иначе вернуть ошибку «Если валюты по Дб и Кр совпадают, то и суммы должны совпадать»;

Если p@CreateMainPostings = '\*CREATE ', то поправить поля Associated customer основных проводок (см. описание ниже) и создать их вызовом программы MMBGLGPI2;

Создать необходимые полупроводки для межбранча и конверсии (см. описание ниже);

Если передан параметр p@NumOfCreatedPostings, то вернуть количество созданных полупроводок;

Если передан параметр p@CreatedPostings, то вернуть структуру каждой созданной полупроводки;

Если передан параметр p@CreatedPostingsRRN, то вернуть массив RRN'ов созданных полупроводок;

В случае ошибок если не передан параметр p@ReturnCode, то завершить работу программы с исключением;

В случае ошибки при создании полупроводок прогаммой MMBGLGPI2 вернуть ошибку «07 Не удалось создать проводку по счёту:»;

Схема создания проводок:

Определить клиентов отделений по дебету DbBrcaCust и по кредиту CrBrcaCust по таблице SDBRCHPD;

Определить interBrcaACOD и interBrcaACSQ по таблице MMBGLIBP (обычно это 1305-01);

Определить positionACOD и positionACSQ: это 6005-01;

Определить DbCcyBceACOD, DbCcyBceACSQ, CrCcyBceACOD и CrCcyBceACSQ по таблице MMBGLBCP;

1) У проводок DbCCY = CrCCY = eqCCY и DbBRCA = CrBRCA = eqBRCA:

Ничего лишнего не создаём

DbCnum eqCCY DbACOD DbACSQ eqBRCA -> CrCnum eqCCY CrACOD CrACSQ eqBRCA

2) У проводок DbCCY = CrCCY = eqCCY и DbBRCA <> CrBRCA:

Валюта по Дб и Кр совпадает, бранчи отличаются. Тогда создаются

следующие пары полупроводок:

- No 1. Дб: Счет дебета из проводки, сумма по дебету.

Кр: Счет МФР в отделении по дебету, валюта счета дебета, сумма по дебету.

- No 2. Дб: Счет МФР в отделении по кредиту, валюта счета дебета, сумма по дебету.

Кр: Счет кредита из проводки, сумма по кредиту.

DbCnum eqCCY DbACOD DbACSQ DbBRCA -> CrBrcaCust eqCCY 1305 01 DbBrch

DbBrcaCust eqCCY 1305 01 CrBrca -> CrCnum eqCCY CrACOD CrACSQ CrBRCA

3) У проводок DbCCY <> CrCCY и DbBRCA = CrBRCA = eqBRCA:

Бранчи по Дб и Кр совпадают, валюта отличается. Тогда создаются

следующие пары полупроводок:

- No 1. Дб: Счет дебета из проводки, сумма по дебету.

Кр: Счет позиции для валюты по дебету, сумма по дебету.

- No 2. Дб: Счет позиции для валюты по кредиту, сумма по кредиту.

Кр: Счет кредита из проводки, сумма по кредиту.

- No 3. Дб: Счет BCE для валюты по дебету, сумма рублевого эквивалента проводки.

Кр: Счет BCE для валюты по кредиту, сумма рублевого эквивалента проводки.

DbCnum DbCCY DbACOD DbACSQ eqBRCA -> eqBrcaCust DbCCY 6005 001 eqBrca

eqBrcaCust DbCCY 6005 001 eqBrca -> CrCnum eqCCY CrACOD CrACSQ eqBRCA

eqBrcaCust RUR DbCCYAcod eqBrca -> eqBrcaCust RUR CrCCYAcod eqBrca

4) У проводок DbCCY <> CrCCY и DbBRCA <> CrBRCA, конверсия в дебете

Бранчи по Дб и Кр отличаются, валюты также различны, конверсия в отеделении по дебету

Тогда создаются следующие пары полупроводок:

No 1. Дб: Счет дебета из проводки, сумма по дебету.

Кр: Счет позиции для валюты по дебету, сумма по дебету.

No 2. Дб: Счет позиции для валюты по кредиту, сумма по кредиту.

Кр: Счет МФР в отделении по дебету, валюта счета кредита, сумма по кредиту

No 2. Дб: Счет МФР в отделении по кредиту, валюта счета кредита, сумма по кредиту

Кр: Счет кредита из проводки, сумма по кредиту.

No 4. Дб: Счет BCE для валюты по дебету, сумма рублевого эквивалента проводки.

Кр: Счет BCE для валюты по кредиту, сумма рублевого эквивалента проводки.

DbCnum DbCCY DbACOD DbACSQ DbBRCA -> DbBrcaCust DbCCY 6005 001 DbBrca

DbBrcaCust CrCCY 6005 001 DbBrca -> CrBrcaCust CrCCY 1305 01 DbBrch

DbBrcaCust CrCCY 1305 01 CrBrca -> CrCnum CrCCY CrACOD CrACSQ CrBRCA

DbBrcaCust RUR DbCCYAcod DbBrca -> DbBrcaCust RUR CrCCYAcod DbBrca

5) У проводок DbCCY <> CrCCY и DbBRCA <> CrBRCA, конверсия в кредите

Бранчи по Дб и Кр отличаются, валюты также различны, конверсия в отеделении по кредиту

Тогда создаются следующие пары полупроводок:

No 1. Дб: Счет дебета из проводки, сумма по дебету.

Кр: Счет МФР в отделении по дебету, валюта счета дебета, сумма по дебету.

No 2. Дб: Счет МФР в отделении по кредиту, валюта счета дебета, сумма по дебету

Кр: Счет позиции для валюты по дебету, сумма по дебету.

No 3. Дб: Счет позиции для валюты по кредиту, сумма по кредиту.

Кр: Счет кредита из проводки, сумма по кредиту.

No 4. Дб: Счет BCE для валюты по дебету, сумма рублевого эквивалента проводки.

Кр: Счет BCE для валюты по кредиту, сумма рублевого эквивалента проводки.

DbCnum DbCCY DbACOD DbACSQ DbBRCA -> CrBrcaCust DbCCY 1305 01 DbBrch

DbBrcaCust DbCCY 1305 01 CrBrca -> CrBrcaCust DbCCY 6005 001 CrBrca

CrBrcaCust CrCCY 6005 001 CrBrca -> CrCnum CrCCY CrACOD CrACSQ CrBRCA

CrBrcaCust RUR DbCCYAcod CrBrca -> CrBrcaCust RUR CrCCYAcod CrBrca

При заполеннии Associated customer полупроводки используется поле Customer парной полупроводки.

## MMBGLSU0 — RPGLE — Make project and return its RRN

Новая программа (API), позволяющая ввести проджект и узнать RRN созданной записи в RSACMTPD. Дополнительно в нормальном виде (выходным параметром) возвращается текст сообщения об ошибке.

Программа является обёрткой вокруг AOMPOSU0.

Параметры запуска

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Тип | Входной/выходной | Описание |
| p@Mode | Char(1) | Входной | Режим работы |
| p@BRCA | Char(3) | Входной | Отделение |
| p@CNUM | Numeric(6 0) | Входной | Клиентский номер |
| p@CCY | Char(3) | Входной | Валюта |
| p@ACOD | Numeric(4 0) | Входной | Код счёта |
| p@ACSQ | Numeric(2 0) | Входной | Сиквенс счёта |
| p@PSTA | Decimal(13 0) | Входной | Сумма |
| p@DRCR | Decimal(1 0) | Входной | Дебет/кредит |
| p@VALD | Decimal(5 0) | Входной | Дата валютирования |
| p@TRYP | Char(2) | Входной | Тип операции |
| p@TNMR | Char(6) | Входной | Номер операции |
| p@PNAR | Char(30) | Входной | Комментарий |
| p@CQNR | Numeric(8 0) | Входной | Номер чека |
| p@RvIn | Char(1) | Входной | Признак сторнирования |
| p@RtCd | Decimal(3 0) | Выходной | Код возврата |
| p@ErrM | Char(80) | Выходной | Сообщение об ошибке |
| p@RRN | Binary(9 0) | Выходной | RRN проджекта в RSACMTPD |

Примечания

* Тип параметров CNUM, ACOD, ACSQ, CQNR изменён (по сравнению с AOMPOSU0) с Decimal на Numeric для единообразия.
* Тип параметра DRCR изменён (по сравнению с AOMPOSU0) с Char(1) на Decimal(1 0) для единообразия. Возможные значения: 0 — дебет; 1 — кредит.
* В случае ошибки индикаторы U7 и U8 гасятся автоматически.
* В поле p@RtCd (код возврата) возвращается содержимое поля LDA.DBASE, непоспредственно описывающее ошибку.
* Добавлен выходной параметр p@ErrM, содержащий стандартный текст ошибки (определяется по LDA.DBASE).
* Добавлен необязательный выходной параметр p@RRN, содержащий RRN сделанной записи в файле RSACMTPD.
* Логика обработки параметра p@RvIn изменена по сравнению с оригинальной программой. 29-ый символ комментария не затирается. 30-ый символ переписывается (заменяется на звёздочку) только если он не заполнен.
* Параметр p@Mode (режим запуска) сделан первым. Для закрытия файлов и освобождения ресурсов может использоваться как код 'C' (как в AOMPOSU0) так и 'T' (общий стандарт в ПО in-house).
* При первом запуске Data Area LDA создаётся автоматически.

Алгоритм работы

При первом запуске создать в библиотеке QTEMP Data Area с именем LDA длиной 256 символов. Если она уже существует, ничего не делать.

Если p@Mode равен 'C' или 'T', запустить программу AOMPOSU0 в режиме освобождения ресурсов (Mode = 'C'), закрыть все файлы и завершить работу.

Если p@RvIn = 'Y' и последний символ p@PNAR пустой (равен пробелу), изменить последний символ p@PNAR на звёздочку (\*).

Вызвать программу AOMPOSU0:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| BRCA | Midas account | p@BRCA |
| CNUM | cast p@CNUM as Decimal(6 0) |
| CCY | p@CCY |
| ACOD | cast p@ACOD as Decimal(4 0) |
| ACSQ | cast p@ACSQ as Decimal(2 0) |
| PSTA | Movement Amount | p@PSTA |
| DRCR | Debit or Credit | Если p@DRCR = 0, то 'D'. Иначе 'C'. |
| VALD | Value Date | p@VALD |
| TRYP | Transaction Type | p@TRYP |
| TNMR | Transaction Number | p@TNMR |
| PNAR | Narrative Description | Если p@RvIn = 'Y' and %Subst(p@PNAR: 30: 1) = ' ', то %Subst(p@PNAR: 1: 29) + '\*'. Иначе p@PNAR. |
| CQNR | Cheque Number | cast p@CQNR as Decimal(8 0) |
| Mode | Режим работы | p@Mode |
| RvIn | Признак сторнирования | ' ' (Пробел) |
| Rtrn | Код возврата |  |

Если код возврата (Rtrn) не равен ' ' (произошла ошибка):

потушить индикаторы U7 и U8;

прочитать Data Area LDA;

определить по полю DBASE текст сообщения об ошибке (см. приложение I.VI) и записать его в выходной параметр p@ErrM;

если значение DBASE лежит на отрезке [1, 14], записать его в выходной параметр p@RtCd; в противном случае положить p@RtCd = 15;

Если параметр p@RRN передан, положить p@RRN = 0.

Если код возврата (Rtrn) равен ' ' (обработка прошла успешно):

положить p@RtCd = 0 и p@ErrM = \*Blanks;

если параметр p@RRN передан, определить RRN сделанной записи в файле RSACMTPD; записать его в выходной параметр p@RRN.

## IFXPPRPR — RPGLE — Recreate projects (in the morning)

Новая программа, работающая в COB After. Она осуществляет утреннее пересоздание проджектов по всем подтверждённым блокировкам.

Параметры запуска

* Нет

Алгоритм работы

Для каждой записи файла IFXPPABL, описывающей подтверждённую блокировку (ABTYPE = 'A') выполнить следующие действия:

Пересоздать проджект с помощью программы IFXPPMPR:

Параметры вызова

|  |  |  |
| --- | --- | --- |
| Параметр | Описание | Что передать |
| p@Action | Режим работы | '\*RECREATE' |
| p@Block | Блокировка | Запись файла IFXPPABP по блокировке |
| p@RtCd | Код возврата |  |
| p@ErrM | Текст ошибки |  |

Проанализировать результат работы API. Если код возврата (p@RtCd) не равен '0', записать ошибку в лог IFXPPJRP.

IFXPPJRP

|  |  |  |
| --- | --- | --- |
| Имя поля | Описание | Чем заполнять |
| JRID | Внутренний индентификатор блокировки | ABID |
| JRBLKSRC | Код системы-инициатора блокировки | ABBLKSRC |
| JRBLKID | Идентификатор блокировки во внешней системе | ABBLKID |
| JROBJREF | Референс объекта системы-инициатора | ABOBJREF |
| JRBRCA | Счёт в формате MIDAS | ABBRCA |
| JRCNUM | ABCNUM |
| JRCCY | ABCCY |
| JRACOD | ABACOD |
| JRACSQ | ABACSQ |
| JRDATE | Операционная дата | Текущая операционная дата |
| JRTS | Время | Текущее системное время |
| JRRTCD | Код ошибки | p@RtCd |
| JRERRM | Текст ошибки | p@ErrM |

## IFXSTPRR — RPGLE — Process postings

Данная программа является центральным модулем русификации всех клиенстких проводок. Она осуществляет первичную обрабокту записей и вызывает обработчики, специфичные для отдельных модулей.

Необходимо добавить сохранение поля SWCR проводки для дальнейшего его использования в обработчике IFXST00R:

Записать поле SWCR проводки в поле IFXSTSTP.STSTDNBR.

## IFXPPDDR — RPGLE — Удал. сокращ. проводки AE из GLPOSTPA COB Before

Новая программа, работающая в COB Before. Она осуществляет удаление пар проводка+сторнирующая проводка из таблицы MIDAS проводок прошедшего дня GLPOSTPA.

Программа является клоном существующего функционала для удаления сторнированных проводок K+TP.

Параметры запуска

* Нет

Алгоритм работы

Для каждой записи файла IFXPPAMP за прошедший день (AMPSTD = текущей дате MIDAS), описывающей сторнирующее и еще не удаленное движение (AMSTORNO = '1' и AMRRN <> -1), выполнить следующие действия:

#### Найти сторнированное движение в файле IFXPPAMP.

#### Сторнированное и сторнирующее движения:

#### - совпадают по дате (AMPSTD),

#### - совпадают по счету MIDAS (AMBRCA,AMCNUM,AMCCY,AMACOD,AMACSQ),

#### - совпадают по сумме (AMAMT),

#### - совпадают по референсу (AMOTRF),

#### - имеют противоположное значение флага дебет/кредит (AMDRCR),

#### - нарратив (AMPNAR) совпадает с точностью до знака '\*' в начале.

#### Для каждой найденной пары движений выполнить следующие действия:

##### Собрать все движения, относящиеся к запросу со сторнированной (далее: запрос A) и сторнирующей проводкой (далее: запрос B).

У движений, входящих в один запрос, совпадает признак IFXPPAMP.AMLHID[[20]](#footnote-21).

По IFXPPAMP.AMRRN (RRN записи в GLPOSTPA) движений запроса A построить группу проводок А – проводки, созданные по запросу А.

По IFXPPAMP.AMRRN движений запроса B построить группу проводок B – проводки, созданные по запросу B.

##### Убедиться, что проводки группы B в точности сторнируют все проводки группы A.

У сторнированной и сторнирующей проводки в GLPOSTPA совпадают все сущности кроме:

- нарратива (PNAR): у сторнирующей проводки в начале идет символ '\*',

- индикатор дебет/кредит (DRCR) (его значения у таких проводок противоположны)

В случае, если одна группа проводок в точности сторнирует другую, необходимо выполнить:

###### Удаление проводок из GLPOSTPA.

Из GLPOSTPA необходимо удалить все проводки с RRN = IFXRMAMP.AMRRN по всем проводкам групп A и B.

###### Обновить IFXRMAMP.

Каждому движению запросов A и B необходимо проставить AMRRN = -1.

## IFXSTPNR — RPGLE — Process internal postings

Данная программа — аналог IFXSTPRR для неклиентских проводок.

Необходимо добавить сохранение поля SWCR проводки для дальнейшего его использования в обработчике IFXST00R:

Записать поле SWCR проводки в поле IFXSTSTP.STSTDNBR.

## IFXST00R — RPGLE — Process postings DEFAULT (other modules)

Данная программа осуществляет русификацию проводок тех модулей (source of posting), по которым нет специфичных настроек.

Необходимо добавить в неё следующую логику:

Если поле STSTDNBR (в нём был сохранён SWCR) проводки начинается на константу 'AM', проверить наличие этой проводки в таблице IFXPPAMP (по условию IFXPPAMP.AMID = Posting.STSTDNBR).

Если запись нашлась, взять русский комментарий из неё:

STSTRNAR = IFXPPAMP.AMRNAR

STSTRCNAR = IFXPPAMP.AMSRNAR

## MT940EXTM — PF — Формат окончательной выписки для STAMT (детали операции)

В связи с необходимостью передавать в STAMT источник операции и идентификатор исходного документа в системе-источнике, расширяется формат файла MT940EXTM:

Добавляется новое поле TSRCSYS длиной 32 символа.

Поле TOTRF расширяется до 60 символов.

## MT940SMG — RPGLE — Выгрузка окончательной выписки в STAMT

В связи с необходимостью передавать в STAMT источник операции и идентификатор исходного документа в системе-источнике, расширяется формат файла MT940EXTM.

Алгоритм определения источника проводки дорабатывается следующим образом:

Если SWCR проводки начинается на константу 'AM', проверить наличие этой проводки в таблице IFXPPAMP (по условию IFXPPAMP.AMID = Posting.SWCR).

Если запись нашлась, считаем, что источник — IFXPP.

Алгоритм заполнения деталей операции дорабатывается следующим образом:

Если источник проводки — IFXPP, то в поле TOTRF необходимо указать референс объекта системы-инициатора IFXPPAMP.AMOBJREF; в TSRCSYS следует указать наименование системы-инициатора IFXPPAMP.AMMVMSRC.

Если источник проводки — не IFXPP, то в поле TSRCSYS следует указать константу 'MIDAS'.

## MT942EXTM — PF — Формат внутридневной выписки для STAMT (детали операции)

В связи с необходимостью передавать в STAMT источник операции и идентификатор исходного документа в системе-источнике, расширяется формат файла MT942EXTM:

Добавляется новое поле TSRCSYS длиной 32 символа.

Поле TOTRF расширяется до 60 символов.

## MT942SMG — RPGLE — Выгрузка внутридневной выписки в STAMT

В связи с необходимостью передавать в STAMT источник операции и идентификатор исходного документа в системе-источнике, расширяется формат файла MT942EXTM.

Алгоритм определения источника проджекта дорабатывается следующим образом:

Проанализировать значение поля CQNR:

|  |  |  |
| --- | --- | --- |
| CQNR | Описание | В каком файле искать детали операции |
| 10 | Проджект по блокировке | IFXPPABK1 |
| 20 | Сторнирующий проджект по блокировке | IFXPPABK2 |
| 30 | Проджект по движению | IFXPPAMK1 |
| Иное | Иное (источник не IFXPP) |  |

Если поле CQNR содержится в таблице пункта 5.14.1.1, определить RRN прочитанной записи файла RSACMTPD и проверить наличие соответствующей записи в файле с деталями операции (блокировки или движения). Для блокировок нужно дополнительно убедиться, что дата проджекта (поле ABLPD) совпадает с текущей.

Если запись нашлась, считаем, что источник — IFXPP. В противном случае обрабатываем как обычно.

Алгоритм заполнения деталей операции дорабатывается следующим образом:

Если источник проджекта — IFXPP, то в поле TOTRF необходимо указать референс объекта системы-инициатора (IFXPPABP.ABOBJREF или IFXPPAMP.AMOBJREF); в TSRCSYS следует указать наименование системы-инициатора (IFXPPABP.ABBLKSRC или IFXPPAMP.AMMVMSRC).

Если источник проджекта — не IFXPP, то в поле TSRCSYS следует указать константу 'MIDAS'.

## Изменения в настройках

### Настройки выгрузки DX'ов — IFXDXGSP

Для выгрузки формата EOD\_MVT добавляется новое поле TSRCSYS (в конец).

Для выгрузки формата INTRADAY\_MVT добавляется новое поле TSRCSYS (в конец).

### Настройки автоматической очистки файлов — IMBCLSTP

Необходимо автоматически очищать файл IFXPPLHP на основе поля LHTS с глубиной хранения данных 90 дней.

Необходимо автоматически очищать файл IFXPPLBP на основе поля LBTS с глубиной хранения данных 90 дней.

Необходимо автоматически очищать файл IFXPPLMP на основе поля LMTS с глубиной хранения данных 90 дней.

Необходимо автоматически очищать файл IFXPPABP на основе поля ABDLTD с глубиной хранения данных 90 дней.

Необходимо автоматически очищать файл IFXPPAMP на основе поля AMPSTD с глубиной хранения данных 90 дней.

Необходимо автоматически очищать файл IFXPPJRP на основе поля JRDATE с глубиной хранения данных 90 дней.

# Форматы сообщений

## Входящий — QueryAmountPreBlockCreate — создание неподтверждённых блокировок

Сообщение QueryAmountPreBlockCreate содержит список неподтверждённых блокировок, которые необходимо создать.

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryAmountPreBlockCreate». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Block |  | Да | Блокировка. Данная группа повторяется многократно. |
| Block.ID |  | Да | Уникальный идентификатор блокировки. Составной атрибут. |
| Block.ID.SystemCode | Char(32) | Да | Код системы-инициатора блокировки. |
| Block.ID.Reference | Char(32) | Да | Идентификатор блокировки во внешней системе. |
| Block.ObjectReference | Char(32) | Да | Референс объекта системы-инициатора, при работе с которым потребовалось создать блокировку. |
| Block.Account | Char(20) | Да | Счёт в формате ЦБ, на котором требуется заблокировать сумму. |
| Block.Amount | Char(23) | Да | Сумма блокировки в валюте счёта (в единицах валюты). |
| Block.StartDate | Char(8) | Нет | Дата начала действия блокировки (в формате YYYYMMDD). |
| Block.EndDate | Char(8) | Нет | Дата конца действия блокировки (в формате YYYYMMDD). |
| Block.UseOverdraft | Char(1) | Нет | Учитывать лимит овердрафта. 0 —нет; 1 — да. |
| Block.Comment | Char(300) | Да | Комментарий к блокировке. |
| Block.ParentID |  | Нет | Уникальный идентификатор родительской (основной) блокировки. Составной атрибут. |
| Block.ParentID.SystemCode | Char(32) | Нет | Код системы-инициатора блокировки. |
| Block.ParentID.Reference | Char(32) | Нет | Идентификатор блокировки во внешней системе. |

## Входящий — QueryAmountBlockCreate — создание подтверждённых блокировок

Сообщение QueryAmountBlockCreate содержит список подтверждённых блокировок, которые необходимо создать.

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryAmountBlockCreate». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Block |  | Да | Блокировка. Данная группа повторяется многократно. |
| Block.ID |  | Да | Уникальный идентификатор подтверждённой блокировки. Составной атрибут. |
| Block.ID.SystemCode | Char(32) | Да | Код системы-инициатора блокировки. |
| Block.ID.Reference | Char(32) | Да | Идентификатор блокировки во внешней системе. |
| Block.PreBlockID |  | Нет | Уникальный идентификатор неподтверждённой блокировки. Составной атрибут. |
| Block.PreBlockID.SystemCode | Char(32) | Да | Код системы-инициатора блокировки. |
| Block.PreBlockID.Reference | Char(32) | Да | Идентификатор блокировки во внешней системе. |
| Block.ObjectReference | Char(32) | Да | Референс объекта системы-инициатора, при работе с которым потребовалось создать блокировку. |
| Block.Account | Char(20) | Да | Счёт в формате ЦБ, на котором требуется заблокировать сумму. |
| Block.AmountType | Char(10) | Да | Тип блокировки.  Возможные значения:   * \*AMOUNT — блокировка на указанную сумму * \*EXCESS — блокировка на сумму всего остатка свыше указанной суммы * \*PERCENT — блокировка процента от суммы остатка |
| Block.Amount | Char(23) | Нет | Сумма блокировки в валюте счёта (в единицах валюты).  Поле должно быть заполнено в случае Block.AmountType in ('\*AMOUNT', '\*EXCESS'). |
| Block.Percent | Char(8) | Нет | Процент от суммы остатка.  Поле должно быть заполнено в случае Block.AmountType = '\*PERCENT'. |
| Block.StartDate | Char(8) | Нет | Дата начала действия блокировки (в формате YYYYMMDD). |
| Block.EndDate | Char(8) | Нет | Дата конца действия блокировки (в формате YYYYMMDD). |
| Block.ProjectTrnTyp | Char(2) | Да | TransactionType для проджекта. |
| Block.ProjectTrnNbr | Char(6) | Нет | TransactionNumber для проджекта. |
| Block.ProjectNarrative | Char(30) | Да | Комментарий к проджекту. |
| Block.UseOverdraft | Char(1) | Да | Учитывать лимит овердрафта. 0 — нет; 1 — да. |
| Block.IgnoreBlockFlag | Char(1) | Нет | Флаг обхода блокировки all/credit/debit. 0 — нет; 1 — да. |
| Block.IgnoreBalance | Char(1) | Нет | Флаг игнорирования доступного остатка. 0 — нет; 1 — да. |
| Block.Comment | Char(300) | Да | Комментарий к блокировке. |
| Block.ParentID |  | Нет | Уникальный идентификатор родительской (основной) блокировки. Составной атрибут. |
| Block.ParentID.SystemCode | Char(32) | Нет | Код системы-инициатора блокировки. |
| Block.ParentID.Reference | Char(32) | Нет | Идентификатор блокировки во внешней системе. |

## Входящий — QueryPreBlockPartialAuthorize — частичная авторизация неподтверждённых блокировок

Сообщение QueryPreBlockPartialAuthorize содержит список неподтверждённых блокировок, которые необходимо частично авторизовать.

Формат сообщения идентичен формату QueryAmountBlockCreate со следующими исключениями:

Поле MessageFormat должно содержать константу «QueryPreBlockPartialAuthorize».

Поле Block.PreBlockID обязательно для заполнения.

## Входящий — QueryAmountBlockCancel — удаление блокировок

Сообщение QueryAmountBlockCancel содержит список блокировок (подтверждённых или неподтверждённых), которые необходимо удалить.

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryAmountBlockCancel». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Block |  | Да | Блокировка. Данная группа повторяется многократно. |
| Block.ID |  | Да | Уникальный идентификатор блокировки. Составной атрибут. |
| Block.ID.SystemCode | Char(32) | Да | Код системы-инициатора блокировки. |
| Block.ID.Reference | Char(32) | Да | Идентификатор блокировки во внешней системе. |

## Входящий — QueryAmountBlockReverse — перевод подтверждённых блокировок в неподтверждённые

Сообщение QueryAmountBlockReverse содержит список подтверждённых блокировок, которые необходимо перевести в неподтверждённые.

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryAmountBlockReverse». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Block |  | Да | Блокировка. Данная группа повторяется многократно. |
| Block.ID |  | Да | Уникальный идентификатор блокировки. Составной атрибут. |
| Block.ID.SystemCode | Char(32) | Да | Код системы-инициатора блокировки. |
| Block.ID.Reference | Char(32) | Да | Идентификатор блокировки во внешней системе. |

## Входящий — QueryMovementCreate — cоздание движений

Сообщение QueryMovementCreate содержит список движений, которые необходимо создать.

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryMovementCreate». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Movement |  | Да | Движение. Данная группа повторяется многократно. |
| Movement.ID |  | Да | Уникальный идентификатор движения. Составной атрибут. |
| Movement.ID.SystemCode | Char(32) | Да | Код системы-инициатора движения. |
| Movement.ID.Reference | Char(32) | Да | Идентификатор движения во внешней системе. |
| Movement.BlockID |  | Нет | Идентификатор блокировки. Составной атрибут. |
| Movement.BlockID.SystemCode | Char(32) | Нет | Код системы-инициатора блокировки. |
| Movement.BlockID.Reference | Char(32) | Нет | Идентификатор блокировки во внешней системе. |
| Movement.ObjectReference | Char(32) | Да | Референс объекта системы-инициатора, при работе с которым потребовалось создать движение. |
| Movement.Account | Char(20) | Да | Счёт в формате ЦБ, по которому требуется совершить движение. |
| Movement.Amount | Char(23) | Да | Сумма движения в валюте счёта (в единицах валюты). |
| Movement.DebitCredit | Char(1) | Да | Списание ('D'), или зачисление ('C') . |
| Movement.SPOS | Char(7) | Да | Источник проводок. |
| Movement.OTRF | Char(15) | Да | Идентификатор проводки (Original transaction reference). |
| Movement.Department | Char(3) | Да | Департамент. |
| Movement.ProfitCenter | Char(4) | Нет | Центр прибыли. |
| Movement.BookCode | Char(2) | Нет | Book Code. |
| Movement.PostingTrnTyp | Char(5) | Нет | Тип транзакции. |
| Movement.PostingNarrative | Char(30) | Да | Английский комментарий к проводке. |
| Movement.ProjectTrnTyp | Char(2) | Да | Transaction Type для проджекта. |
| Movement.ProjectTrnNbr | Char(6) | Нет | Transaction Number для проджекта. |
| Movement.ProjectNarrative | Char(30) | Нет | Комментарий к проджекту. |
| Movement.EntryDate | Char(8) | Нет | Дата и время поступления требования к счёту[[21]](#footnote-22). |
| Movement.ValueDate | Char(8) | Нет | Дата валютирования. |
| Movement.Comment | Char(300) | Да | Русский комментарий к проводке. |
| Movement.ShortComment | Char(100) | Да | Короткий русский комментарий к проводке. |
| Movement.UseOverdraft | Char(1) | Нет[[22]](#footnote-23) | Учитывать лимит овердрафта. 0 — нет; 1 — да. |
| Movement.IgnoreBlockFlag | Char(1) | Нет | Флаг обхода блокировки all/credit/debit. 0 — нет; 1 — да. |
| Movement.IgnoreBalance | Char(1) | Нет | Флаг игнорирования доступного остатка. 0 — нет; 1 — да. |
| Movement.Storno | Char(1) | Нет | Движение является СТОРНО[[23]](#footnote-24). 0 — нет; 1 — да. |
| Movement.PrimeBlockID | Char(6) | Нет | Идентификатор блокировки средств на карте в системе Prime[[24]](#footnote-25). |

## Входящий — AEMovementCreate — cоздание движений

Сообщение AEMovementCreate содержит движение по дебету и движение по кредиту, которые необходимо создать, при необходимости дополнив необходимыми полупроводками с межбранчом и конверсией;

Формат сообщения

|  |  |  |  |
| --- | --- | --- | --- |
| Поле | Тип | Обяз. | Описание |
| MessageFormat | Char(32) | Да | Константа «QueryMovementCreate». Идентифицирует формат сообщения. |
| RequestID | Char(64) | Да | Уникальный идентификатор запроса. |
| Movement |  | Да | Движение. Данная группа повторяется многократно. |
| Movement.ID |  | Да | Уникальный идентификатор движения. Составной атрибут. |
| Movement.ID.SystemCode | Char(32) | Да | Код системы-инициатора движения. |
| Movement.ID.Reference | Char(32) | Да | Идентификатор движения во внешней системе. |
| Movement.BlockID |  | Нет | Идентификатор блокировки. Составной атрибут. |
| Movement.BlockID.SystemCode | Char(32) | Нет | Код системы-инициатора блокировки. |
| Movement.BlockID.Reference | Char(32) | Нет | Идентификатор блокировки во внешней системе. |
| Movement.ObjectReference | Char(32) | Да | Референс объекта системы-инициатора, при работе с которым потребовалось создать движение. |
| Movement.AdditionalReference | Char(32) | Нет | Референс текущего объекта, при работе с которым потребовалось создать движение: например. референс комиссии и .п. |
| Movement.Account | Char(20) | Да | Счёт в формате ЦБ, по которому требуется совершить движение. |
| Movement.Amount | Char(23) | Да | Сумма движения в валюте счёта (в единицах валюты). |
| Movement.DebitCredit | Char(1) | Да | Списание ('D'), или зачисление ('C') . |
| Movement.SPOS | Char(7) | Да | Источник проводок. |
| Movement.OTRF | Char(15) | Да | Идентификатор проводки (Original transaction reference). |
| Movement.Department | Char(3) | Да | Департамент. |
| Movement.ProfitCenter | Char(4) | Нет | Центр прибыли. |
| Movement.BookCode | Char(2) | Нет | Book Code. |
| Movement.PostingTrnTyp | Char(5) | Нет | Тип транзакции. |
| Movement.PostingNarrative | Char(30) | Да | Английский комментарий к проводке. |
| Movement.ProjectTrnTyp | Char(2) | Да | Transaction Type для проджекта. |
| Movement.ProjectTrnNbr | Char(6) | Нет | Transaction Number для проджекта. |
| Movement.ProjectNarrative | Char(30) | Нет | Комментарий к проджекту. |
| Movement.EntryDate | Char(8) | Нет | Дата и время поступления требования к счёту[[25]](#footnote-26). |
| Movement.ValueDate | Char(8) | Нет | Дата валютирования. |
| Movement.Comment | Char(300) | Да | Русский комментарий к проводке. |
| Movement.ShortComment | Char(100) | Да | Короткий русский комментарий к проводке. |
| Movement.UseOverdraft | Char(1) | Нет[[26]](#footnote-27) | Учитывать лимит овердрафта. 0 — нет; 1 — да. |
| Movement.IgnoreBlockFlag | Char(1) | Нет | Флаг обхода блокировки all/credit/debit. 0 — нет; 1 — да. |
| Movement.IgnoreBalance | Char(1) | Нет | Флаг игнорирования доступного остатка. 0 — нет; 1 — да. |
| Movement.Storno | Char(1) | Нет | Движение является СТОРНО[[27]](#footnote-28). 0 — нет; 1 — да. |
| Movement.PrimeBlockID | Char(6) | Нет | Идентификатор блокировки средств на карте в системе Prime[[28]](#footnote-29). |
| Movement.FAN | Char(1) | Нет | Флаг необходимости создать веерную проводку по движению; |

## Входящий — ControlCommand — внутренняя команда

Сообщение ControlCommand содержит команду самому адаптеру на изменение режима работы. Оно должно иметь тип\*DATAGRAM и не предполагает ответа. Отсылаются подобные сообщения программой IFXPPSTR (см. пункт 5.1).

Формат сообщения

|  |  |  |
| --- | --- | --- |
| Поле | Тип | Описание |
| MessageFormat | Char(32) | Константа «ControlCommand». Идентифицирует формат сообщения. |
| Action | Char(10) | Команда. Допустимы значения \*START и \*STOP. |

## Исходящий — StatusAmountBlock — результат выполнения запроса по работе с блокировками

Сообщение StatusAmountBlock выдаётся в ответ на запрос QueryAmountPreBlock\*\*\* и QueryAmountBlock\*\*\*.

Формат сообщения

|  |  |  |
| --- | --- | --- |
| Поле | Тип | Описание |
| MessageFormat | Char(32) | Константа «StatusAmountBlock». Идентифицирует формат сообщения. |
| Status | Char(1) | Статус обработки. Допустимые значения приведены в приложении I.I. |
| Message | Char(80) | Текст ошибки. |
| Block |  | Блокировка. Данная группа присутствует в стольких же экземплярах, сколько и в исходном запросе. |
| Block.ID |  | Уникальный идентификаторблокировки.Составной атрибут. |
| Block.ID.SystemCode | Char(32) | Код системы-инициатора блокировки. |
| Block.ID.Reference | Char(32) | Идентификатор блокировки во внешней системе. |
| Block.ErrorCode | Char(2) | Код ошибки. Допустимые значения приведены в приложении I.II. |
| Block.ErrorText | Char(80) | Текст ошибки. |
| Block.Amount | Char(23) | Величина заблокированной суммы. |
| Block.Type | Char(1) | Статус блокировки. A — подтверждённая; P — неподтверждённая. |
| Block.ABSReference | Char(32) | Референс блокировки в АБС. |

## Исходящий — StatusMovement — результат выполнения запроса по работе с движениями

Сообщение StatusMovement выдаётся в ответ на запрос QueryMovementCreate.

Формат сообщения

|  |  |  |
| --- | --- | --- |
| Поле | Тип | Описание |
| MessageFormat | Char(32) | Константа «StatusMovement». Идентифицирует формат сообщения. |
| Status | Char(1) | Статус обработки. Допустимые значения приведены в приложении I.I. |
| Message | Char(80) | Текст ошибки. |
| Movement |  | Движение. Данная группа присутствует в стольких же экземплярах, сколько и в исходном запросе. |
| Movement.ID |  | Уникальный идентификатор движения. Составной атрибут. |
| Movement.ID.SystemCode | Char(32) | Код системы-инициатора движения. |
| Movement.ID.Reference | Char(32) | Идентификатор движения во внешней системе. |
| Movement.ErrorCode | Char(2) | Код ошибки. Допустимые значения приведены в приложении I.II. |
| Movement.ErrorText | Char(80) | Текст ошибки. |
| Movement.ABSReference | Char(32) | Референс движения в АБС. |

## Исходящий — StatusUnknownFormat — ответ на неизвестный запрос

Сообщение StatusUnknownFormat выдаётся в случае, когда запрос имеет неподдерживаемый формат.

Формат сообщения

|  |  |  |
| --- | --- | --- |
| Поле | Тип | Описание |
| MessageFormat | Char(32) | Константа «StatusUnknownFormat». Идентифицирует формат сообщения. |
| Status | Char(1) | Статус обработки. Всегда равен '3' (глобальная ошибка). |
| Message | Char(80) | Текст ошибки. |

# Используемые MQ-очереди

Очередь запросов

Очередь запросов AEMovementCreate:

UCBRU.ADP.MIDAS.AEPOSTING.REQUEST

Очередь всех остальных запросов:

UCBRU.ADP.MIDAS.BLPR.REQUEST

Очереди ответов

Очередь ответов на запросы AEMovementCreate:

UCBRU.ADP.AEPOSTING.MDS.RESPONSE

Очередь всех остальных ответов:

UCBRU.AMDSCASABLAM.V4.PRBLCR.RESPONSE

1. Табличные данные
   1. Глобальные статусы

|  |  |
| --- | --- |
| Статус | Описание |
| 0 | Обработка прошла успешно |
| 1 | Дублирование запроса |
| 2 | Ошибка обработки одной из блокировок / движений |
| 3 | Глобальная ошибка |
| 4 | В АБС идёт COB |

* 1. Статусы обработки блокировок / движений

|  |  |
| --- | --- |
| Код ошибки | Текст ошибки |
| 00 | Всё корректно, ошибок нет |
| 01 | Группа Block.ID не заполнена |
| 02 | Блокировка с идентификатором Block.ID уже присутствует в системе |
| 03 | Блокировка с идентификатором Block.ID отсутствует в системе |
| 03 | Подтверждённая блокировка с идентификатором Block.ID отсутствует в системе |
| 03 | Блокировка с идентификатором Block.ID уже удалена |
| 04 | Поле Block.ObjectReference не заполнено |
| 05 | Поле Block.Account не заполнено |
| 06 | Счёт Block.Account не найден в MIDAS |
| 07 | Счёт Block.Account закрыт в MIDAS |
| 08 | Поле Block.Amount не заполнено |
| 09 | Поле Block.Amount содержит недопустимые символы |
| 09 | Поле Block.Amount имеет некорректный формат |
| 10 | Сумма блокировки не может равняться нулю |
| 10 | Сумма блокировки не может быть отрицательной |
| 11 | Поле Block.StartDate имеет некорректный формат |
| 11 | Блокировка не может начинаться ранее 1972-го года |
| 12 | Поле Block.EndDate имеет некорректный формат |
| 12 | Дата окончания блокировки не должна превышать 2245-10-14 |
| 13 | Дата окончания блокировки должна быть больше или равна дате начала |
| 14 | Поле Block.UseOverdraft содержит некорректное значение |
| 15 | Поле Block.IgnoreBlockFlag содержит некорректное значение |
| 16 | Поле Block.IgnoreBalance содержит некорректное значение |
| 17 | Поле Block.ProjectTrnTyp не заполнено |
| 18 | Группа Block.ParentID содержит некорректное значение |
| 19 | Поле Block.ProjectNarrative не заполнено |
| 20 | Поле ObjectReference разное в неподтверждённой блокировке и в запросе |
| 20 | Поле Account разное в неподтверждённой блокировке и в запросе |
| 20 | Поле Amount разное в неподтверждённой блокировке и в запросе |
| 20 | Поле StartDate разное в неподтверждённой блокировке и в запросе |
| 20 | Поле EndDate разное в неподтверждённой блокировке и в запросе |
| 20 | Поле UseOverdraft разное в неподтверждённой блокировке и в запросе |
| 20 | Поле Comment разное в неподтверждённой блокировке и в запросе |
| 20 | Группа ParentID разная в неподтверждённой блокировке и в запросе |
| 21 | Счёт Block.Account должен иметь тип R (Retail) |
| 22 | На счёте недостаточно денег для осуществления операции |
| 25 | Ошибка IFXPPMPR: <текст ошибки> |
| 26 | Ошибка MMBGLBLR: <текст ошибки> |
| 31 | Поле Block.AmountType содержит некорректное значение |
| 32 | Поле Block.Percent содержит некорректное значение |
| 33 | Группа Block.PreBlockID не заполнена |
| 34 | Группа Block.PreBlockID совпадает с Block.ID |
| 34 | Группа Block.PreBlockID не совпадает с Block.ID |
| 35 | Не найдена неподтверждённая блокировка с идентификатором Block.PreBlockID |
| 35 | Блокировка с идентификатором Block.PreBlockID уже удалена |
| 36 | Для блокировки по активному счёту необходимо явное указание суммы |
| 37 | Сумма частичной авторизации превышает заблокированную сумму |
| 38 | Дата начала действия блокировки не может быть больше текущего дня |
| 51 | Группа Movement.ID не заполнена |
| 52 | Группа Movement.BlockID содержит некорректное значение |
| 53 | Движение с идентификатором Movement.ID уже присутствует в системе |
| 54 | Поле Movement.ObjectReference не заполнено |
| 55 | Поле Movement.Account не заполнено |
| 56 | Счёт Movement.Account не найден в MIDAS |
| 57 | Счёт Movement.Account закрыт в MIDAS |
| 58 | Поле Movement.Amount не заполнено |
| 59 | Поле Movement.Amount содержит недопустимые символы |
| 59 | Поле Movement.Amount имеет некорректный формат |
| 60 | Сумма движения не может равняться нулю |
| 60 | Сумма движения не может быть отрицательной |
| 61 | Поле Movement.DebitCredit не заполнено |
| 61 | Поле Movement.DebitCredit содержит некорректное значение |
| 62 | Кредитовое движение не может быть создано на основе блокировки |
| 63 | Поле Movement.SPOS не заполнено |
| 64 | Поле Movement.OTRF не заполнено |
| 65 | Поле Movement.Department не заполнено |
| 65 | Поле Movement.Department содержит некорректное значение |
| 66 | Поле Movement.ProfitCenter содержит некорректное значение |
| 67 | Поле Movement.BookCode содержит некорректное значение |
| 68 | Поле Movement.PostingTrnTyp содержит некорректное значение |
| 69 | Поле Movement.PostingNarrative не заполнено |
| 70 | Поле Movement.ProjectTrnTyp не заполнено |
| 71 | Поле Movement.EntryDate имеет некорректный формат |
| 72 | Поле Movement.ValueDate имеет некорректный формат |
| 72 | Дата валютирования слишком далеко в прошлом |
| 72 | Дата валютирования слишком далеко в будущем |
| 73 | Поле Movement.Comment не заполнено |
| 74 | Поле Movement.ShortComment не заполнено |
| 75 | Поле Movement.UseOverdraft содержит некорректное значение |
| 76 | Поле Movement.IgnoreBlockFlag содержит некорректное значение |
| 77 | Поле Movement.IgnoreBalance содержит некорректное значение |
| 78 | Поле Movement.Storno содержит некорректное значение |
| 79 | По счёту запрещены дебетовые движения |
| 79 | По счёту запрещены кредитовые движения |
| 79 | По счёту запрещены все движения |
| 80 | Поле ObjectReference разное в неподтверждённой блокировке и в запросе |
| 80 | Поле Account разное в неподтверждённой блокировке и в запросе |
| 80 | Поле Amount разное в неподтверждённой блокировке и в запросе |
| 80 | Поле UseOverdraft разное в неподтверждённой блокировке и в запросе |
| 80 | Поле Comment разное в неподтверждённой блокировке и в запросе |
| 81 | Поле ObjectReference разное в подтверждённой блокировке и в запросе |
| 81 | Поле Account разное в подтверждённой блокировке и в запросе |
| 81 | Поле Amount разное в подтверждённой блокировке и в запросе |
| 81 | Поле UseOverdraft разное в подтверждённой блокировке и в запросе |
| 81 | Поле IgnoreBlockFlag разное в подтверждённой блокировке и в запросе |
| 81 | Поле IgnoreBalance разное в подтверждённой блокировке и в запросе |
| 81 | Поле Comment разное в подтверждённой блокировке и в запросе |
| 82 | На счёте недостаточно денег для осуществления операции |
| 83 | Счёт Movement.Account должен иметь тип R (Retail) |
| 85 | Ошибка IFXPPMPR: <текст ошибки> |
| 86 | Ошибка MMBGLGPI2: <текст ошибки> |
| 87 | Ошибка MMBGLSU0: <текст ошибки> |
| 88 | Ошибка MMBGLAPR: <текст ошибки> |
| 99 | Прочие программные ошибки |

* 1. Соответствие форматов запросов и ответов

|  |  |
| --- | --- |
| Формат запроса | Формат ответа |
| QueryAmountPreBlockCreate | StatusAmountBlock |
| QueryAmountBlockCreate | StatusAmountBlock |
| QueryAmountBlockCancel | StatusAmountBlock |
| QueryAmountBlockReverse | StatusAmountBlock |
| QueryMovementCreate | StatusMovement |
| AEMovementCreate | StatusMovement |
| ControlCommand | Ответ не отсылается |
| Всё остальное | StatusUnknownFormat |

* 1. Возможные значения поля IFXPPABP.ABTYPE

|  |  |
| --- | --- |
| Значение | Описание |
| P | Неподтверждённая блокировка |
| A | Подтверждённая блокировка |

* 1. Возможные значения поля IFXPPABP.ABSTS

|  |  |
| --- | --- |
| Значение | Описание |
| A | Блокировка активна |
| D | Блокировка удалена |
| M | На основе блокировки введено движение |

* 1. Сообщения об ошибках при создании проджекта

|  |  |
| --- | --- |
| DBASE | Текст ошибки |
| 1 | GLI01: Switchable feature verification failure |
| 2 | GLI02: Bank details access failure |
| 3 | GLI03: Module details access failure |
| 4 | GLI04: Retail ICD access failure |
| 5 | GLI05: Account not found or closed |
| 6 | GLI06: MEMOS file record access failure |
| 7 | GLI07: PRONO file record access failure |
| 8 | GLI08: PRONO file record access failure or NRF and NOSTRO not set up |
| 9 | GLI09: Value date is equal to zero |
| 10 | GLI10: Value date is greater than run date for non-retail acc-s - DISABLED |
| 11 | GLI11: Value date is greater than system maximum forward date |
| 12 | GLI12: Value date is less than account opening date |
| 13 | GLI13: Value date is not a working date |
| 14 | GLI14: Value date is before back value date |
| Прочее | GLI15: Unknown error |

1. Максимальное допустимое число записей в одном запросе ограничено только возможностями MessageAdapter — 4 МБ на одно сообщение. [↑](#footnote-ref-1)
2. Ответ собирается по логам IFXPPLHP, IFXPPLBP/IFXPPLMP. [↑](#footnote-ref-2)
3. Проверяется по таблице IFXPPABP. [↑](#footnote-ref-3)
4. Таблица привязок ЦБ-счетов к счетам MIDAS — IMBCBADP, к субсчетам — IMBCBHCP. [↑](#footnote-ref-4)
5. Тот факт, что счёт MIDAS открыт, выражается условием ACCNTAB.DACC = 0. [↑](#footnote-ref-5)
6. Это можно определить по счёту MIDAS по признаку ACCNTAB.ATYP = 'R'. [↑](#footnote-ref-6)
7. Таким образом для валюты RUR допустимы суммы вида 100, 100.5, 100.55, 100.00, 100.500, но не 100.501. [↑](#footnote-ref-7)
8. Здесь и далее: под корректными датами подразумеваются даты в формате YYYYMMDD, лежащие в диапазоне от 1972-01-01 до 2245-10-14 (максимальная допустимая дата в формате MIDAS). [↑](#footnote-ref-8)
9. Проверяется по таблице IFXPPAML. [↑](#footnote-ref-9)
10. Допустимость проверяется по справочнику SDDEPTPD. [↑](#footnote-ref-11)
11. Допустимость проверяется по справочнику SDPRFCPD. [↑](#footnote-ref-12)
12. Допустимость проверяется по справочнику SDBOOKPD. [↑](#footnote-ref-13)
13. Допустимость проверяется по справочнику SDRETRPD с учётом признака «дебет/кредит». [↑](#footnote-ref-14)
14. Максимальное допустимое удаление даты валютирование в будущее хранится в поле SDRETLPD.BMNDFV; в прошлое — в поле SDBANKPD.BJBVPD. [↑](#footnote-ref-15)
15. Максимальная допустимая дата MIDAS. [↑](#footnote-ref-16)
16. Проджект вводится в последнюю очередь, чтобы минимизировать риск появления "лишних" операций (по которым был rollback) в intraday-выписке. [↑](#footnote-ref-17)
17. Здесь и в следующем пункте: остаток по счёту рассчитывается так же, как и в пункте 5.2.5. [↑](#footnote-ref-18)
18. Это можно сделать по счёту ЦБ второго порядка по таблице IMBCBAPP. [↑](#footnote-ref-19)
19. Следует иметь в виду, что лимит овердрафта устанавливается не в "копейках", а в целых единицах валюты. [↑](#footnote-ref-20)
20. Поскольку у IFXPPAMP нет ключа по данному полю, предполагается пользоваться логическим IFXPPAML, c ключом по AMMVMSRC, AMMVMID, значения ключа вычитывать по AMLHID из IFXPPLMP. [↑](#footnote-ref-21)
21. Это поле является информационным и никак не используется. [↑](#footnote-ref-22)
22. Обязательно для дебетовых движений [↑](#footnote-ref-23)
23. Это поле является информационным и никак не используется. [↑](#footnote-ref-24)
24. Это поле является информационным и никак не используется. [↑](#footnote-ref-25)
25. Это поле является информационным и никак не используется. [↑](#footnote-ref-26)
26. Обязательно для дебетовых движений [↑](#footnote-ref-27)
27. Это поле является информационным и никак не используется. [↑](#footnote-ref-28)
28. Это поле является информационным и никак не используется. [↑](#footnote-ref-29)