



Installation and Configuration Manual

Контроллер банкоматов (R2)

03.51.30

13.05.2020

СОДЕРЖАНИЕ

1.	Перечни контроллера банкоматов	5
1.1	Настраиваемые перечни контроллера банкоматов	5
1.1.1	Перечень типов банкоматов (ATM Types)	5
1.1.2	Перечень деноминаций (ATM Denominations)	7
1.2	Фиксированные перечни	9
1.2.1	Перечень протоколов банкоматов (ATM Protocols)	9
1.2.2	Перечень операций банкоматов (ATM Operations)	10
1.2.3	Перечень компонентов банкоматов (ATM Hardware Types)	11
1.2.4	Перечень сообщений банкоматов (ATM Message Types)	12
2.	Описание нового банкомата и его конфигурирование	14
2.1	Настройка контракта банкомата и связанного с ним устройства	14
2.1.1	Настройка допустимых операций банкомата	14
2.1.2	Настройка компонентов банкомата (Hardware)	15
2.1.3	Управление состоянием банкомата	17
2.1.4	Специфицирование ключей шифрования	17
2.1.5	Включение режима MAC-подписи	19
2.1.6	Дополнительные параметры устройства	19
2.2	Настройка соединения банкомата с хостом WAY4	20
2.3	Подготовка конфигураций банкоматов и их загрузка	21
2.3.1	Подготовка файлов конфигурации	21
2.3.2	Посылка конфигурации одному банкомату	24
2.3.3	Посылка конфигурации всем банкоматам	24
3.	Конфигурационные файлы контроллера банкоматов	25
3.1	Конфигурационный файл контроллера	25
3.2	Файлы настройки взаимодействия контроллера с банкоматами	27
3.2.1	Принципы работы с конфигурационными скриптами	27
3.2.2	Файл конфигурации обработки запросов	29
3.2.3	Файл конфигурации обработки ответов	32
3.3	Настройка шаблонов чеков и экранов	35
3.3.1	Настройка перекодировки	44

3.3.2	Тестирование шаблонов	54
4.	Схема расчетов по операциям банкомата	61
5.	Автоматическое формирование контроллером сообщения об отмене операции (Reversal)	65

Контроллер банкоматов представляет собой программный компонент, функционирующий на платформе WAY4 Transaction Switch.

Назначение контроллера банкоматов состоит в обеспечении взаимодействия сети банкоматов с процессинговым центром. Данное взаимодействие включает передачу команд управления банкоматам, прием сообщений от банкоматов, передачу кодов ответов и т. д.

Настоящий документ предназначен для системных администраторов WAY4 (сотрудников банков или процессинговых центров), обеспечивающих работу по настройке сети банкоматов.

При работе с данным документом рекомендуется пользоваться следующими источниками из комплекта документации OpenWay:

- "Работа с DB Manager";
- "WAY4™ Transaction Switch. Platform Overview";
- "Ежедневные процедуры";
- "Модуль эквайринга. Руководство пользователя";
- "Модуль эмиссии. Руководство пользователя";
- "Мониторинг сети банкоматов";
- "Пакеты Сервисов системы WAY4™";
- "Настройка сообщений клиентам".

В документе используются следующие обозначения:

- названия полей экранных форм выделяются *курсивом*;
- названия кнопок экранных форм приводятся в квадратных скобках, например, [Approve];
- последовательность выбора пункта в меню пользователя отображается с помощью стрелок следующим образом: "Issuing → Contracts Input & Update";
- последовательность выбора пункта в системном меню отображается с помощью стрелок следующим образом: "Database => Change password";
- комбинации клавиш, используемые при работе с DB Manager, приводятся в угловых скобках, например, <Ctrl>+<F3>;



предостережения в связи с возможностью совершения неправильных действий отмечены специальной пиктограммой и выделены цветом фона;



информация о важных особенностях, дополнительных возможностях или оптимальном использовании некоторых функций системы также отмечена специальной пиктограммой и выделена цветом фона.

1. Перечни контроллера банкоматов

Важным источником информации, используемым при функционировании контроллера банкоматов, являются так называемые перечни (Dictionaries). Перечни представляют собой таблицы, хранящиеся в базе данных системы WAY4 и содержащие однотипную информацию, например, перечень типов банкоматов, перечень сообщений банкоматов и т. д.

В системе используются два типа перечней:

- настраиваемые – перечни, содержимое которых может изменяться пользователем;
- фиксированные – перечни, содержимое которых может изменяться только представителями поставщика системы WAY4; в некоторых случаях изменение данных перечней может осуществляться ответственными специалистами банков или процессинговых центров под контролем представителей поставщика системы WAY4.

1.1 Настраиваемые перечни контроллера банкоматов

1.1.1 Перечень типов банкоматов (ATM Types)

Все типы банкоматов, с которыми взаимодействует система WAY4, должны быть зарегистрированы в специальном перечне типов банкоматов.

Задание типа банкомата осуществляется при конфигурировании устройств (см. раздел "Информация об устройствах" документа "Модуль эквайринга. Руководство пользователя").

Доступ к перечню типов банкоматов осуществляется путем выбора в меню пользователя пункта "Full → Configuration Setup → Merchant Device Setup → ATM Types".

ATM Types									
Code	Name	Brand	Model	Conf File	Outgoing Console	Cash Account Code	Cash Account Code	EMV Conf File	
NCR5070A	NCR 5070 Ange	NCR3G-ASD	5070	ndc3g.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5084	NCR 5084	NCR3G	5084	ndc3g.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5084A	NCR 5084 Ange	NCR3G-ASD	5084	ndc3g.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5305A	NCR 5305 Ange	NCR4G-ASD	5305	ndc.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5670	NCR 5670	NCR	5670	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5674	NCR 5674	NCR	5674	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5684	NCR 5684	NCR	5684	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		
NCR5840	NCR 5840 (Pers)	NCR	5840	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		MPD:1010010000
NCR5870	NCR 5870 (Pers)	NCR	5870	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		MPD:1010010000
NCR5874	NCR 5874 (Pers)	NCR	5874	ndcplus.cfg	Outgoing Console	ATM Cassette	Cash Dispenser		MPD:1010010000

Таблица типов банкоматов, зарегистрированных в системе

Данная таблица содержит следующие поля:

- *Code* – код, идентифицирующий тип банкомата;

- *Name* – описатель типа банкомата;
- *Brand* – условный бренд банкомата, который в сочетании с протоколом определяет диалект, используемый контроллером:
 - Aprta NDC – бренды: NCR, NCR4G, NCR5G, NCR4G-ASD, NCR5G-ASD, PERSONAS, GRG, WINCOR-APTRA, NAUTILUS-APTRA, DIEBOLD;
 - ProCash NDC – бренды: WINCOR, BANQIT;
 - ProCash MDS912 – бренды: WINCOR, BANQIT, GRG, NAUTILUS;
 - Diebold MDS912 – бренды: DIEBOLD, IBM;
- *Model* – модель банкомата;
- *Configuration File* – имя конфигурационного файла с указанием пути доступа к нему относительного каталога WEB-INF/inv/atm/conf (каталог inv/atm/conf необходимо самостоятельно создать в корневом каталоге WAY4 Transaction Switch);
- *Outgoing Console* – компонент банкомата, с помощью которого осуществляется управление банкоматом;
- *Cassette Account Code*, *Cash Account Code*, *Remaining Account Code* – поля определяют наименования типов счетов из Схемы Счетов контракта банкомата, отражающих движение средств при инкассации банкомата (см. "[Схема расчетов по операциям банкомата](#)");
- *EMV Conf File* – параметры устройства; задаются списком "тег:значение" через точку с запятой; например, с помощью тегов MPD, VPD могут задаваться атрибуты исходящего сообщения (тип устройства, способ чтения карты и т. д.), формируемого интерфейсными сервисами WAY4 Transaction Switch для отправки системам MasterCard или VISA соответственно;
- *Transaction Attributes* – дополнительные параметры транзакции;
- *Protocol Id* – наименование протокола (см. "[Перечень протоколов банкоматов \(ATM Protocols\)](#)").

При работе с данной табличной формой пользователь имеет возможность добавлять записи в таблицу, используя кнопку [Ins], а также удалять записи, используя кнопку [Del].

При попытке удалить из таблицы "ATM Types" запись, соответствующую типу банкомата, для которого в системе зарегистрирован контракт устройства (см. раздел "Ввод новых контрактов устройств" документа "Модуль эквайринга. Руководство пользователя"), на экране будет представлено соответствующее предупреждение.

Кнопка [Dflt Oper] позволяет задать список операций, которые по умолчанию будут доступны для выполнения на выбранном типе банкомата. Список формируется на основе перечня "ATM Operations" (см. "[Перечень операций банкоматов \(ATM Operations\)](#)"). Кнопка [Dflt Hardware] позволяет задать список типов компонентов, которые по умолчанию будут использоваться на выбранном типе банкомата. Для выбора доступны компоненты из перечня "ATM Hardware Types" (см. "[Перечень компонентов банкоматов \(ATM Hardware Types\)](#)").

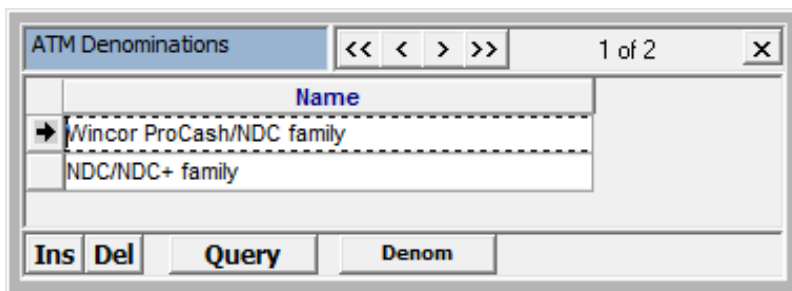
1.1.2 Перечень деноминаций (ATM Denominations)

Каждому банкомату в системе ставится в соответствие тот или иной тип деноминации, определяющий типы кассет, которые могут использоваться банкоматом, номинал и валюту банкнот, загружаемых в кассеты при инкассации, а также некоторые параметры выдачи банкнот. Правила выдачи банкнот могут быть определены значением параметра `ctx.dispenseType` (см. "Файл конфигурации обработки запросов"):

- MINIMAL_NOTES – выдача с минимизацией по количеству банкнот (по возможности выбираются банкноты наибольшего номинала);
- MINIMAL_VALUES – выдача с минимизацией по номиналу банкнот (по возможности выбираются банкноты наименьшего номинала);
- PENALTY_BASED – выдача в соответствии с заданным алгоритмом комплектования (см. ниже описание поля Dispense Parameters).

Задание типа деноминации для банкомата осуществляется путем выбора из списка при конфигурировании устройств (см. раздел "Конфигурирование устройств" документа "Модуль эквайринга. Руководство пользователя").

Доступ к перечню деноминаций осуществляется путем выбора в меню пользователя пункта "Full → Configuration Setup → Merchant Device Setup → ATM Denominations".



Справочник деноминаций

При работе с данной табличной формой пользователь имеет возможность добавлять записи в таблицу, используя кнопку [Ins], а также удалять записи, используя кнопку [Del].

При попытке удалить из таблицы "ATM Denominations" запись, соответствующую типу деноминации, используемому для зарегистрированного в системе банкомата (см. раздел "Ввод новых контрактов устройств" документа "Модуль эквайринга. Руководство пользователя"), на экране будет представлено соответствующее предупреждение.

Каждому типу деноминации соответствуют определенные параметры загрузки кассет банкнотами и параметры алгоритма выдачи банкнот. Для доступа к таблице, содержащей указанные параметры, следует, выбрав в таблице "ATM Denominations" требуемый тип деноминации, нажать на кнопку [Denom]. По этой команде на экране будет представлена табличная форма "Denom for <тип деноминации>".

Denom for NDC/NDC+ family							<< < > >>		1 of 4		b x	
	Direction	Item Type	Name	Mapped to	Currency	Denomination	Limit	Dispense Parameters	Cycle Type Code			
→	Debit	Banknote	1		EUR	500,00	40					
	Debit	Banknote	2		EUR	10,00	40					
	Debit	Banknote	3		EUR	100,00	40					
	Debit	Banknote	4		EUR	50,00	40					
Ins Del Query												

Параметры загрузки кассет банкнотами

Данная таблица содержит следующие поля:

- *Direction* – назначение кассеты по направлению движения средств ("Debit" – выдача, "Credit" – прием);
- *ItemType* – тип кассеты ("Banknote" – банкноты, "Coin" – монеты (в текущей версии работа с монетами не поддерживается));
- *Name* – наименование кассеты; в зависимости от типа банкомата используются цифры "1", "2", "3", "4", "5", "6", "7";
- *Mapped to* – ссылка на наименование кассеты для приема (например, отбракованных купюр из текущей кассеты);
- *Currency* – поле для выбора из списка наименования валюты денежных знаков, загруженных в кассету;
- *Denom* – номинал банкнот, загруженных в кассету;
- *Limit* – максимальное количество банкнот (от 0 до 99), выдаваемых из кассеты за одну операцию;
- *Dispense Parameters* – правило комплектования набора банкнот для выдачи; задается с помощью тегов "PENALTY" и "PENALTY_INCR" (разделяются запятой), например:

PENALTY=1/4/3,PENALTY_INCR=0/3/1

где:

- PENALTY – штраф за выдачу банкноты из кассеты; 1/4/3 – значения штрафа для первого, второго и третьего алгоритмов комплектования набора банкнот соответственно;
- PENALTY_INCR – увеличение штрафа за каждую следующую выданную банкноту из кассеты; 0/3/1 – значение увеличения штрафа для первого, второго и третьего алгоритмов комплектования набора банкнот соответственно;

В качестве значений штрафа могут быть указаны целочисленные значения в интервале от 0 до 99 (включительно). При этом с точки зрения эффективности рекомендуется использовать значения в интервале от 0 до 30.



Алгоритм комплектования банкнот в зависимости от пункта меню, выбранного на банкомате, определяется в конфигурации контроллера (см. "Файл конфигурации обработки запросов") с помощью параметра `ctx.dispenseAlgorithmId` (значение по умолчанию - 1 (используется первый алгоритм)).

Выбор банкнот из кассет банкомата при выдаче наличных денег производится так, чтобы минимизировать условный штраф за выдачу некоторого набора банкнот с учетом лимитов.

Значение условного штрафа P вычисляется по следующей формуле:

$$P = \sum_{i=1}^N (n_i \cdot CST_FEE_i + \frac{n_i \cdot (n_i - 1)}{2} \cdot CST_FEE_INCREASE_i),$$

где:

- N – количество кассет, из которых осуществляется выбор банкнот;
- n_i – количество банкнот, выбираемых из i -той кассеты;
- CST_FEE_i – штраф за выдачу банкноты из i -той кассеты (значение тега PENALTY для выбранного алгоритма);
- $CST_FEE_INCREASE_i$ – увеличение штрафа за каждую следующую выданную банкноту из i -той кассеты (значение тега PENALTY_INCR для выбранного алгоритма).

1.2 Фиксированные перечни

1.2.1 Перечень протоколов банкоматов (ATM Protocols)

Протоколы банкоматов регламентируют формат сообщений и правила передачи информации между банкоматом и процессинговым центром.

Перечень протоколов банкоматов содержится в таблице "ATM Protocols".

ATM Protocols			
<< < > >>		1 of 2	X
	Name	Code	Is Adjusting Totals
→	NDC+	NDC+	
<div> <div>Ins</div> <div>Del</div> <div>Query</div> </div>			

Типы протоколов для связи банкоматов с процессинговым центром

Данная таблица содержит следующие поля:

- *Name* – наименование протокола;
- *Code* – код протокола, уникальный в рамках системы.



Текущая реализация контроллера банкоматов на платформе WAY4 Transaction Switch поддерживает протокол "NDC+" и "MDS912".

1.2.2 Перечень операций банкоматов (ATM Operations)

Каждому контракту банкомата в системе ставится в соответствие совокупность операций, осуществляемых с помощью данного устройства, а также некоторый набор компонентов (hardware) (см. "[Перечень компонентов банкоматов \(ATM Hardware Types\)](#)"), необходимых для выполнения этих операций.

Перечень возможных операций банкоматов содержится в таблице "ATM Operations".

ATM Operations											
Name	Code	Transaction Type	Trans Cond	Request	Category	Service Class	Is Checked	Downgradable To	Requires Cassettes	Is Online	Use Coins
Bill Payments by Cash	T-Y1	Pay by Cash		Advice	Authorisation				No	No	No
CE: Conversion	T-X1_D	CE: 5.Conversion		Advice	Authorisation	Transaction			No	No	No
CE: Rounding	T-X1_C	CE: 4.Rounding		Advice	Authorisation	Transaction			No	No	No
CE: Cash Dispense (Chg)	T-X1_B	CE: 3.Cash Dispense		Advice	Authorisation	Transaction			Yes	No	Yes
CE: Cash Dispense	T-X1_A	CE: 2.Cash Dispense		Advice	Authorisation	Transaction			Yes	No	Yes
CE: Currency Exchange	T-X1	CE: Currency Exchar		Advice	Authorisation	Transaction			No	No	No
PWC: Rounding	T-W1_C	PWC: 3.Rounding		Advice	Authorisation	Transaction			No	No	No
PWC: Cash Dispense (C)	T-W1_B	PWC: 2.Cash Dispens		Advice	Authorisation	Transaction			Yes	No	Yes

Список операций банкоматов

Данная таблица содержит следующие поля:

- *Name* – наименование операции;
- *Code* – код операции;
- *Transaction Type* – тип транзакции;
- *Trans Cond* – условия выполнения транзакции;
- *Request* – категория документа, формируемого по сообщению об операции (запрос/уведомление);
- *Category* – категория финансовых/авторизационных сообщений;
- *Service Class* – банковская классификация транзакций; значение данного параметра определяет способ обработки документа в системе; отсутствие значения в данном поле означает по умолчанию значение параметра *Service Class* = "Transaction";
- *Is Checked* – поле с выбором из списка значений для указания необходимости контроля сервисной карты (значение "Yes") при выполнении сервисных операций (Replenishment, End of Day, ATM Service и др.);
- *Downgradable To* – поле для указания наименования операции, которая может быть использована в случае невозможности выполнить данную операцию;
- *Requires Cassettes* – флаг, указывающий на использование кассет при выполнении операции;
- *Is Online* – признак выполнения операции в режиме онлайн (осуществляется ли при выполнении операции запрос к эмитенту);
- *Use Coins* – флаг, указывающий на возможность использования монет при выполнении операции (в текущей версии данная функциональность не поддерживается);
- *Special Parameters* – дополнительные параметры операции; задаются списком "тег=значение" через точку с запятой.

Список операций банкоматов приводится в разделе "undefined" документа "Мониторинг сети банкоматов".

Для вывода на экран табличной формы "Required for <наименование операции>", содержащей список компонентов банкомата (см. "[Перечень компонентов банкоматов \(ATM Hardware Types\)](#)"), необходимых для выполнения соответствующей операции, используется кнопка [Required].

В результате нажатия на кнопку [Fill Default] таблица "ATM Operations" заполняется по умолчанию всеми возможными значениями.

1.2.3 Перечень компонентов банкоматов (ATM Hardware Types)

Каждому контракту банкомата в системе ставится в соответствие некоторый набор компонентов (hardware), необходимых для выполнения банкоматом требуемых операций (см. "[Перечень операций банкоматов \(ATM Operations\)](#)").

Перечень возможных компонентов банкоматов содержится в таблице "ATM Hardware Types".

ATM Hardware Types			
<div> <div><< < > >></div> <div>1 of 25</div> <div>✕</div> </div>			
Name	Code	Protocol	
Withdrawal Area	Wd	Diebold 912	
Alarm	Al		
ATM Systems	Pw		
Outgoing Console	Co		
Incoming Console	Ci		
Dispenser	Dp		
Security Devices	Se		
Vandal Shield	VS	Diebold 912	
Electronic Data Capture	EDC	Diebold 912	
Authorisation Channels	Auth		
Supervisor	SuperV		
Statement Printer	Ps		
Journal Printer	Pj		
Consumer Printer	Pc		
Card Reader	Cr		
Depository	Dps		
Night Safe	Ns	NDC+	
<div> <div>Ins</div> <div>Del</div> <div>Query</div> <div>Messages</div> </div>			

Таблица компонентов банкоматов

Данная таблица содержит следующие поля:

- *Name* – наименование компонента;
- *Code* – код компонента в системе;
- *Protocol* – тип протокола, допускающего использование соответствующего компонента (данное поле не заполняется при использовании компонента, совместимого со всеми зарегистрированными типами протоколов).

Кнопка [Messages] предназначена для вывода на экран табличной формы "Messages for <наименование компонента>", содержащей список сообщений, формируемых в системе при работе данного компонента (см. "[Перечень сообщений банкоматов \(ATM Message Types\)](#)").

Список компонентов банкоматов приводится в разделе "Компоненты банкомата (Hardware)" документа "Мониторинг сети банкоматов".

1.2.4 Перечень сообщений банкоматов (ATM Message Types)

При работе банкомата в системе могут формироваться определенные статусные сообщения. Хост WAY4 может отправлять контроллеру, банкомату или группе банкоматов управляющие сообщения (команды). Например, через консоль управления (см. раздел "Изменение статуса банкомата и управление банкоматом" документа "Мониторинг сети банкоматов") возможна установка конфигурации банкомата или загрузка конфигурационных файлов контроллера.

Банкоматы, в свою очередь, присылают сообщения о результатах выполнения команд хоста, а также сообщения с информацией о состоянии своих устройств. В результате в БД WAY4 фиксируются несколько сообщений для соответствующего банкомата, каждое из которых относится к его отдельному компоненту.

Перечень возможных сообщений содержится в таблице "ATM Message Types".

ATM Message Types								
Message Scope	Protocol	Device Type	Hardware Type	Name	Code	Error Level	Group Code	Security
Device	Diebold 912		Alarm	Terminal is not in a transaction request state	2D41161	Information		100
Device	Diebold 912		Security Devices	Terminal Master Key Expired	TMKEXPIRED	Information	#S341	100
Device	NDC+		Security Devices	Terminal Master Key Expired	TMKEXPIRED	Information	#S340	100
Device	Diebold 912		Security Devices	Terminal PIN Key Expired	TPKEXPIRED	Information	#S341	100
Device	NDC+		Security Devices	Terminal PIN Key Expired	TPKEXPIRED	Information	#S340	100
Device	Diebold 912		EMV Smart Card	Terminal Risk Management failed.	SV77770401	Warning		100
Device	Diebold 912		ATM Systems	Terminal status is error	TERMS3	Information		100
Device	NDC+		ATM Systems	Terminal status is error	TERMS3	Information		100
Device	Diebold 912		ATM Systems	Terminal status is fatal error	TERMS5	Information		100
Device	NDC+		ATM Systems	Terminal status is fatal error	TERMS5	Information		100

Таблица сообщений, формируемых при работе банкоматов

Данная таблица содержит следующие поля:

- *Message Scope* – получатель/источник сообщения:
 - "Device" – банкомат;
 - "Device Group" – группа банкоматов;
 - "Physical Channel" – контроллер;
- *Protocol* – наименование протокола (см. "[Перечень протоколов банкоматов \(ATM Protocols\)](#)")
- *Device Type* – типы банкоматов, зарегистрированные в системе; данное поле используется для детализации сообщений по типам устройств, например, для банкоматов разных типов при выполнении одной и той же операции (при одинаковом значении в поле *Code*) могут формироваться сообщения с различными значениями в поле *Error Level*;
- *Hardware Type* – наименование компонента банкомата (см. "[Перечень компонентов банкоматов \(ATM Hardware Types\)](#)"), при работе которого формируется данное сообщение; указывается только для *Message Scope* = "Device";
- *Name* – описание сообщения;
- *Code* – код сообщения;

- *Error Level* – уровень ошибки, в который переводится соответствующий компонент банкомата при получении данного сообщения; в зависимости от уровня ошибки полученного сообщения текущий статус компонента и банкомата может изменяться следующим образом:
- "OK" (код "0") – статус компонента "OK"; статус банкомата "OK" – если на других компонентах нет, и не было ошибок; статус банкомата "Information" – если на других компонентах нет ошибок, а на данном была; статус банкомата "Information", "Warning" или "Error" – в зависимости от статуса других компонентов;
- "Information" (код "1") – статус компонента и банкомата не изменяется;
- "Warning" (код "2") – статус компонента "Warning"; статус банкомата "Information";
- "Error" (код "3") – статус компонента "Error"; статус банкомата "Warning";
- "Not Configured" код ("4") – статус компонента "Not Configured"; статус банкомата "Information";
- "Unavailable" (код "6") – статус компонента "Unavailable"; статус банкомата "Information";
- "Fatal Error" (код "5") – статус компонента "Fatal Error"; статус банкомата "Error";
- *Group Code* – поле для ввода инструкции банкомату, выполняемой при получении данного сообщения (формат инструкции: #<код операции> или некоторое мнемоническое имя); например:
- DI01=1113 – установка статусов кассет, где DI01 – код диспенсера в соответствии с перечнем "ATM Hardware Types", 1, 1, 1, 3 – коды статусов кассет (см. коды в описании поля [Error Level](#));
- #S340 – отправка ключей на банкомат (где S340 – код соответствующей операции в перечне "ATM Operations");
- STARTUP – выполнение последовательности команд: вывод из обслуживания, запрос конфигурационной информации и счётчиков, ввод в обслуживание;
- *Security* – уровень доступа оператора процессингового центра к управляющим командам контроллера банкоматов. Уровень доступа для групп пользователей задается в поле *Security Level* формы "Constants for <наименование группы>", вызываемой нажатием на кнопку [Constants] в табличной форме "User Groups and Users - View" (Full → DB Administrator Utilities → Users & Grants → User Groups and Users – View).

Кнопка [Description] предназначена для вызова на экран формы "Description for <описание сообщения>", содержащей дополнительную информацию к сообщению.

2. Описание нового банкомата и его конфигурирование

2.1 Настройка контракта банкомата и связанного с ним устройства

Описание ввода нового контракта банкомата приведено в разделе "Ввод новых контрактов устройств" документа "Модуль эквайринга. Руководство пользователя".

Описание настройки устройства под контрактом банкомата приведено в разделе "Конфигурирование устройств" документа "Модуль эквайринга. Руководство пользователя").

2.1.1 Настройка допустимых операций банкомата

Настройка допустимых операций банкомата производится с помощью табличной формы "Operations for <наименование банкомата>".

Данная форма может быть вызвана на экран двумя способами:

- после выбора в меню пользователя пункта "Acquiring → ATM Controller → ATM Device Management" следует выбрать требуемый банкомат и в форме "ATM Device Management" нажать на кнопку [Operations];
- после выбора в меню пользователя пункта "Acquiring → Acquiring Contracts" следует выбрать требуемый счетовой контракт, нажать в форме счетового контракта на кнопку [Devices], выбрать требуемый банкомат, нажать в форме контракта устройства на кнопку [ATM] и в форме конфигурирования устройства нажать на кнопку [Operations].

Operations for MERCHANT 5				
<div> <div><< < > >></div> <div>1 of 38</div> <div>b x</div> </div>				
Operation Type	Status	Hardware Problem	Last Changed	
→ Cash Dispense with ticket	Active		27/05/02 14:26.38	
Balance Inquire	Active		27/05/02 14:26.38	
Cash Dispense without ticket	Active		27/05/02 14:26.38	
Master key Change	Active		22/03/02 13:24.43	
COMM key under Master key Change	Active		22/03/02 13:24.43	
COMM key under COMM key Change	Active		22/03/02 13:24.43	
MAC key under Master key Change	Active		22/03/02 13:24.43	
MAC key under COMM key Change	Active		22/03/02 13:24.43	
Set B-key as current COMM key	Active		22/03/02 13:24.43	
Set B-key as current MAC key	Active		22/03/02 13:24.43	
Send new Configuration Data	Active		22/03/02 13:24.25	
Send new Screens/Keyboard Data	Active		22/03/02 13:24.25	
Send new State Tables	Active		22/03/02 13:24.43	
<div> <div>Ins Del Query Ch Status History</div> </div>				

Список операций банкомата, разрешенных для выполнения

Удаление операции из списка разрешенных осуществляется после выбора требуемой строки в таблице нажатием на кнопку [Del].

Запретить выполнение операции можно также путем изменения ее статуса после нажатия на кнопку [Ch Status]. Нажатие на данную кнопку изменяет статус разрешенной операции со значения "Active" на значение "Closed".

Поле *Last Changed* содержит дату и время последнего изменения статуса операции.

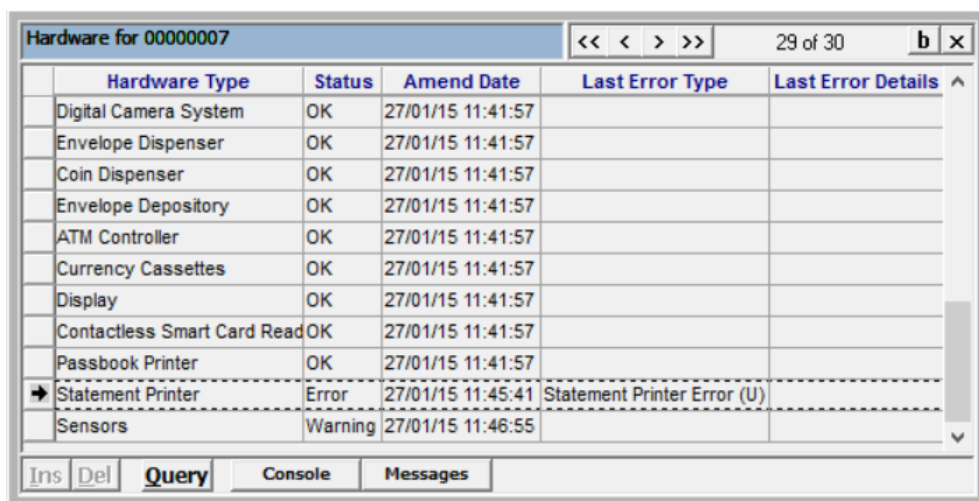
Восстановление списка разрешенных операций (в соответствии с перечнем "ATM Operations") после удаления строк из таблицы можно осуществить с помощью кнопки [Setup] с выбором пункта контекстного меню [Check and Fill] в форме "ATM Device Management" или в форме конфигурирования устройства "ATM for <идентификатор устройства>".

2.1.2 Настройка компонентов банкомата (Hardware)

Настройка компонентов банкомата производится с помощью табличной формы "Hardware for <наименование банкомата>".

Данная форма может быть вызвана на экран двумя способами:

- после выбора в меню пользователя пункта "Acquiring → ATM Controller → ATM Device Management" следует выбрать требуемый банкомат и в форме "ATM Device Management" нажать на кнопку [Hardware];
- после выбора в меню пользователя пункта "Acquiring → Acquiring Contracts" следует выбрать требуемый счетовой контракт, нажать в форме счетового контракта на кнопку [Devices], выбрать требуемый банкомат, нажать в форме контракта устройства на кнопку [ATM] и в форме конфигурирования устройства нажать на кнопку [Hardware].




Hardware Type	Status	Amend Date	Last Error Type	Last Error Details
Digital Camera System	OK	27/01/15 11:41:57		
Envelope Dispenser	OK	27/01/15 11:41:57		
Coin Dispenser	OK	27/01/15 11:41:57		
Envelope Depository	OK	27/01/15 11:41:57		
ATM Controller	OK	27/01/15 11:41:57		
Currency Cassettes	OK	27/01/15 11:41:57		
Display	OK	27/01/15 11:41:57		
Contactless Smart Card Read	OK	27/01/15 11:41:57		
Passbook Printer	OK	27/01/15 11:41:57		
Statement Printer	Error	27/01/15 11:45:41	Statement Printer Error (U)	
Sensors	Warning	27/01/15 11:46:55		

Таблица для настройки компонентов банкомата

Список доступных для настройки компонентов банкомата формируется следующим образом:

- на основе списка компонентов, заданных по умолчанию для данного типа банкомата (см. "[Перечень типов банкоматов \(ATM Types\)](#)");

- если для данного типа банкомата не указаны компоненты, используемые по умолчанию, то список формируется на основе всего перечня "ATM Hardware Types" (см. "[Перечень компонентов банкоматов \(ATM Hardware Types\)](#)").

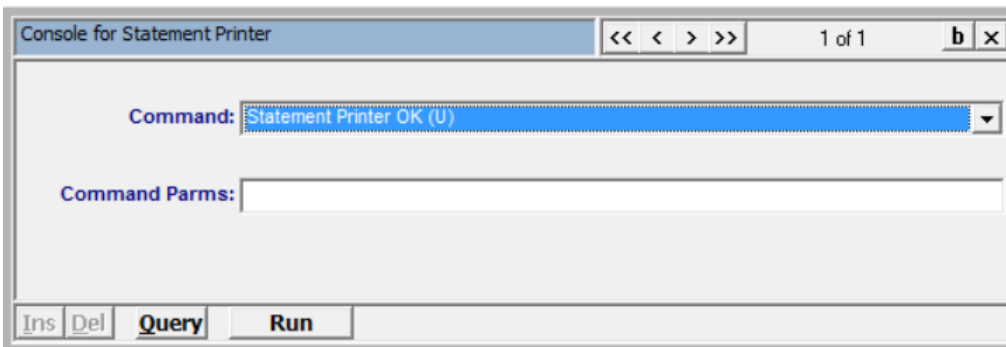
 В этом случае компоненты, отсутствующие в банкомате, необходимо отключить, установив им статус "Not Configured" (порядок изменения статусов компонентов описан ниже).

- Заполнение данного списка записями, актуальными для текущей конфигурации, осуществляется с помощью кнопки [Setup] с выбором пункта контекстного меню [Check and Fill] в форме "ATM Device Management" или в форме конфигурирования устройства "ATM for <идентификатор устройства>".

Поле *Amend Date* табличной формы "Hardware for <наименование банкомата>" содержит дату и время последнего изменения статуса компонента. В случае возникновения ошибки поля *Last Error Type* и *Last Error Details* содержат тип и текст ошибки соответственно.

Нормальная работа компонента банкомата возможна в том случае, если статус компонента установлен в значение "OK". Компонент также может функционировать в статусе "Warning", но данное состояние сигнализирует о возможных последующих проблемах в его работе.

Для изменения статуса компонента следует, выбрав требуемую строку таблицы, нажать на кнопку [Console]. По этой команде на экране будет представлена форма "Console for <наименование компонента>". В поле *Command* данной формы следует выбрать требуемую управляющую команду, например, "<наименование компонента> OK", и нажать на кнопку [Run].



Форма для ввода команд управления компонентами банкомата

Установить статус "OK" одновременно для всех компонентов в списке можно с помощью кнопки [Setup] с выбором пункта контекстного меню [Set to OK] в форме "ATM Device Management" или в форме конфигурирования устройства "ATM for <идентификатор устройства>".

Для отключения компонента, выбранного в форме "Hardware for <наименование банкомата>", следует выбрать в поле *Command* формы "Console for <наименование компонента>" управляющую команду "<наименование компонента> Not Configured" и нажать на кнопку [Run]. Подробнее об управляющих командах банкомата см. в разделе "Команды управления банкоматом" документа "Мониторинг сети банкоматов".

Информация об истории сообщений, связанных с данным компонентом, содержится в табличной форме "Messages for <наименование компонента>". Данная форма вызывается из

таблицы "Hardware for <наименование банкомата>" после выбора требуемой строки и нажатия на кнопку [Messages].

Messages for Statement Printer								<< < > >>		3 of 93	b x
Record ID	Message Time	Message Text	Message Type	Error Level	Code	Status	Hardware				
3773	27/01/15 11:45:41		Statement Printer Error (U)	Error	1VE003	Posted	Statement Printer				
3771	27/01/15 11:44:54		Statement Printer OK (U)	OK	1VE0	Posted	Statement Printer				
→ 3769	27/01/15 11:44:18		Statement Printer Fatal Error (U)	Fatal Error	1VE004	Posted	Statement Printer				
1796	18/11/14 14:43:31		Statement Ribbon - Not Configured	Not Configured	2F1S220	Posted	Statement Printer				
1795	18/11/14 14:43:31		Statement Paper - Not Configured	Not Configured	2F1S210	Posted	Statement Printer				
1773	18/11/14 14:43:31		Statement printer - Routine error	Information	2F1F211	Posted	Statement Printer				
1749	18/11/14 14:43:30		Statement Ribbon - Not Configured	Not Configured	2F1S220	Posted	Statement Printer				
1748	18/11/14 14:43:30		Statement Paper - Not Configured	Not Configured	2F1S210	Posted	Statement Printer				
1726	18/11/14 14:43:30		Statement printer - No error	OK	2F1F210	Posted	Statement Printer				

Табличная форма системных сообщений, формируемых при изменении статуса компонента банкомата

2.1.3 Управление состоянием банкомата

При нажатии на кнопку [State] в форме "ATM Device Management" или в форме конфигурирования устройства "ATM for <идентификатор устройства>" предоставляется следующая информация о текущем состоянии банкомата:

- *Status* – текущий статус банкомата ("OK", "Information", "Warning", "Error", "Not Configured", "Fatal Error");
- *Online* – признак доступности банкомата в режиме онлайн ("Yes"/"No");
- *Online Service* – код коммуникационного сервиса, идентифицирующего соединение контроллера с банкоматом.

Перевод банкомата в рабочее состояние (статус "OK") осуществляется с помощью кнопки [Setup] с выбором пункта контекстного меню [Set to OK] в форме "ATM Device Management" или в форме конфигурирования устройства "ATM for <идентификатор устройства>". При этом в статус "OK" переходят все компоненты данного устройства.

Для временного вывода банкомата из обслуживания, при котором система WAY4 игнорирует все запросы и сообщения, полученные от устройства, необходимо воспользоваться кнопкой [Setup] с выбором пункта контекстного меню [Set to Repair]. При этом банкомат переходит в статус "Not Configured". Возобновление обслуживания банкомата осуществляется переводом в статус "OK" (как описано выше).

2.1.4 Специфицирование ключей шифрования

Ключи шифрования создаются сотрудником системы безопасности с помощью криптографического оборудования и представляют собой числа определенной разрядности.

Ключи шифрования хранятся в системе и в банкомате только в зашифрованном с помощью того или иного ключа шифрования виде. Для контроля правильности ключа шифрования используется так называемая контрольная величина ключа (Check Value). Данная величина

определяется только значением ключа шифрования и не зависит от того, каким способом он зашифрован.

Специфицирование ключей шифрования осуществляется в форме "Keys for <наименование банкомата>".

Данная форма может быть вызвана на экран двумя способами:

- после выбора в меню пользователя пункта "Acquiring → ATM Controller → ATM Device Management" следует выбрать требуемый банкомат и в форме "ATM Device Management" нажать на кнопку [Keys];
- после выбора в меню пользователя пункта "Acquiring → Acquiring Contracts" следует выбрать требуемый счетовой контракт, нажать в форме счетового контракта на кнопку [Devices], выбрать требуемый банкомат, нажать в форме контракта устройства на кнопку [ATM] и в форме конфигурирования устройства нажать на кнопку [Keys].

Keys for TEST ACQ										<< < > >>		1 of 3		b x	
Key Algorithm	Key Type	Key Name	DES Key	Key Check	Used as MK	Storage MK	Serial Number	Is Active	Date From	Date To	Max Usage	M			
→ 3DES ABA	Terminal Authentication	Terminal Authentication	U9CB60C31A96C742	5AE541				Active	18/11/14 00:00/00/00	0	0	0			
3DES ABA	Terminal PIN Key	Terminal PIN Key	UD387E838B70E8FAE	A7223F				Active	18/11/14 00:00/00/00	0	0	0			
3DES ABA	Terminal Authentication	Terminal Authentication						Inactive	20/11/14 00:00/00/00	0	0	0			
<div>< ></div>															
Ins Del Query		Manage		Key Options											

Форма для специфицирования дополнительных ключей шифрования банкомата

Данная форма содержит следующие поля:

- *Key Algorithm* – выбор из списка для указания алгоритма шифрования, для которого данный ключ будет использоваться;
- *Key Type* – тип ключа шифрования, указываемый с помощью выбора из списка, формируемого на основе системного перечня "PM Key Types";
- *Key Name* – наименование ключа шифрования;
- *DES Key* – поле для ввода значения ключа шифрования, зашифровано с помощью локального мастер ключа (Local Master Key) HSM (Host Security Module);
- *Key Check* – контрольная величина ключа шифрования;
- *Used As MK* – поле определяет, будет ли ключ использоваться в качестве мастер-ключа;
- *Storage MK* – поле служит для выбора из списка мастер-ключа, используемого для шифрования данного ключа при передаче его на терминал; в список попадают ключи со значением "Yes" в поле *Used as MK*;
- *Serial Number* – идентификатор ключа, определяющий его значение среди ключей одного типа;
- *Is Active* – поле, указывающее на возможность использования ключа шифрования; возможные значения:
 - "Active" – активный ключ (используется при шифровании);
 - "Inactive" – неактивный ключ (не используется при шифровании);
 - "Locked" – ключ, заблокированный в результате превышения числа попыток неправильного использования (не используется при шифровании);
- *Date From* – поле для ввода начальной даты интервала времени, в течение которого данный ключ может быть использован;

- *Date To* – поле для ввода конечной даты интервала времени, в течение которого данный ключ может быть использован;
- *Max Usage* – поле для ввода числа, определяющего, сколько раз данный ключ шифрования может быть использован;
- *Max Wrong Attempts* – число попыток неправильного использования ключа, после которого происходит его блокировка;
- *Wrong Attempts Threshold* – число попыток неправильного использования ключа, после которого происходит сигнализация о данном факте;
- *Current Usage* – поле, содержащее значение, указывающее, сколько раз был использован данный ключ шифрования;
- *Wrong Attempts* – счетчик попыток неправильного использования ключа;
- *Storage Form* – форма хранения ключа в БД;
- *Key Code* – значение типа ключа (*Key Type*), представленное в форме, указанной в поле *Storage Form*;
- *Parent Key* – родительский ключ;
- *Add Data* – дополнительные данные.

Кнопка [Manage] формы "Keys for <наименование банкомата>" вызывает на экран форму "DES Management Mode", предназначенную для генерации ключей.

Кнопка [Key Options] формы "Keys for <наименование банкомата>" вызывает на экран форму "Key Options for Terminal PIN Key", предназначенную для управления дополнительными параметрами ключа.

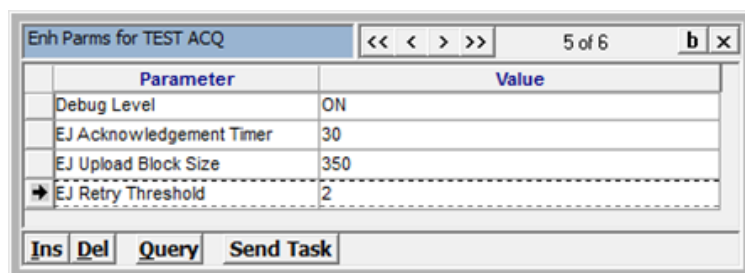
При работе с данной формой пользователь имеет возможность добавлять записи в таблицу, используя кнопку [Ins], а также удалять записи, используя кнопку [Del].

2.1.5 Включение режима MAC-подписи

Для включения режима использования кода аутентификации сообщений (Message Authentication Code, MAC) следует установить в поле *Mac Status* формы "ATM Device Management" или "ATM for <наименование банкомата>") значение "Mandatory". В случае установки в данном поле значения "None" указанный режим отключен.

2.1.6 Дополнительные параметры устройства

Список конфигурационных параметров устройства может быть расширен путем указания дополнительных параметров в табличной форме "Enh Parms for <наименование банкомата>". Данная форма вызывается из форм "ATM for <наименование банкомата>" или "ATM Device Management" с помощью кнопки [Enh Parms].



Parameter	Value
Debug Level	ON
EJ Acknowledgement Timer	30
EJ Upload Block Size	350
EJ Retry Threshold	2

Buttons: Ins, Del, Query, Send Task

Дополнительные параметры устройства

Форма содержит следующие поля:

- *Parameter* – наименование одного из доступных для выбора параметров;
- *Value* – значение параметра.

Кнопка [Send Task] для параметров банкоматов не используется.

2.2 Настройка соединения банкомата с хостом WAY4

Взаимодействие банкомата с хостом WAY4 осуществляется посредством TCP-соединений, обеспечивающих передачу сообщений в обоих направлениях. Каждое соединение TCP уникально и однозначно идентифицируется парой сокетов (набор из четырех элементов, определяющий две конечные точки соединения: локальный IP-адрес терминала, локальный порт TCP, удаленный IP-адрес хоста и удаленный порт TCP) в объединенной сети.

Для обеспечения уникальности соединения TCP банкомат должен использовать динамически назначаемые порты, то есть порты с непродолжительным временем жизни.

Для получения актуальной информации о состоянии соединения должен быть настроен таймер контроля работоспособности (keep-alive) (см. описание параметра `tcp_keepAlive` в разделе ["Конфигурационный файл контроллера"](#)).

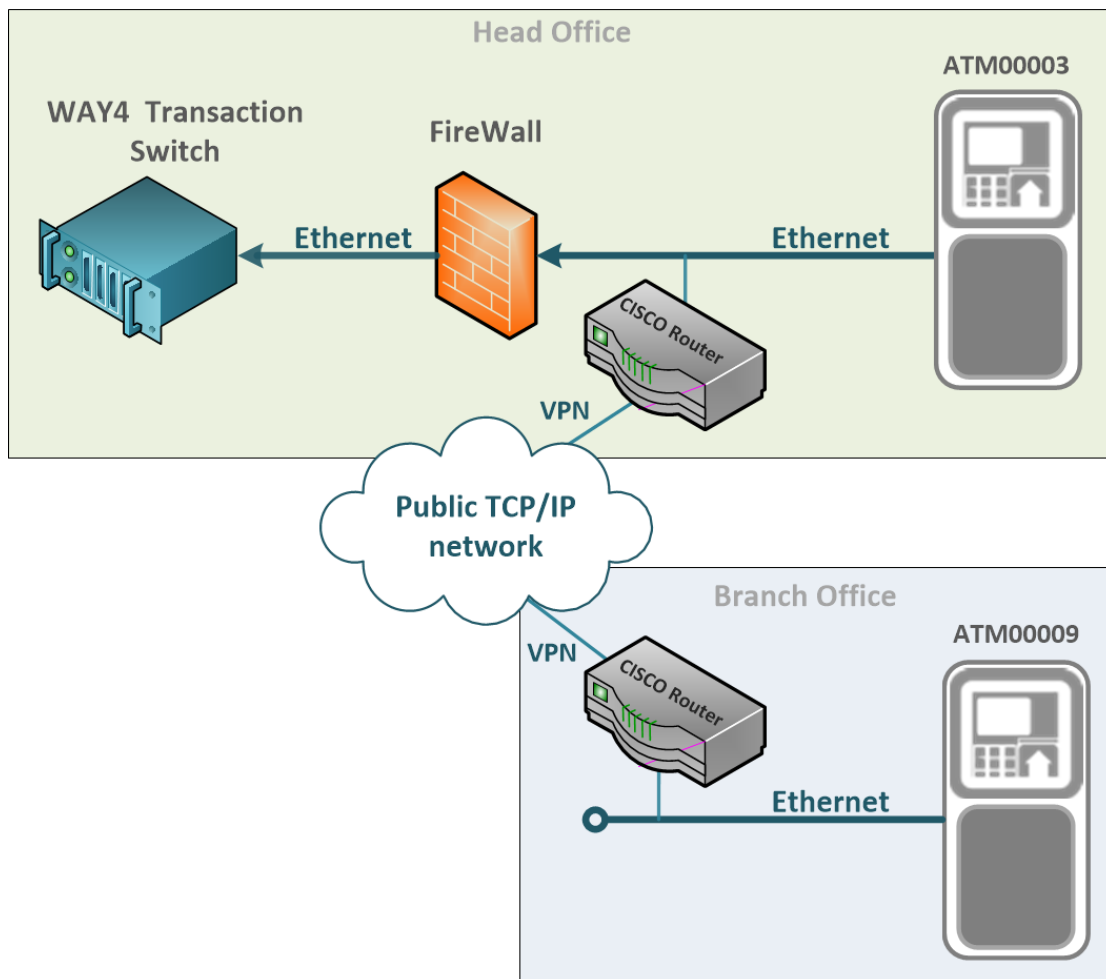
Банкомат устанавливает связь с контроллером на основании IP-адреса и номера порта, которые задаются в настройках программного обеспечения устройства. Соответствующий порт должен быть указан в качестве прослушиваемого в настройках контроллера (см. ["Конфигурационный файл контроллера"](#)). При установке соединения контроллер проверяет наличие зарегистрированного в БД WAY4 устройства с IP-адресом, соответствующим данному банкомату (см. раздел ["Конфигурирование устройств"](#) документа ["Модуль эквайринга. Руководство пользователя"](#)).

Для работы с банкоматом контроллер формирует специальный код коммуникационного сервиса, состоящий из идентификатора узла WAY4 Transaction Switch, на котором функционирует соответствующий сервис (параметр `"node"` конфигурационного файла `node.properties`), и наименования сервиса, с которым установлено соединение. После соединения данный код сохраняется в БД WAY4 и отображается в форме состояния устройства в поле *Online Service* (доступ через ["Acquiring → ATM Controller → ATM Device Management"](#) по кнопке [State]).

При возникновении проблем соединения банкомат может динамически переключаться между узлами WAY4 Transaction Switch (например, между основным и резервным). В этом случае для

банкомата устанавливается новый код коммуникационного сервиса, который фиксируется в БД WAY4 для соответствующего устройства.

Схема подключения банкоматов к WAY4 Transaction Switch по выделенной линии с использованием протокола TCP/IP представлена на рисунке:



Подключение банкоматов к WAY4 Transaction Switch по выделенным линиям с использованием протокола TCP/IP

2.3 Подготовка конфигураций банкоматов и их загрузка

Ниже представлен порядок подготовки и загрузки конфигурационных файлов банкомата.

2.3.1 Подготовка файлов конфигурации

Файл конфигурации банкомата может содержать следующую информацию:

- базовые состояния банкомата (States) – компоненты конфигурационного файла, определяющие последовательность выполнения банкоматом операций в зависимости от действий пользователя, при отказах оборудования, отказе в обслуживании и т.д.;

- шаблоны экранов банкомата (Screens) – компоненты конфигурационного файла, определяющие внешний вид экрана банкомата на всех возможных этапах выполнения операций (экранные меню, сообщения пользователю и т. д.);
- таблица финансовых институтов (FIT) – компоненты конфигурационного файла, используемые для определения принадлежности банковской карты к той или иной платежной системе;
- конфигурационные параметры, различные дополнительные установки, логический номер банкомата, значения временных интервалов и задержек;
- расширенные конфигурационные параметры, различные дополнительные установки, логический номер банкомата, значения временных интервалов и задержек;
- конфигурационные данные зарезервированных шаблонов экранов, которые используются во время операций;
- конфигурационные данные для операций по смарт-картам;
- скриптовые команды контроллеру банкоматов:
- #S331 – изменение мастер ключа банкомата, который пересылается в банкомат зашифрованным с помощью текущего мастер-ключа;
- #S332 – изменение PIN-ключа банкомата, который пересылается в банкомат зашифрованным с помощью текущего мастер-ключа;
- #S333 – изменение PIN-ключа банкомата, который пересылается в банкомат зашифрованным с помощью текущего PIN-ключа;
- #S335 – изменение ключа электронной подписи (MAC-ключа), который пересылается в банкомат зашифрованным с помощью текущего мастер-ключа;
- #S336 – изменение ключа электронной подписи (MAC-ключа), который пересылается в банкомат зашифрованным с помощью текущего PIN-ключа;
- #S340 – изменение ключа электронной подписи (MAC-ключа) и PIN-ключа банкомата, которые пересылаются в банкомат зашифрованным с помощью текущего мастер-ключа;
- #S341 – изменение ключа электронной подписи (MAC-ключа) и PIN-ключа банкомата, которые пересылаются в банкомат зашифрованным с помощью текущего мастер-ключа, и установка текущего значения идентификатора конфигурации (Configuration ID);
- #S334 – установка в банкомате PIN-ключа из ячейки, которая содержит ключ, введенный сотрудниками безопасности непосредственно на банкомате;
- #S337 – установка в банкомате ключа электронной подписи (MAC-ключа) из ячейки, которая содержит ключ, введенный сотрудниками безопасности непосредственно на банкомате;
- # S322 – операция по установке времени и даты в банкомате.

Конфигурационный файл может содержать несколько вариантов конфигурации банкомата, различающихся идентификатором конфигурации (Configuration ID).

Каталог WAY4 Transaction Switch, в который помещаются файлы конфигурации для каждого типа банкоматов, а также наименование этих файлов должны соответствовать значениям в поле *Configuration File* перечня типов банкоматов (см. "[Перечень типов банкоматов \(ATM Types\)](#)").

Фрагмент конфигурационного файла:

```

100002001 cfJGTEMPORARILY OFF-LINE
210000205 J05500005505500000000000000
300000 0000000003203000116000_000_006075070020807509072
510000000 00010309800425525500000013200000000003113800
@10000001 C04ZZZZ5ZZ9699
A00000 000_006075070020807509072
B10000 00000000000000000000000000
G00000D0010 034
*00000#S332

```

Файл конфигурации банкомата имеет следующий формат:

Позиция в строке	Значение	Параметр
1	1	Шаблон экрана банкомата.
	2	Базовое состояние банкомата.
	3	Конфигурационные параметры, значения временных задержек.
	5	Таблицы финансовых институтов.
	A	Дополнительные параметры.
	B	Параметр, определяющий, для каких полей сообщения будет использоваться код аутентификации сообщений (Message Authentication Code, MAC).
	G	Параметр, определяющий перезапись системных шаблонов экранов банкомата.
	*	Инструкция контроллеру банкоматов.
2	1/0	Флажок, определяющий, используется ли код аутентификации сообщений (MAC) для соответствующей строки файла конфигурации; "1" – MAC используется, "0" – MAC не используется.
3-6	Числа от 0000 до 9999	Идентификатор конфигурации.

Позиция в строке	Значение	Параметр
7-...	Символы в соответствии со спецификацией устройства	Данные о базовых состояниях, шаблонах экранов, финансовых институтах и т. д.

2.3.2 Посылка конфигурации одному банкомату

Обновление конфигурации банкомата осуществляется через консоль управления путем выполнения управляющей команды контроллера банкоматов "Send new Configuration to ATM" (см. раздел "Команды загрузки конфигурационных данных" документа "Мониторинг сети банкоматов").

2.3.3 Посылка конфигурации всем банкоматам

Обновление конфигурации всех банкоматов, зарегистрированных в системе, может быть выполнено через консоль управления группой банкоматов командой "Send new Configuration to GROUP" (см. раздел "Команды загрузки конфигурационных данных" документа "Мониторинг сети банкоматов"). При этом конфигурация конкретного банкомата будет загружена в соответствии со значением идентификатора конфигурации, заданным в поле *Configuration* для соответствующего банкомата (см. раздел "Конфигурирование устройств" документа "Модуль эквайринга. Руководство пользователя).

3. Конфигурационные файлы контроллера банкоматов

Настройка контроллера банкоматов включает в себя:

- настройку сервиса ATMController на платформе WAY4 Transaction Switch;
- подготовку скриптов обработки запросов и ответных сообщений для обеспечения процессов взаимодействия контроллера с банкоматами;
- подготовку шаблонов для вывода информации в чеках и на экране банкомата.

Для корректной работы контроллера банкоматов может также потребоваться настройка следующих сервисов WAY4 Transaction Switch:

- AcqDBService – сервис взаимодействия с модулем эквайринга системы WAY4 (конфигурационный файл AcqDB.s.xml);
- AuthService – сервис авторизации в системе WAY4 (конфигурационный файл AuthService.s.xml);
- HSMThalesAdapter – сервис взаимодействия с криптографическим оборудованием (конфигурационный файл HSMAdapter.s.xml);
- IntranetAdapter – сервис взаимодействия с программными компонентами интранет-инфраструктуры системы WAY4 (конфигурационный файл Intranet.s.xml);
- IssDBService – сервис взаимодействия с модулем эмиссии системы WAY4 (конфигурационный файл IssDB.s.xml);
- AppRoutingService – сервис маршрутизации сообщений между сервисами WAY4 Transaction Switch (конфигурационный файл Routing.s.xml).

3.1 Конфигурационный файл контроллера

Конфигурационный файл контроллера банкоматов размещается в каталоге WEB-INF/conf/application относительно рабочего каталога WAY4 Transaction Switch и по умолчанию имеет наименование ATMController.s.xml.

Основные параметры:

- port – параметр определяет номер свободного порта прослушивания для установки TCP-соединения с банкоматом;
- tcp_keepAlive – параметр определяет необходимость (true/false) включения механизма контроля работоспособности (keep-alive) соединения (рекомендуется установить значение "true");
- recTimeoutSec – параметр определяет максимальный временной интервал в секундах для получения сообщения в рамках TCP-соединения;
- sendTimeoutSec – параметр определяет максимальный временной интервал в секундах для отправки сообщения в рамках TCP-соединения;

- **allowedOperations** – список операций (через пробел), поддерживаемых в дополнении к набору базовых операций решения "ATM Management". Функциональность, которую реализуют указанные в списке операции, предоставляется по отдельному соглашению с OpenWay.

К базовым операциям решения "ATM Management" (поддерживаемых по умолчанию) относятся:

- **CASH_WITHDRAWAL** – снятие наличных;
- **BALANCE_INQ** – запрос баланса;
- **MINI_STATEMENT** – запрос мини-выписки;
- **PIN_CHANGE** – смена PIN;
- **REPLENISHMENT** – инкассация кассет банкомата;
- **END_OF_DAY** – получение выписки по состоянию кассет;
- **ATM_SERVICE** – техническое обслуживание банкомата.

При запуске сервиса ATMController полный список доступных операций фиксируется в log-файле Transaction Switch и в базе данных WAY4 (доступен для просмотра в форме "Full\Process Log\Process Log").

Пример конфигурационного файла ATMController.s.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<service xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  singleton="true"
  autoStarted="true"
>
  <appComponent xsi:type="ATMController"
    allowedOperations="OPER_NAME1 OPER_NAME2">
    <transport
      xsi:type="ExternalServer"
      port="5262"
      tcp_keepAlive="true"
      recTimeoutSec="5"
      sendTimeoutSec="50">
      <filter
        xsi:type="BinLen"
        headerLength="2"/>
      </transport>
    </appComponent>
    <dependency service="AcqDB"/>
    <dependency service="HSMAdapter"/>
  </service>
```

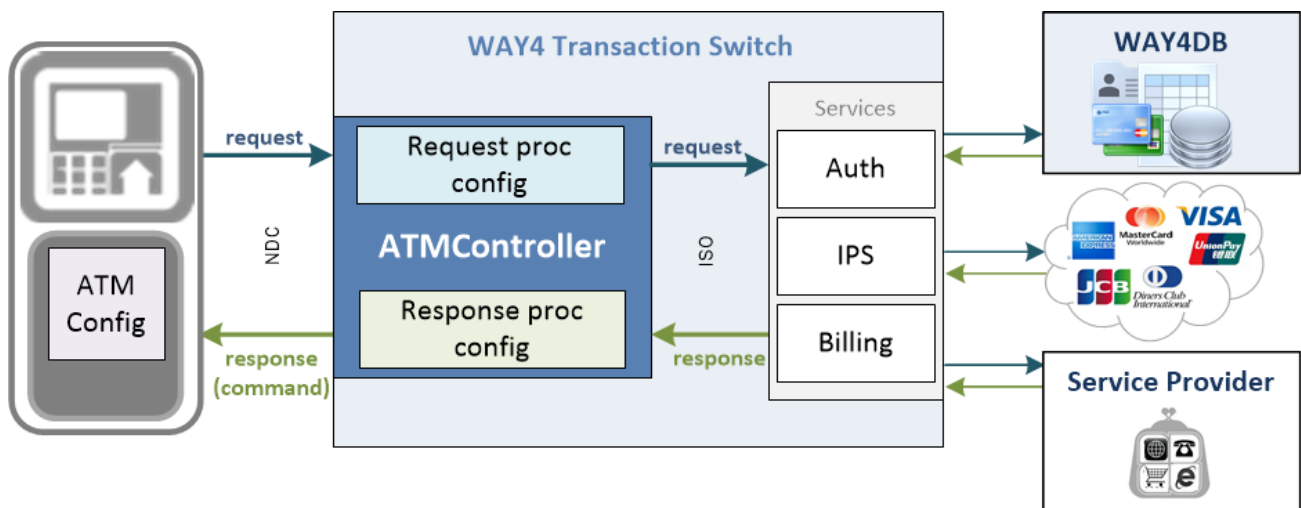


Изменение конфигурационного файла рекомендуется производить только под контролем сотрудников OpenWay.

3.2 Файлы настройки взаимодействия контроллера с банкоматами

Настройка взаимодействия контроллера с банкоматами включает в себя:

- определение правил обработки полученных от банкомата запросов, для их дальнейшего преобразования в необходимый формат и передачи интерфейсным сервисам WAY4 Transaction Switch (с банковскими, платежными системами или системами приема платежей) (см. "[Файл конфигурации обработки запросов](#)");
- определение правил обработки ответов, полученных от интерфейсных каналов для формирования соответствующих команд банкомату (см. "[Файл конфигурации обработки ответов](#)").



Взаимодействие банкомата с контроллером

Указанные правила определяются в соответствующих конфигурационных файлах, реализованы на языке Java Groovy. При необходимости редактирования или добавления новых groovy-скриптов необходимо придерживаться синтаксиса данного языка (см. <http://www.groovy-lang.org/>).

3.2.1 Принципы работы с конфигурационными скриптами

Все используемые в конфигурации groovy-скрипты должны быть размещены в каталоге WEB-INF/inv/atm/scripts/, при этом для работы с текущей конфигурацией контроллер использует файл /inv/<atm_conf>/resources/configuration.prd.

В случае изменения конфигурационных скриптов для применения новых настроек необходимо использовать следующие консольные команды:

- compileScripts - компиляция новой версии скриптов на основе изменений, внесенных в WEB-INF/inv/atm/scripts/, и создание нового файла configuration.prd.
- installScripts - установка вновь созданного configuration.prd в качестве используемого ATM-контроллером.

При неудачной компиляции новых скриптов соответствующие ошибки фиксируются в log-файле (выводятся в окно консоли), новый configuration.prд не создается.

После успешной компиляции старая версия конфигурации архивируется в файл вида configuration.prд.yyyuMMddHHmmss, где в качестве даты используется дата компиляции данной версии. Например, если конфигурация была создана 6 февраля 2019 в 15:20:00, то архивный файл будет иметь наименование configuration.prд.20190206152000.

Описанные действия по компиляции не распространяются на файл Encoding.groovy, размещенный в каталоге WEB-INF/inv/atm/scripts/.

При запуске контроллера проверяется наличие скриптов и файла configuration.prд. Если скрипты и configuration.prд отсутствуют, то запуск сервиса не выполняется. Если скрипты отсутствуют, а файл configuration.prд некорректен, то запуск контроллера прекращается. Если файл configuration.prд отсутствует (или некорректен), то он создается автоматически на основе скриптов, размещенных в каталоге WEB-INF/inv/atm/scripts/.

Каждый из groovy-скриптов может быть вызван неоднократно в процессе одной транзакции. Данные, накопленные в контексте транзакции, сохраняются между вызовами скриптов, что позволяет реализовать сложную логику обработки транзакции, а также использовать указанные данные в шаблонах чеков и экранов (посредством структуры "params", см. "[Настройка шаблонов чеков и экранов](#)").

В каждый groovy-скрипт должна быть импортирована специальная структура:

```
com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionContext
```

Данная структура хранит в себе все параметры текущей транзакции и специальный объект – ссылку на процессы, которые можно исполнить в скрипте в контексте транзакции. Каждый скрипт в результате своей работы должен возвращать данную структуру.

Через объект-ссылку можно запустить следующие процессы:

- startOperation – процесс, инициирующий выполнение заданного типа операции в системе WAY4 (формирование эквайрингового документа); параметры отсутствуют;
- cassettesReport – процесс, инициирующий формирование отчета по кассетам для выдачи наличных в системе WAY4; параметры отсутствуют;
- bnaReport – процесс, инициирующий формирование отчета по кассетам для приема наличных в системе WAY4; параметры отсутствуют;
- closeCycle – процесс, инициирующий закрытие финансового цикла банкомата в системе WAY4; в качестве параметра могут использоваться значения типа com.openwaygroup.ts.fp.shared.atm.replenishment.ReplenishmentCycleType:

```
proc.closeCycle.exec { // процесс закрытия цикла выдачи наличных
    ReplenishmentCycleType.CASH_OUT
}

proc.closeCycle.exec { // процесс закрытия цикла приема наличных
    ReplenishmentCycleType.CASH_IN
}
```

- logOperationName – процесс, фиксирующий наименование текущей операции в БД WAY4; параметры отсутствуют;
- setInReplenishment – процесс, переводящий устройство в статус инкассации в БД WAY4; параметры отсутствуют;
- setPostProcess – процесс устанавливает сценарий, используемый после завершения транзакции; в качестве параметра используется значение типа com.openwaygroup.ts.fp.shared.atm.scenario.ScenarioType:

```
proc.setPostProcess.exec { // запрос конфигурации BNA
    ScenarioType.RETRIEVE_NOTE_DEFS
}
```

3.2.2 Файл конфигурации обработки запросов

Правила обработки запросов, поступающих от банкомата, определяются в конфигурационном скрипте RequestMapper.groovy. Наименование данного файла может быть переопределено в конфигурации сервиса ATMController.

Пример конфигурационного скрипта:

```

package com.openwaygroup.ts.atm.controller.handler.scripts // пакет скрипта
//Импорт объектов, используемых в скрипте.
import

com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionContext
import com.openwaygroup.ts.fp.shared.Currency
import com.openwaygroup.ts.atm.controller.handler.scripts.model.ScreenReference
import com.openwaygroup.ts.fp.shared.OperStatus
import com.openwaygroup.ts.fp.shared.atm.command.KeyPosition
import com.openwaygroup.ts.fp.shared.atm.command.KeyStatus
import com.openwaygroup.ts.fp.shared.atm.scripts.AccountType
//Статический импорт.
import static com.openwaygroup.ts.fp.shared.atm.scripts.OperationType.*
import static com.openwaygroup.ts.fp.shared.atm.scripts.DispenseAlgorithmType.*
logger.debug("executing RequestMapper v 3.0")
//Получение контекста из окружения (следующая строка не обязательна, но
//такое объявление удобно для работы автодополнения в IDE)
def ctx = (TransactionContext) binding.ctx
//Определение значения Key Buffer из Transaction Request
def keys = ctx.getMessage().getOperationKeyBuffer();
//Определение типа счета - com.openwaygroup.ts.fp.shared.atm.scripts.AccountType
ctx.accountFrom = AccountType.DEFAULT
//Определение валюты операции
ctx.currency = Currency.byName("RUR")
//Определение языка (язык - любой строковый идентификатор, который также
//можно использовать в шаблонах чеков и экранов). Например, в шаблоне,
//описанном ниже, это выражение: <#if lang == "RU">
ctx.language = (keys =~ /.....A/) ? "RU" : "EN"
//По умолчанию печать чеков разрешена
ctx.printReceipt = true
//Определение типа операции
switch (keys) {
    case ~/C...../: //Данная конструкция groovy позволяет провести
                        //сопоставление строки с образцом - регулярным
                        //выражением
                        //Определение типа операции (enum
                        //com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
    ctx.operation = CASH_WITHDRAWAL
    //Определение типа расчета выдачи наличных (enum
    //com.openwaygroup.ts.fp.shared.atm.scripts.DispenseAlgorithmType)
    ctx.dispenseType = PENALTY_BASED

    //Определение флага печати чека
    ctx.printReceipt = (keys =~ /..A...../)
    break
    case ~/BAAAB.../:
    //Определение типа операции ( enum
    //com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
    ctx.operation = MINI_STATEMENT
    break
    case ~/BAAAA.../:
    //Определение типа операции ( enum
    com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
    ctx.operation = BALANCE_INQ
    //Определение типа счета -

```

```

        //com.openwaygroup.ts.fp.shared.atm.scripts.AccountType
ctx.accountFrom = AccountType.SAVINGS
//Установка тега для вызова start operation
ctx.setDocTag("LANG", "ru")
break
case ~/D...../:
//Определение типа операции ( enum
        //com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
ctx.operation = PIN_CHANGE
//Установка значение новго пипа ( из buffer b )
// ctx.setNewPin(ctx.getMessage().getBufferB())
//или из customer selected pin buffer
ctx.setNewPin(ctx.getMessage().getCspData())
break
case ~/I.AA..../:
//Определение типа операции ( enum
        //com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
ctx.operation = REPLENISHMENT
break
case ~/I.AB..../:
//Определение типа операции ( enum
        //com.openwaygroup.ts.fp.shared.atm.scripts.OperationType )
ctx.operation = END_OF_DAY
break
default: // UNDEFINED – можно не определять (такой тип операции задан
//по-умолчанию )
ctx.operation = _UNDEFINED
break
}
//Использование объект-ссылки на процесс
//для вызова startOperation.
proc.startOperation.exec()
if(ctx.getOperation() in [REPLENISHMENT, END_OF_DAY]){
//При необходимости протоколируется имя операции.
proc.logOperationName.exec()
}
//В случае приема наличных определяется блок для инетрактивной сессии с
//клиентом.
//с последующим выполнением замыкания после получения ответа от банкомата.
ctx.afterProcess { TransactionContext context, def proc ->
//Анализ пполученного bufferB.
def bufferB = context.getMessage().getBufferB()
//Если была нажата кнопка отмены на экране("D")
// cancel на pin-pad ("E")
//или истек таймаут ("T"),
if(!bufferB || (bufferB.substring(0, 1) in ["D", "E", "T"] ))
{
//то отклоняем операцию с кодом CUSTOMER_CANCELLATION (17)
context.setRc(OperStatus.CUSTOMER_CANCELLATION)
}
context
}
}
//Возврат объекта TransactionContext
Ctx

```

В данном примере используется ссылка на шаблон экрана (scr.ftl). Более подробная информация о настройке шаблонов экранов и чеков приведена в разделе "[Настройка шаблонов чеков и экранов](#)".

Так как в представленном примере не определено наименование скрипта, который необходимо использовать для обработки результатов выполнения перечисленных операций, то будет исполняться скрипт RCMapper.groovy (см. "[Файл конфигурации обработки ответов](#)").

3.2.3 Файл конфигурации обработки ответов

Правила обработки ответов (результатов выполнения операций), определяются в конфигурационном скрипте RCMapper.groovy. Данное наименование используется по умолчанию и может быть переопределено через поле stx.rcMapperName в скрипте конфигурации обработки запросов (см. "[Файл конфигурации обработки запросов](#)"). Для обработки результатов операций могут использоваться несколько скриптов, описывающих порядок выполнения дополнительных действий, формирования чеков, экранов и т. д.

Пример конфигурационного скрипта:


```

package com.openwaygroup.ts.atm.controller.handler.scripts
import

com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionContext
import
    com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionPhase
import com.openwaygroup.ts.fp.shared.atm.command.PrintType
import com.openwaygroup.ts.fp.shared.atm.replenishment.ReplenishmentCycleType
import com.openwaygroup.ts.fp.shared.atm.scenario.ScenarioType
import static

com.openwaygroup.ts.atm.controller.handler.scripts.model.PrintReference.printReference
import static
    com.openwaygroup.ts.atm.controller.handler.scripts.model.ScreenReference.screen
import static com.openwaygroup.ts.fp.shared.atm.scripts.OperationType.*
logger.debug("rc script ver. 2.1")
def ctx = (TransactionContext) binding.ctx
//Получение RC (код ответа хранится в виде объекта
//com.openwaygroup.ts.fp.shared.OperStatus, но тут удобнее получить
//стандартный числовой)
def rcCode = ctx.getRc().getResponseCode()
//Определение состояния по умолчанию
ctx.nextStateId = "396" // default next state id
//Очистка буфера экранов
ctx.clearScreens()
//Очистка буфферов чеков
ctx.clearPrints()
switch(rcCode)
{
    case 0:
        if (ctx.operation == CASH_WITHDRAWAL) {
            ctx.nextStateId = "701"
            //Добавление экранов в список для формирования
            //Transaction Reply
            ctx.addScreen (screen("070"))
            ctx.addScreen (screen("071"))
            ctx.addScreen (screen("072"))
            //В зависимости от флага формируется список чеков
            if (ctx.printReceipt) {
                ctx.addPrint (printReference(PrintType.RECEIPT, "consumer1")) //
                //Добавление чека с типом RECEIPT и шаблоном consumer1.ftl
                ctx.addPrint (printReference(PrintType.JOURNAL, "journal1")) //
                //Добавление чека с типом JOURNAL и шаблоном journal1.ftl
            } else {
                ctx.addPrint (printReference(PrintType.JOURNAL, "journal1"))
            }
        }
        if (ctx.operation == BALANCE_INQ) {
            ctx.nextStateId = "701"
            ctx.addScreen(screen("072"))
            ctx.addPrint (printReference(PrintType.RECEIPT, "consumer1"))
            ctx.addPrint (printReference(PrintType.JOURNAL, "journal1"))
        }
        if (ctx.operation == MINI_STATEMENT)

```

```
{
    ctx.nextStateId = "701"
    ctx.addScreen(screen("072"))
    ctx.addPrint( printReference(PrintType.RECEIPT, "statement1"))
    ctx.addPrint( printReference(PrintType.JOURNAL, "journal1"))
}
if(ctx.operation == REPLENISHMENT)
{
    //Использование объекта-ссылки на процессы
    //для выполнения cassettesReport в БД WAY4
    proc.cassettesReport.exec()
    ctx.nextStateId = "701"
    ctx.addScreen(screen("072"))
    ctx.addPrint( printReference(PrintType.JOURNAL_RECEIPT, "admin1"))
    //В случае APPROVE устанавливается скрипт, который будет
    //выполнен после завершения транзакции.
    //скрипт может быть указан как в виде замыкания или лямбды, так
    //и в виде ссылки с именем скрипта
    if(ctx.getPhase() == TransactionPhase.APPROVE)
    {
        //в данном случае ожидается, что в каталоге скриптов будет
        //находиться groovy-скрипт Replenishment.groovy
        ctx.postScriptWith ("Replenishment")
    }
}
break
case 1:
case 2:
case 5:
    ctx.nextStateId = "203"
    break
case 3:
case 15:
    ctx.nextStateId = "202"
    break
case 4:
case 7:
    ctx.nextStateId = "205"
    ctx.addPrint(printReference(PrintType.JOURNAL, "journal1"))
    ctx.retainCard = true
    break
case 13:
    ctx.nextStateId = "209"
    break
case 14:
    ctx.nextStateId = "214"
    break
default:
    ctx.nextStateId = "396"
    break
}
ctx
```

В данном примере используются ссылки на шаблоны чеков (consumer1.ftl, journal1.ftl). Более подробная информация о настройке шаблонов экранов и чеков приведена в разделе "[Настройка шаблонов чеков и экранов](#)".

Также в представленном примере запланировано выполнение скрипта Replenishment.groovy в случае успешного завершения транзакции типа REPLENISHMENT. Replenishment.groovy может иметь, например, следующее содержание:

```
Package com.openwaygroup.ts.atm.controller.handler.scripts
import

com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionContext
import com.openwaygroup.ts.fp.shared.atm.scenario.ScenarioType
logger.debug("replenishment script part 1. version 1.0")
def ctx = (TransactionContext) binding.ctx
//Планирование асинхронного процесса отправки команд на АТМ с последующим исполнением
скрипта ReplenishmentAfter.groovy
ctx.sendAtmCommand().withCommand(ScenarioType.OP_OUT_OF_SERVICE).afterProcess(
"ReplenishmentAfter")
ctx
```

Пример содержания ReplenishmentAfter.groovy:

```
package com.openwaygroup.ts.atm.controller.handler.scripts
import

com.openwaygroup.ts.atm.controller.handler.algorithms.transaction.TransactionContext
import com.openwaygroup.ts.fp.shared.atm.replenishment.ReplenishmentCycleType
logger.debug("replenishment script part 2. version 1.0")
def ctx = (TransactionContext) binding.ctx
//Использование объекта-ссылки на процессы
//для закрытия финансового цикла банкомата в БД WAY4.
proc.closeCycle.exec()
//Использование объекта-ссылки на процессы
// для закрытия финансового цикла с другим типом в БД WAY4.
proc.closeCycle.exec {
    ReplenishmentCycleType.COINS_OUT
}
//Использование объекта-ссылки на процесс
//устанавливающий статус инкасации для устройства в БД WAY4.
proc.setInReplenishment.exec()
ctx
```

3.3 Настройка шаблонов чеков и экранов

Шаблоны для вывода информации в чеках и на экране банкомата формируются посредством шаблонизатора Java FreeMarker (для получения более подробной информации см. <http://freemarker.org/>). Для работы с шаблонами может использоваться редактор jEdit.

В шаблонах могут использоваться следующие ASCII коды:

Код	ASCII значение	Код	ASCII значение	Код	ASCII значение
ACK	Acknowledgment	EM	End of Medium	LF	Line Feed
BS	Back Space	ETX	End of Text	NAK	Negative Acknowledgement
BEL	Bell	EOT	End of Transmission	NUL	Null char
CAN	Cancel	ETB	End of Transmit Block	RS	Record Separator
CR	Carriage Return	ENQ	Enquiry	SI	Shift In / X-Off
DLE	Data Line Escape	ESC	Escape	SO	Shift Out / X-On
DC1	Device Control 1 (oft. XON)	FS	File Separator	SOH	Start of Heading
DC2	Device Control 2	FF	Form Feed	STX	Start of Text
DC3	Device Control 3 (oft. XOFF)	GS	Group Separator	SUB	Substitute
DC4	Device Control 4	HT	Horizontal Tab	SYN	Synchronous Idle
US	Unit Separator	VT	Vertical Tab		

В шаблонах могут использоваться следующие объекты:

Имя	Тип	Поля	Описание
date	java.util.Date		Текущая дата и время.
time	java.util.Date		Текущая дата и время.
rrn	java.lang.String		RRN
stan	java.lang.String		STAN
authCode	java.lang.String		Код авторизации.

Имя	Тип	Поля	Описание
lang	java.lang.String		Язык, установленный для транзакции.
operation	java.lang.String		Тип операции.
rc	java.lang.String		Response code
lun0	java.lang.String		Идентификатор терминала.
terminalId	java.lang.String		Идентификатор терминала.
card	java.lang.String		PAN
applicationId	java.lang.String		EMV AID
applicationLabel	java.lang.String		EMV Application label
cardName	java.lang.String		Тип банковской карты.
amount	com.openwaygroup.ts.fp.shared.Amount	<ul style="list-style-type: none"> currency externalCurrency amount 	Сумма и валюта для снятия наличных.
request	com.openwaygroup.ts.fp.shared.Amount		Сумма и валюта, запрошенная держателем банковской карты.
fee	com.openwaygroup.ts.fp.shared.Amount		Сумма и валюта комиссионных эквайпера.
amounts	java.util.List<com.openwaygroup.ts.fp.shared.ExtendedAmount>	<ul style="list-style-type: none"> amountType accountType 	Список счетов, сумм и валют.
statement	java.util.List<com.openwaygroup.ts.fp.shared.MinistatementRecord>	<ul style="list-style-type: none"> date operationType amount 	Список с записями минивыписки.

Имя	Тип	Поля	Описание
dispense	java.util.List<com.openwaygroup.ts.atm.controller.handler.dispense.DispenseValue>		Количество выданных купюр по кассетам.
atm	java.lang.String		Тип банкомата.
vendor	java.lang.String		Тип банкомата.
location	com.openwaygroup.ts.fp.shared.Location	<ul style="list-style-type: none"> country city address state postalCode street 	Структура, описывающая адресные данные банкомата.
merchant	com.openwaygroup.ts.fp.shared.Merchant	<ul style="list-style-type: none"> name id SIC SICGroupCode url 	Информация о торговце.
acquirer	com.openwaygroup.ts.fp.shared.Acquirer	<ul style="list-style-type: none"> bin location acqBankCode 	Информация об эквайере.
info	java.util.Map<java.lang.String, java.lang.String>		Ассоциативный массив с тегами, выгружаемыми из БД WAY4.
params	java.util.Map<java.lang.String, java.lang.Object>		Ассоциативный массив с параметрами, определенными в groovy-скриптах (см. "Файлы настройки взаимодействия контроллера с банкоматами").
cardsRetained	java.lang.Long		Количество удержанных карт.

Имя	Тип	Поля	Описание
cassettes	java.util.List<com.openwaygroup.ts.fp.shared.atm.device.Cassette>	<ul style="list-style-type: none"> • hardware • status • position • denomination • deviceId • name • denominationValue • cycleNumber • dispenseLimit • remaining • dispensed • deposited • lastTransactionDispensed • lastTransactionDeposited • loaded • diverted1 • diverted2 • retracted • checkDispensed • checkRetracted • checkLoaded • reactivated 	Список кассет (виды деноминации, счетчики).
cassetteAmounts	java.util.List<com.openwaygroup.ts.fp.shared.Amount>		Суммы по кассетам.
protocol	com.openwaygroup.ts.fp.shared.atm.protocol		Тип используемого протокола ("NDC+", "MDS912").
collectionCycle	java.lang.Long		Идентификатор текущего цикла по учету принятых купюр (для включения в инкассаторский чек).
cashinData	java.util.List<com.openwaygroup.ts.fp.shared.atm.message.BNAREc>	<ul style="list-style-type: none"> • amount • notes <ul style="list-style-type: none"> • noteType • denomination • count 	Сумма принятых средств со списком данных по типу купюр (тип купюры (NDC), деноминация (DDC), количества).

Имя	Тип	Поля	Описание
biller	java.lang.String		Наименование провайдера услуг.
numRC	java.lang.String		Числовое значение кода ответа.
srn	java.lang.String		Внутренний идентификатор транзакции (SRN).
transactionDescription	java.lang.String		Данные транзакции (например, номер телефона, для оплаты).
textDetails	java.lang.String		Стока с данными (используется в операциях оплаты провайдерам услуг).
minDenomination	com.openwaygroup.ts.fp.shared.Amount		Минимальный номинал банкнот в банкомате.
maxDenomination	com.openwaygroup.ts.fp.shared.Amount		Максимальный номинал банкнот в банкомате.
dispenserId	java.lang.Int		Идентификатор устройства выдачи.
maxAmounts	java.util.List<com.openwaygroup.ts.fp.shared.Amount>		Максимальная суммы для снятия по купюрам.
settlement	com.openwaygroup.ts.fp.shared.Amount		Сумма и валюта расчетов.
settlement2	com.openwaygroup.ts.fp.shared.Amount		Сумма и валюта расчетов. (для второго деспенсера).
fee2	com.openwaygroup.ts.fp.shared.Amount		Сумма и валюта комиссионных эмитента.
additionalBalances	java.util.List<com.openwaygroup.ts.atm.controller.handler.support.AdditionalBalance>	<ul style="list-style-type: none"> accountName amount 	Список остатков на дополнительных счетах (суммы и валюты).

Имя	Тип	Поля	Описание
cashOut	java.lang.Boolean		Признак присутствия данных по кассетам.
cashOut2	java.lang.Boolean		Признак присутствия данных по кассетам (для второго диспенсера).
replCycle	java.lang.Int		Номер закрываемого цикла (для включения в инкассаторский чек).
replCycle2	java.lang.Int		Номер закрываемого цикла (для включения в инкассаторский чек для второго диспенсера).
cassettes2	java.util.List<com.openwaygroup.ts.fp.shared.atm.device.Cassette>	Аналогично переменной "cassettes"	Список кассет (виды деноминации, счетчики для второго диспенсера).
cassetteAmounts2	java.util.List<com.openwaygroup.ts.fp.shared.Amount>		Суммы по кассетам.
cassetteTotals	java.util.List<com.openwaygroup.ts.fp.shared.Amount>		Общая сумма по кассетам в разрезе валюты.
cassetteTotals2	java.util.List<com.openwaygroup.ts.fp.shared.Amount>		Общая сумма по кассетам в разрезе валюты (для второго диспенсера).
otpList	java.util.List<String>		Список одноразовых паролей.
userId	java.lang.String		Логин для доступа к сервисам дистанционного банковского обслуживания.
password	java.lang.String		Пароль для доступа к сервисам дистанционного банковского обслуживания.

Имя	Тип	Поля	Описание
prePaid	java.util.List<com.openwaygroup.ts atm.controller.handler.template.model.PrePaidData>	<ul style="list-style-type: none"> • code • serialNumber • count • expiry 	Данные скрэтч-карты.
pickList	java.util.Map<java.lang.String, java.lang.String>		Ассоциативный массив для выбора счета из списка счетов.
paymentCode	java.lang.String		Код платежа (Service Code).

Для обращения к полю структуры или элементу ассоциативного массива необходимо использовать точечную нотацию. Например, код валюты для любого объекта Amount может быть получен следующим образом:

```

${amount.currency.name} - для трехбуквенного кода валюты
${amount.currency.code} - для числового ISO-кода

```

Пример шаблона чека:

```

<#setting number_format="0.00" />
<#setting datetime_format="dd.MM.yy HH:mm" />
<#if vendor == "NCR" || vendor == "NCR3G">
    <#if lang == "EN">
        ---- ${ESC}(2BANK -----
        Tel.(322) 32-23-34
        Atm number: ${terminalId}
        Date:      Time: Card number:
        ${time?datetime} ${card[0..*6]}..${card[card?length - 1..]}
        <#if operation == "CASH_WITHDRAWAL" && (amount.amount)??>
        Dispense amount: ${amount.amount?left_pad(12)} <#if (amount.currency)??>${
        {amount.currency.name}</#if>
        </#if>
        <#if amounts??>
            <#list amounts as amt>
                <#switch amt.amountType>
                    <#case "01">
Ledger:          ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>
                    <#case "02">
Available:       ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>
                    <#case "91">
Credit limit:   ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>
                    <#default>
                </#switch>
            </#list>
        </#if>
        ${authCode}/${rrn}
        ${FF}
    </#if>
    <#if lang == "RU">
        ---- ${ESC}(2BANK -----
        Tel.(322) 32-23-34
        Банкомат: ${lun0}
        Дата:      Время:   карта:
        ${time?datetime} ${card[0..*6]}..${card[card?length - 1..]}
        ${ESC}(2
        <#if operation == "CASH_WITHDRAWAL" && (amount.amount)??>
        ВЫДАНА СУММА: ${amount.amount?left_pad(12)} <#if (amount.currency)??>${
        {amount.currency.name}</#if>
        </#if>
        <#if amounts??>
            <#list amounts as amt>
                <#switch amt.amountType>
                    <#case "01">
текущий         ${ESC}(2: ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>
                    <#case "02">
доступный       ${ESC}(2: ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>
                    <#case "91">
кредитный       ${ESC}(2: ${amt.amount?left_pad(12)} ${amt.currency.name}
                    <#break>

```

```
        <#default>
        </#switch>
    </#list>
</#if>
${ESC}(2${authCode}/${rrn}
${FF}
</#if>
</#if>
```

Подготовленные файлы шаблонов *.ftl необходимо размещать в каталоге WEB-INF/inv/atm/templates. Ссылки на данные файлы могут быть использованы в конфигурационных скриптах обработки запросов и результатов выполнения операций (см. "[Файлы настройки взаимодействия контроллера с банкоматами](#)").

Контроллер осуществляет постоянный мониторинг содержания каталога /templates, применяя изменения в шаблонах "на лету", исключая необходимости перезапуска сервиса.

3.3.1 Настройка перекодировки

Для корректного формирования экранов и чеков на основе созданных шаблонов в каталог WEB-INF/inv/atm/templates необходимо включить скрипт Encoding.groovy, определяющий таблицу перекодировки в зависимости от используемых языков. Данный скрипт загружается только при запуске сервиса ATMController. Пример содержимого Encoding.groovy:

```
package com.openwaygroup.ts.atm.controller.handler.scripts
import com.openwaygroup.ts.atm.controller.handler.scripts.encoding.Encoder
import com.openwaygroup.ts.atm.controller.handler.scripts.encoding.EncoderType
import com.openwaygroup.ts.atm.controller.handler.scripts.encoding.StringEncoder
import groovy.transform.BaseScript
@BaseScript com.openwaygroup.ts.platform.ext.script.BaseScript thisScript
encoderScript {

    def rus = new StringEncoder(controlSequence: "/u001B(1", charMap: [
        "A": "A",
        "B": "B",
        "C": "C",
        "D": "D",
        "E": "E",
        "F": "F",
        "G": "G",
        "H": "H",
        "I": "I",
        "J": "J",
        "K": "K",
        "L": "L",
        "M": "M",
        "N": "N",
        "O": "O",
        "P": "P",
        "Q": "Q",
        "R": "R",
        "S": "S",
        "T": "T",
        "U": "U",
        "V": "V",
        "W": "W",
        "X": "X",
        "Y": "Y",
        "Z": "Z",
        "a": "A",
        "b": "B",
        "c": "C",
        "d": "D",
        "e": "E",
        "f": "F",
        "g": "G",
        "h": "H",
        "i": "I",
        "j": "J",
        "k": "K",
        "l": "L",
        "m": "M",
        "n": "N",
        "o": "O",
        "p": "P",
        "q": "Q",
        "r": "R",
        "s": "S",
```

```
"t": "T",
"u": "U",
"v": "V",
"w": "W",
"x": "X",
"y": "Y",
"z": "Z",
"Ě": "t",
"ě": "t",
"A": "f",
"Б": "~",
"B": "d",
"Г": "u",
"Д": "l",
"E": "t",
"Ж": "|",
"З": "p",
"И": "b",
"Й": "q",
"К": "r",
"Л": "k",
"М": "v",
"Н": "y",
"О": "j",
"П": "g",
"Р": "h",
"С": "c",
"Т": "n",
"У": "e",
"Ф": "a",
"Х": "{",
"Ц": "w",
"Ч": "x",
"Ш": "i",
"Щ": "o",
"Ъ": "}",
"Ы": "s",
"Ь": "m",
"Э": "`",
"Ю": "/u007F",
"Я": "z",
"a": "f",
"б": "~",
"в": "d",
"г": "u",
"д": "l",
"е": "t",
"ж": "|",
"з": "p",
"и": "b",
"й": "q",
"к": "r",
"л": "k",
"м": "v",
"н": "y",
"о": "j",
```

```
"п": "g",
"р": "h",
"с": "c",
"т": "n",
"у": "e",
"ф": "a",
"х": "{",
"ц": "w",
"ч": "x",
"ш": "i",
"щ": "o",
"ъ": "}",
"ы": "s",
"ь": "m",
"э": "`",
"ю": "/u007F",
"я": "z"
])
def en = new StringEncoder(controlSequence: "/u001B(2", charMap: [
    "A": "A",
    "B": "B",
    "C": "C",
    "D": "D",
    "E": "E",
    "F": "F",
    "G": "G",
    "H": "H",
    "I": "I",
    "J": "J",
    "K": "K",
    "L": "L",
    "M": "M",
    "N": "N",
    "O": "O",
    "P": "P",
    "Q": "Q",
    "R": "R",
    "S": "S",
    "T": "T",
    "U": "U",
    "V": "V",
    "W": "W",
    "X": "X",
    "Y": "Y",
    "Z": "Z",
    "a": "A",
    "b": "B",
    "c": "C",
    "d": "D",
    "e": "E",
    "f": "F",
    "g": "G",
    "h": "H",
    "i": "I",
    "j": "J",
    "k": "K",
```

```

"l": "L",
"m": "M",
"n": "N",
"o": "O",
"p": "P",
"q": "Q",
"r": "R",
"s": "S",
"t": "T",
"u": "U",
"v": "V",
"w": "W",
"x": "X",
"y": "Y",
"z": "Z",
"Ё": "t",
"ё": "t",
"А": "f",
"Б": "~",
"В": "d",
"Г": "u",
"Д": "l",
"Е": "t",
"Ж": "|",
"З": "p",
"И": "b",
"Й": "q",
"К": "r",
"Л": "k",
"М": "v",
"Н": "y",
"О": "j",
"П": "g",
"Р": "h",
"С": "c",
"Т": "n",
"У": "e",
"Ф": "a",
"Х": "{",
"Ц": "w",
"Ч": "x",
"Ш": "i",
"Щ": "o",
"Ъ": "}",
"Ы": "s",
"Ь": "m",
"Э": "`",
"Ю": "/u007F",
"Я": "z",
"а": "f",
"б": "~",
"в": "d",
"г": "u",
"д": "l",
"е": "t",
"ж": "|",

```



```
        "з": "p",
        "и": "b",
        "й": "q",
        "к": "r",
        "л": "k",
        "м": "v",
        "н": "y",
        "о": "j",
        "п": "g",
        "р": "h",
        "с": "c",
        "т": "n",
        "у": "e",
        "ф": "a",
        "х": "{",
        "ц": "w",
        "ч": "x",
        "ш": "i",
        "щ": "o",
        "ъ": "}",
        "ы": "s",
        "ь": "m",
        "э": "`",
        "ю": "/u007F",
        "я": "z"
    ])
    def en2 = new StringEncoder(controlSequence: "/u001B(I", charMap: [
        "A": "A",
        "B": "B",
        "C": "C",
        "D": "D",
        "E": "E",
        "F": "F",
        "G": "G",
        "H": "H",
        "I": "I",
        "J": "J",
        "K": "K",
        "L": "L",
        "M": "M",
        "N": "N",
        "O": "O",
        "P": "P",
        "Q": "Q",
        "R": "R",
        "S": "S",
        "T": "T",
        "U": "U",
        "V": "V",
        "W": "W",
        "X": "X",
        "Y": "Y",
        "Z": "Z",
        "a": "A",
        "b": "B",
        "c": "C",
```

```
"d": "D",  
"e": "E",  
"f": "F",  
"g": "G",  
"h": "H",  
"i": "I",  
"j": "J",  
"k": "K",  
"l": "L",  
"m": "M",  
"n": "N",  
"o": "O",  
"p": "P",  
"q": "Q",  
"r": "R",  
"s": "S",  
"t": "T",  
"u": "U",  
"v": "V",  
"w": "W",  
"x": "X",  
"y": "Y",  
"z": "Z",  
"Ё": "t",  
"ё": "t",  
"А": "f",  
"Б": "~",  
"В": "d",  
"Г": "u",  
"Д": "l",  
"Е": "t",  
"Ж": "|",  
"З": "p",  
"И": "b",  
"Й": "q",  
"К": "r",  
"Л": "k",  
"М": "v",  
"Н": "y",  
"О": "j",  
"П": "g",  
"Р": "h",  
"С": "c",  
"Т": "n",  
"У": "e",  
"Ф": "a",  
"Х": "{",  
"Ц": "w",  
"Ч": "x",  
"Ш": "i",  
"Щ": "o",  
"Ъ": "}",  
"Ы": "s",  
"Ь": "m",  
"Э": "`",  
"Ю": "/u007F",
```

```
"я": "z",
"a": "f",
"б": "~",
"в": "d",
"г": "u",
"д": "l",
"е": "t",
"ж": "|",
"з": "p",
"и": "b",
"й": "q",
"к": "r",
"л": "k",
"м": "v",
"н": "y",
"о": "j",
"п": "g",
"р": "h",
"с": "c",
"т": "n",
"у": "e",
"ф": "a",
"х": "{",
"ц": "w",
"ч": "x",
"ш": "i",
"щ": "o",
"ъ": "}",
"ы": "s",
"ь": "m",
"э": "`",
"ю": "/u007F",
"я": "z"
])
def all = new StringEncoder(charMap: [
    "A": "A",
    "B": "B",
    "C": "C",
    "D": "D",
    "E": "E",
    "F": "F",
    "G": "G",
    "H": "H",
    "I": "I",
    "J": "J",
    "K": "K",
    "L": "L",
    "M": "M",
    "N": "N",
    "O": "O",
    "P": "P",
    "Q": "Q",
    "R": "R",
    "S": "S",
    "T": "T",
    "U": "U",
```

```
"V": "V",  
"W": "W",  
"X": "X",  
"Y": "Y",  
"Z": "Z",  
"a": "A",  
"b": "B",  
"c": "C",  
"d": "D",  
"e": "E",  
"f": "F",  
"g": "G",  
"h": "H",  
"i": "I",  
"j": "J",  
"k": "K",  
"l": "L",  
"m": "M",  
"n": "N",  
"o": "O",  
"p": "P",  
"q": "Q",  
"r": "R",  
"s": "S",  
"t": "T",  
"u": "U",  
"v": "V",  
"w": "W",  
"x": "X",  
"y": "Y",  
"z": "Z",  
"Ё": "t",  
"А": "f",  
"Б": "~",  
"В": "d",  
"Г": "u",  
"Д": "l",  
"Е": "t",  
"Ж": "|",  
"З": "p",  
"И": "b",  
"Й": "q",  
"К": "r",  
"Л": "k",  
"М": "v",  
"Н": "y",  
"О": "j",  
"П": "g",  
"Р": "h",  
"С": "c",  
"Т": "n",  
"У": "e",  
"Ф": "a",  
"Х": "{",  
"Ц": "w",  
"Ч": "x",
```

```
"ш": "i",
"щ": "o",
"ъ": "j",
"ы": "s",
"ь": "m",
"э": "`",
"ю": "/u007F",
"я": "z",
"a": "f",
"б": "~",
"в": "d",
"г": "u",
"д": "l",
"е": "t",
"ж": "|",
"з": "p",
"и": "b",
"й": "q",
"к": "r",
"л": "k",
"м": "v",
"н": "y",
"о": "j",
"п": "g",
"р": "h",
"с": "c",
"т": "n",
"у": "e",
"ф": "a",
"х": "{",
"ц": "w",
"ч": "x",
"ш": "i",
"щ": "o",
"ъ": "j",
"ы": "s",
"ь": "m",
"э": "`",
"ю": "/u007F",
"я": "z",
"ё": "t",
])
def encoder1 = new Encoder(type: EncoderType.PRINTER, vendor: "NCR").add(en).add
(rus).add(en2).add(all)
def encoder2 = new Encoder(type: EncoderType.SCREEN, vendor: "NCR").add(all)
def encoder3 = new Encoder(type: EncoderType.PRINTER, vendor: "NCR3G").add(en).a
dd(rus).add(en2).add(all)
// load to encoder store
addEncoder(encoder1)
addEncoder(encoder2)
addEncoder(encoder3)
}
```

3.3.2 Тестирование шаблонов

Перед использованием новых шаблонов чеков и экранов рекомендуется выполнить их проверку. Это можно сделать вызовом команды `testTemplate` в консоли управления (см. документ "Веб-консоль управления WAY4-приложениями"). В качестве параметра команды необходимо указать путь к предварительно подготовленному xml-файлу с данными для подстановки в шаблон. В результате выполнения команды при корректной работе шаблона выводится текст, сформированный на основе файла данных, в противном случае - сообщение об ошибке.

Пример xml-файла с данными для тестирования шаблона (наименование может быть произвольным, например, `template_test_data.xml`):

```

<TemplateData _sys_type_id="atm.TemplateData" _ver="0" applicationId="app_id"
applicationLabel="VISA" authCode="987654" biller="biller" card="1000040163547882"
card2="1000040163547883" cardName="CARD NAME" cardsRetained="11" cashOut="true"
cashOut2="true" collectionCycle="11" date="2019-02-26 13:04:00.0" dispenserId="1"
encoderType="PRINTER" lang="RU" maxSize="0" numRC="0" operation="CASH_WITHDRAWAL"
password="password" protocol="NDC" rc="00" replCycle="9" replCycle2="11" rrn="111222
333" srn="J05732HN0TD2" stan="012345" templateName="admin_dual" terminalId="ATM00000
1" textDetails="text details" transactionDescription="tran description" userId="user
_id" vendor="NCR">
  <additionalBal _sys_type_id="_al">
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-977320">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="1000.00"
amountType="01">
        <currency>EUR</currency>
      </amount>
    </item>
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-977320">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="1000.00"
amountType="02">
        <currency>EUR</currency>
      </amount>
    </item>
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-848060">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="1000.00"
amountType="01">
        <currency>USD</currency>
      </amount>
    </item>
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-848060">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="1000.00"
amountType="02">
        <currency>USD</currency>
      </amount>
    </item>
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-411950">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="1000.00"
amountType="01">
        <currency>RUR</currency>
      </amount>
    </item>
    <item _sys_type_id="atm.AdditionalBalance" _ver="0" accountName="001-
P-411950">
      <amount _sys_type_id="ext_amnt" _ver="0" accountType="00" amount="-2000.00"
amountType="02">
        <currency>RUR</currency>
      </amount>
    </item>
  </additionalBal>
</amounts _sys_type_id="_al">

```

```

    <item _sys_type_id="ext_amnt" _ver="0" accountType="01" amount="100.00"
amountType="02">
      <currency>RUR</currency>
    </item>
  </amounts>
  <cashinData _sys_type_id="_al">
    <item _sys_type_id="atm.BNARec" _ver="0">
      <amount _ver="0" amount="150.00">
        <currency>RUR</currency>
      </amount>
      <notes _sys_type_id="_al">
        <item _sys_type_id="atm.BNANote" _ver="0" count="1" denomination="50"/>
        <item _sys_type_id="atm.BNANote" _ver="0" count="1" denomination="100"/>
      </notes>
    </item>
    <item _sys_type_id="atm.BNARec" _ver="0">
      <amount _ver="0" amount="2000.00">
        <currency>RUR</currency>
      </amount>
      <notes _sys_type_id="_al">
        <item _sys_type_id="atm.BNANote" _ver="0" count="1" denomination="1000"/
>
        <item _sys_type_id="atm.BNANote" _ver="0" count="2" denomination="500"/>
      </notes>
    </item>
  </cashinData>
  <cassetteAmounts _sys_type_id="_al">
    <item _ver="0" amount="750.00">
      <currency>RUR</currency>
    </item>
    <item _ver="0" amount="2000.00">
      <currency>RUR</currency>
    </item>
    <item _ver="0" amount="250000.00">
      <currency>RUR</currency>
    </item>
    <item _ver="0" amount="100000.00">
      <currency>RUR</currency>
    </item>
  </cassetteAmounts>
  <cassetteAmounts2 _sys_type_id="_al">
    <item _ver="0" amount="5.00">
      <currency>USD</currency>
    </item>
    <item _ver="0" amount="20.00">
      <currency>USD</currency>
    </item>
    <item _ver="0" amount="150.00">
      <currency>USD</currency>
    </item>
    <item _ver="0" amount="400.00">
      <currency>USD</currency>
    </item>
  </cassetteAmounts2>
  <cassetteTotals _sys_type_id="_al">
    <item _ver="0" amount="352750.00">

```



```

        <currency>RUR</currency>
    </item>
</cassetteTotals>
<cassetteTotals2 _sys_type_id="_al">
    <item _ver="0" amount="575.00">
        <currency>USD</currency>
    </item>
</cassetteTotals2>
<cassettes _sys_type_id="_al">
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="9" denomination="1"
        deposited="0" dispensed="15" diverted1="0" diverted2="0"
lastTransactionDeposited="0" lastTransactionDispensed="0"
        loaded="1000" name="DBP1" position="1" remaining="0" retracted="0">
        <denominationValue _ver="0" amount="50.00">
            <currency>RUR</currency>
        </denominationValue>
        <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
            <penalty _sys_type_id="_hm"/>
        </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="9" denomination="2"
        deposited="0" dispensed="20" diverted1="0" diverted2="0"
lastTransactionDeposited="0" lastTransactionDispensed="0"
        loaded="1000" name="DBP2" position="2" remaining="0" retracted="0">
        <denominationValue _ver="0" amount="100.00">
            <currency>RUR</currency>
        </denominationValue>
        <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
            <penalty _sys_type_id="_hm"/>
        </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="9" denomination="3"
        deposited="0" dispensed="50" diverted1="0" diverted2="0"
lastTransactionDeposited="0" lastTransactionDispensed="0"
        loaded="1000" name="DBP3" position="3" remaining="0" retracted="0">
        <denominationValue _ver="0" amount="5000.00">
            <currency>RUR</currency>
        </denominationValue>
        <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
            <penalty _sys_type_id="_hm"/>
        </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="9" denomination="4"
        deposited="0" dispensed="100" diverted1="0" diverted2="0"
lastTransactionDeposited="0" lastTransactionDispensed="0"
        loaded="2000" name="DBP4" position="4" remaining="0" retracted="0">
        <denominationValue _ver="0" amount="1000.00">
            <currency>RUR</currency>
        </denominationValue>
        <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
            <penalty _sys_type_id="_hm"/>
        </dispenseLimit>
    </item>

```

```

    </item>
  </cassettes>
  <cassettes2 _sys_type_id="_al">
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="11"
      denomination="1" deposited="0" dispensed="1" diverted1="0" diverted2="0"
" lastTransactionDeposited="0"
      lastTransactionDispensed="0" loaded="1000" name="DI02DBP1" position="1"
remaining="0" retracted="0">
      <denominationValue _ver="0" amount="5.00">
        <currency>USD</currency>
      </denominationValue>
      <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
        <penalty _sys_type_id="_hm"/>
      </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="11"
      denomination="2" deposited="0" dispensed="2" diverted1="0" diverted2="0"
lastTransactionDeposited="0"
      lastTransactionDispensed="0" loaded="1000" name="DI02DBP2" position="2"
remaining="0" retracted="0">
      <denominationValue _ver="0" amount="10.00">
        <currency>USD</currency>
      </denominationValue>
      <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
        <penalty _sys_type_id="_hm"/>
      </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="11"
      denomination="3" deposited="0" dispensed="3" diverted1="0" diverted2="0"
lastTransactionDeposited="0"
      lastTransactionDispensed="0" loaded="1000" name="DI02DBP3" position="3"
remaining="0" retracted="0">
      <denominationValue _ver="0" amount="50.00">
        <currency>USD</currency>
      </denominationValue>
      <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
        <penalty _sys_type_id="_hm"/>
      </dispenseLimit>
    </item>
    <item _sys_type_id="atm.Cassette" _ver="0" checkDispensed="0" checkLoaded="0"
checkRetracted="0" cycleNumber="11"
      denomination="4" deposited="0" dispensed="4" diverted1="0" diverted2="0"
lastTransactionDeposited="0"
      lastTransactionDispensed="0" loaded="2000" name="DI02DBP4" position="4"
remaining="0" retracted="0">
      <denominationValue _ver="0" amount="100.00">
        <currency>USD</currency>
      </denominationValue>
      <dispenseLimit _sys_type_id="atm.DispenseLimit" _ver="0" limit="40">
        <penalty _sys_type_id="_hm"/>
      </dispenseLimit>
    </item>
  </cassettes2>

```

```

<dispense _sys_type_id="_al">
  <item _sys_type_id="atm.DispenseValue" _ver="0" amount="10000" count="1"
denomination="1" denominationValue="10000"
  dispenserId="DBP1" limit="0" maxCount="0" penalty="0" penaltyIncrease="0
" positionId="1"/>
</dispense>
<fee _ver="0" amount="1000">
  <currency>RUR</currency>
</fee>
<fee2 _ver="0" amount="1000">
  <currency>RUR</currency>
</fee2>
<info _sys_type_id="_hm">
  <entry key="info1">test</entry>
</info>
<location _ver="0" city="City">
  <country>RUS</country>
</location>
<maxAvailAmounts _sys_type_id="_al">
  <item _ver="0" amount="200000.00">
    <currency>RUR</currency>
  </item>
  <item _ver="0" amount="4000.00">
    <currency>USD</currency>
  </item>
</maxAvailAmounts>
<maxDenomination _ver="0" amount="5000.00">
  <currency>RUR</currency>
</maxDenomination>
<minDenomination _ver="0" amount="50.00">
  <currency>RUR</currency>
</minDenomination>
<otplList _sys_type_id="_al">
  <item>111111</item>
  <item>222222</item>
</otplList>
<params _sys_type_id="_hm">
  <entry key="param1">value1</entry>
</params>
<request _ver="0" amount="1000">
  <currency>RUR</currency>
</request>
<settlement _ver="0" amount="1000">
  <currency>RUR</currency>
</settlement>
<settlement2 _ver="0" amount="1000">
  <currency>RUR</currency>
</settlement2>
<statement _sys_type_id="_al">
  <item _sys_type_id="atm.miniStatementProxy" _ver="0" date="2013-10-08
00:00:00.0" operationType="A">
    <amount _ver="0" amount="-13.00">
      <currency>RUR</currency>
    </amount>
  </item>

```

```

<item _sys_type_id="atm.miniStatementProxy" _ver="0" date="2013-10-08
00:00:00.0" operationType="A">
  <amount _ver="0" amount="-34.00">
    <currency>RUR</currency>
  </amount>
</item>
<item _sys_type_id="atm.miniStatementProxy" _ver="0" date="2013-10-08
00:00:00.0" operationType="A">
  <amount _ver="0" amount="-25.00">
    <currency>RUR</currency>
  </amount>
</item>
<item _sys_type_id="atm.miniStatementProxy" _ver="0" date="2013-10-08
00:00:00.0" operationType="A">
  <amount _ver="0" amount="-17.00">
    <currency>RUR</currency>
  </amount>
</item>
</statement>
</TemplateData>

```

В данном примере заполнена большая часть полей, которые могут присутствовать в шаблонах. Для тестирования часть полей в файле можно не заполнять.

Служебные поля в тестовом xml-файле (не используются в шаблонах):

- `templateName` - имя шаблона для компиляции;
- `encoderType` - (PRINTER/SCREEN) определяет какой тип перекодировки будет применен при компиляции шаблона. Если поле не заполнено - перекодирование выходного текста не выполняется.
- `maxSize` - максимальный размер выходного текста после компиляции и перекодирования в символах. Если поле не заполнено или содержит 0 - текст не обрезается.
- `protocol` - (NDC/DDC) определяет какой тип перекодирования будет применен при компиляции шаблона. Если поле не заполнено - перекодирование выходного текста не выполняется.
- `vendor` - тип банкомата, например NCR3G или NCR. Поле определяет тип перекодирования при компиляции шаблона. Если поле не заполнено - перекодирование выходного текста не выполняется.

Пример содержимого для тестового xml-файла также можно получить из log-файла АТМ-контроллера, при ведении протокола с уровнем логирования 60.

4. Схема расчетов по операциям банкомата

В процессе работы банкоматов (прием/выдача наличных, выполнение операций по инкассации) в системе WAY4 формируются проводки, отражающие движение денежных средств.

Типы счетов, между которыми осуществляются проводки, определяются Схемой Счетов контракта банкомата. Модуль эквайринга содержит эталонную Схему Счетов "001-Default ATM Scheme" (см. Схемы Счетов эквайринговых продуктов в пункте меню "Full → Configuration Setup → Products → Acquiring Products → Acquiring Account Schemes"), устанавливающую взаимосвязь между счетами следующих типов:

- "ATM Cassette" – тип счета "Деньги в пути"; используется для счетов, отражающих сумму денежных средств:
 - в кассетах, выданных инкассаторам из кассы банка для последующей загрузки в банкомат;
 - в кассетах, изъятых инкассаторами из банкомата для последующей передачи в кассу банка.
- "Cash Dispenser" – тип счета "Касса банкомата"; используется для счетов, отражающих сумму денежных средств в банкомате в период между загрузкой и разгрузкой банкомата;
Если предполагается поддержка операций приема и выдачи наличных через банкомат, то в Схеме Счетов должны использоваться два отдельных типа счета, например, "Cash Dispenser In" и "Cash Dispenser Out" для отражения принятых и доступных для выдачи банкоматом средств соответственно.
- "Merchant Receivable" – тип счета "Счет оплат"; используется для счетов, отражающих сумму операций, выполненных клиентами через банкомат в течение операционного дня.
Если предполагается поддержка операций приема и выдачи наличных через банкомат, то в Схеме Счетов должны использоваться два отдельных типа счета, например, "Merchant Receivable In" и "Merchant Receivable Out" для отражения принятых и выданных клиентам средств в течение операционного дня.

Между типами счетов "Cash Dispenser" и "Merchant Receivable" настроена срочная нормализация, определяющая необходимость при открытии нового операционного дня отражать накопленную сумму выданных/принятых средств на счете типа "Касса банкомата".

Проводки, отражающие движение средств между счетами в процессе работы и обслуживания банкоматов, формируются двумя способами:

- в "ручном" режиме – при обработке финансовых документов, созданных оператором (например, через форму работы с документами "Doc - General");
- в "автоматическом" режиме – при обработке финансовых документов, созданных в результате:
 - срочной нормализации, настроенной на Схеме Счетов контракта банкомата;
 - выполнения операций инкассации (загрузка/разгрузка банкомата).

Ниже представлена схема расчетов по операциям банкомата, основанная на эталонных настройках модуля эквайринга и включающая следующие проводки:

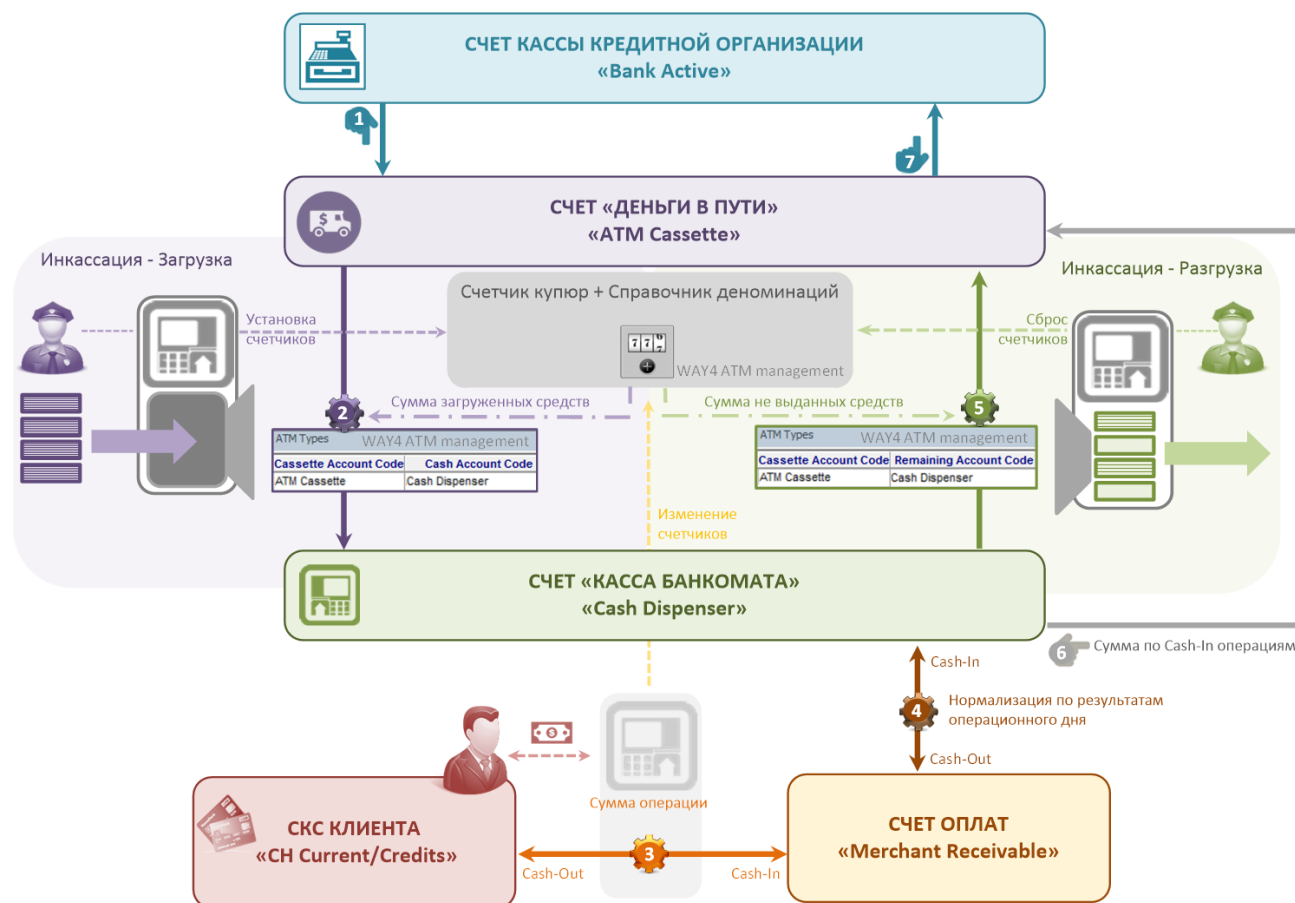


Схема расчетов по операциям банкомата

Шаг 1. Оператор в "ручном" режиме создает документ по транзакции типа "ATM cash replenishment" (см. перечень типов транзакций в пункте меню "Full → Configuration Setup → Transaction Types → Transactions - All"). При обработке данного документа формируется проводка между счетами следующих типов:

- Dt "ATM Cassette" Ct "Bank Active"

на сумму средств, выданных инкассаторам для последующей загрузки в банкомат.



Здесь и далее – процесс обработки документов может запускаться несколько раз в день, и не привязан к процессу открытия банковского дня.

Шаг 2. После загрузки кассет в банкомат инкассатор, используя сервисную карту, выполняет операцию ATM_SERVICE. При этом в меню банкомата вводится информация о количестве банкнот, загруженных в каждой кассете. На основе указанных данных в системе WAY4 устанавливаются кассетные счетчики купюр.

На каждую кассету формируется отдельный документ, сумма которого определяется на основании указанного количества купюр и справочника деноминаций (см. "[Перечень деноминаций \(ATM Denominations\)](#)"). При обработке данных документов формируются проводки

между счетами, типы которых заданы в полях *Cassette Account Code* и *Cash Account Code* для соответствующего типа банкомата (см. "[Перечень типов банкоматов \(ATM Types\)](#)"):

- Dt "Cash Dispenser" Ct "ATM Cassette"

Шаг 3. При успешной операции снятия/внесения наличных в БД WAY4 создается финансовый документ, при обработке которого формируется проводка между счетом клиента и Счетом оплат:

- Dt "CH Current/Credits" Ct "Merchant Receivable" – для схемы расчетов по снятию наличных;
- Dt "Merchant Receivable" Ct "CH Current/Credits" – для схемы расчетов по внесению наличных.



В качестве примера выбран тип счета "CH Current/Credits", используемый для расчетов с "собственными" клиентами банка.

Шаг 4. Процедура открытия нового банковского дня осуществляет срочную нормализацию между Счетом оплат и Кассой банкомата на сумму операций, выполненных в течение предыдущего операционного дня:

- Dt "Merchant Receivable" Ct "Cash Dispenser" – для схемы расчетов по снятию наличных;
- Dt "Cash Dispenser" Ct "Merchant Receivable" – для схемы расчетов по внесению наличных.



Для отражения оборотов средств по выдаче и приему наличных на счете типа "Merchant Receivable" в рамках финансового цикла банкомата может быть использован тег "ENTRY_GROUPING". Для этого необходимо в соответствующей Схеме Счетов (пункт меню "Full → Configuration Setup → Products → Acquiring Products → Acquiring Account Schemes") для счета типа "Merchant Receivable" в поле *TemplateDetails* указать значение "ENTRY_GROUPING=BY_BATCH";.

Шаг 5. После выемки кассет из банкомата инкассатор, используя сервисную карту, выполняет операцию REPLENISHMENT. В результате выполнения данной операции в системе WAY4 для каждой кассеты формируется отдельный документ на сумму не выданных средств (определяется на основании текущих значений счетчиков купюр и справочника деноминаций в БД WAY4). При обработке данных документов формируются проводки между счетами, типы которых заданы в полях *Cassette Account Code* и *Remaining Account Code* для соответствующего типа банкомата (см. "[Перечень типов банкоматов \(ATM Types\)](#)"):

- Dt "ATM Cassette" Ct "Cash Dispenser"

Шаг 6. Для схемы расчетов по внесению наличных оператор в "ручном" режиме создает финансовые документы для изъятых инкассаторами кассет с принятыми купюрами. При обработке данных документов формируются проводки между счетами:

- Dt "ATM Cassette" Ct "Cash Dispenser"

Шаг 7. При получении от инкассаторов средств, изъятых в результате разгрузки банкомата, оператор в "ручном" режиме создает документ по транзакции типа "ATM residual cash collection" (см. перечень типов транзакций в пункте меню "Full → Configuration Setup → Products

→ Acquiring Products → Acquiring Account Schemes"). При обработке данного документа формируется проводка между счетами следующих типов:

- Dt "Bank Active" Ct "ATM Cassette"

В случае выявления расхождений, корректирующие проводки (на сумму излишков/недостач) формируются оператором в "ручном" режиме.

5. Автоматическое формирование контроллером сообщения об отмене операции (Reversal)

Процедура обмена сообщениями при выполнении операции на банкомате представлена на рисунке.



Обмен сообщениями при выполнении операции на банкомате

Контроллер банкоматов автоматически формирует и передает в канал авторизации сообщение об отмене операции в следующих четырех случаях:

- На этапе №4 выполнения операции: при отсутствии ответа канала авторизации в течение заданного времени (50 сек).
- На этапе №5 выполнения операции: при обнаружении событий технического характера, указывающих на факт невыполнения операции.
- На этапе №6 выполнения операции : в случае обнаружения нехватки, отсутствия или искажения данных в поле №39 сообщения стандарта ISO 8583, а также при невозможности сформировать электронную подпись (MAC) сообщения, например, из-за отсутствия связи с аппаратным модулем безопасности (HSM).
- На этапе №8 выполнения операции: в случае получения соответствующего статусного сообщения, например, о том, что набранная сумма не была выдана держателю банковской карты.

Контроллер банкоматов позволяет настроить коды ответов (Response Code, RC), которые будут использоваться в сообщениях об отмене. Правила, определяющие данные коды, задаются в скрипте CustomReversals.groovy, который должен быть помещен в каталог WEB-INF/inv/atm/scripts. Для настройки правил используется объект CustomReversal, содержащий следующие параметры:

- ReversalType – тип отмены. Возможные значения:
 - TIMEOUT_REVERSAL
 - CANCELLATION
 - UNEXPECTED_ERROR_REVERSAL
 - REVERSAL_WITH_ADJUST
 - CANCELLATION_WITH_ADJUST
 - FULL_REVERSAL,
 - PARTIAL_REVERSAL
- OperationType – тип отменяемой операции (необязательный параметр).
- Channel – код канала взаимодействия с платежной системой карты (необязательный параметр).
- OperStatus – код ответа. Возможные значения OperStatus, используемые контроллером банкоматов, и соответствующие им RC:

OperStatus	RC	Description
SUCCESS	0	Successfully completed
SUCCESS_BY_SAF	0	Successfully completed by SAF
REFER_TO_CARD_ISSUER	1	Refer to card issuer
REFER_TO_CARD_ISSUERS_SPECIAL_CONDITION	2	Refer to card issuer's special condition
INVALID_MERCHANT_SOURCE	3	Invalid merchant / source
PICK_UP	4	PICK UP
DO_NOT_HONOUR	5	Do not Honour
OP_ERROR	6	Error
PICK_UP_CARD_SPECIAL_CONDITION	7	Pick-up card, special condition

OperStatus	RC	Description
HONOUR_WITH_IDENTIFICATION	8	Honour with identification
REQUEST_IN_PROGRESS	9	Request in progress
APPROVED_FOR_PARTIAL_AMOUNT	10	Approved for partial amount
APPROVED_VIP	11	Approved (VIP)
INVALID_TRANSACTION	12	Invalid transaction
INVALID_AMOUNT	13	Invalid amount
NO_SUCH_CARD	14	No such card
NO_SUCH_ISSUER	15	No such issuer
APPROVED_UPDATE_TRACK_3	16	Approved, update track 3
CUSTOMER_CANCELLATION	17	Customer cancellation
CUSTOMER_DISPUTE	18	Customer dispute
RE_ENTER_TRANSACTION	19	Re-enter transaction
INVALID_RESPONSE	20	Invalid response
NO_ACTION_TAKEN	21	No action taken
SUSPECTED_MALFUNCTION	22	Suspected malfunction
UNACCEPTABLE_TRANSACTION_FEE	23	Unacceptable transaction fee
FILE_UPDATE_NOT_SUPPORTED_BY_RECEIVER	24	File update not supported by receiver
NO_SUCH_RECORD	25	No such record
DUPLICATE_RECORD_UPDATE_OLD_RECORD_REPLACED	26	Duplicate record update, old record replaced

OperStatus	RC	Description
FILE_UPDATE_FIELD_EDIT_ERROR	27	File update field edit error
FILE_LOCKED_OUT_WHILE_UPDATE	28	File locked out while update
FILE_UPDATE_ERROR_CONTACT_ACQUIRER	29	File update error, contact acquirer
FORMAT_ERROR	30	Format error
ISSUER_SIGNED_OFF	31	Issuer signed-off
COMPLETED_PARTIALLY	32	Completed partially
PICK_UP_EXPIRED_CARD	33	Pick-up, expired card
SUSPECT_FRAUD	34	Suspect Fraud
PICK_UP_CARD_ACCEPTOR_CONTACT_ACQUIRER	35	Pick-up, card acceptor contact acquirer
PICK_UP_CARD_RESTRICTED	36	Pick up, card restricted
PICK_UP_CALL_ACQUIRER_SECURITY	37	Pick up, call acquirer security
PICK_UP_ALLOWABLE_PIN_TRIES_EXCEEDED	38	Pick up, Allowable PIN tries exceeded
NO_CREDIT_ACCOUNT	39	No credit account
REQUESTED_FUNCTION_NOT_SUPPORTED	40	Requested function not supported
PICK_UP_LOST_CARD	41	Pick up, lost card
NO_UNIVERSAL_ACCOUNT	42	No universal account
PICK_UP_STOLEN_CARD	43	Pick up, stolen card
NO_INVESTMENT_ACCOUNT	44	No investment account
DO_NOT_RENEW	50	Do not renew

OperStatus	RC	Description
NOT_SUFFICIENT_FUNDS	51	Not sufficient funds
NO_CHEQUING_ACCOUNT	52	No checking account
NO_SAVINGS_ACCOUNT	53	No savings account
EXPIRED_CARD	54	Expired card / target
INCORRECT_PIN	55	Incorrect PIN
NO_CARD_RECORD	56	No card record
TRANSACTION_NOT_PERMITTED	57	Transaction Not Permitted
TRANSACTION_NOT_PERMITTED_TO_TERMINAL	58	Transaction not permitted to terminal
SUSPECTED_FRAUD	59	Suspected fraud
CARD_ACCEPTOR_CONTACT_ACQUIRER	60	Card acceptor contact acquirer
EXCEEDS_WITHDRAWAL_AMOUNT_LIMIT	61	Exceeds withdrawal amount limit
RESTRICTED_CARD	62	Restricted card
SECURITY_VIOLATION	63	Security violation
WRONG_ORIGINAL_AMOUNT	64	Wrong original amount
EXCEEDS_WITHDRAWAL_FREQUENCY_LIMIT	65	Exceeds withdrawal frequency limit
CALL_ACQUIRERS_SECURITY_DEPARTMENT	66	Call acquirers security department
CARD_TO_BE_PICKED_UP_AT_ATM	67	Card to be picked up at ATM
RESPONSE_RECEIVED_TOO_LATE	68	Response received too late
INVALID_TRANSACTION_CONTACT_CARD_ISSUER	70	Invalid transaction; contact card issuer

OperStatus	RC	Description
DECLINE_PIN_NOT_CHANGED	71	Decline PIN not changed
ALLOWABLE_NUMBER_OF_PIN_TRIES_EXCEEDED	75	Allowed number of PIN tries exceeded
WRONG_PIN_NUMBER_OF_PIN_TRIES_EXCEEDED	76	Wrong PIN, number of PIN tries exceeded
WRONG_REFERENCE_NO	77	Wrong Reference No.
RECORD_NOT_FOUND	78	Record Not Found
ALREADY_REVERSED	79	Already reversed
NETWORK_ERROR	80	Network error
FOREIGN_NETWORK_ERROR_PIN_CRYPTOGRAPHIC_ERROR	81	Foreign network error / PIN cryptographic error
TIMEOUT	82	Time-out at issuer system
TRANSACTION_FAILED	83	Transaction failed
PRE_AUTHORIZATION_TIMED_OUT	84	Pre-authorization timed out
NO_REASON_TO_DECLINE	85	No Reason To Decline
UNABLE_TO_VALIDATE_PIN	86	Unable to validate PIN
PURCHASE_APPROVAL_ONLY	87	Purchase Approval Only
CRYPTOGRAPHIC_FAILURE	88	Cryptographic failure
AUTHENTICATION_FAILURE	89	Authentication failure
CUTOFF_IS_IN_PROGRESS	90	Cutoff is in progress
ISSUER_OR_SWITCH_IS_INOPERATIVE	91	Issuer or switch is inoperative
UNABLE_TO_ROUTE_AT_ACQUIRER_MODULE	92	Unable to route at acquirer module

OperStatus	RC	Description
CANNOT_BE_COMPLETED_VIOLATION_OF_LAW	93	Cannot be completed, violation of law
DUPLICATE_TRANSMISSION	94	Duplicate Transmission
RECONCILE_ERROR_AUTH_NOT_FOUND	95	Reconcile error / Auth Not found
SYSTEM_MALFUNCTION	96	System Malfunction
CREDIT_FAILED	145	Credit authorization declined
ROUTING_UNAVAILABLE	2015	HA: no available DB routes

Также существует возможность определить дополнительный код ответа с помощью объекта OperStatus, например следующим образом:

```
new OperStatus(OperStatus.ResponseClass.ERROR, 34L, "Fraud message")
```

Пример CustomReversals.groovy:

```
package com.openwaygroup.ts.atm.controller.handler.scripts

import com.openwaygroup.ts.atm.controller.handler.operations.CustomReversal
import com.openwaygroup.ts.atm.controller.handler.operations.ReversalType
import com.openwaygroup.ts.fp.shared.OperStatus
import com.openwaygroup.ts.fp.shared.atm.scripts.OperationType

[
    new CustomReversal(ReversalType.CANCELLATION_WITH_ADJUST,
        OperationType.PIN_CHANGE, "E", OperStatus.OP_ERROR),
    new CustomReversal(ReversalType.CANCELLATION_WITH_ADJUST, null, null,
        OperStatus.OP_ERROR)
    new CustomReversal(ReversalType.TIMEOUT_REVERSAL,
        OperationType.CARD_CTRL_REQUEST, "E", new OperStatus(OperStatus.ResponseClass.ERROR,
            168L, "Response received too late"))
]
```

Обработка списка выполняется в порядке следования правил, поэтому применено будет первое правило, соответствующее критериям текущей операции отмены.