

Техническое описание
Модуль тарификации INMan
Версия 0.7.11

Содержание

1. Введение.....	3
1.1 Цель.....	3
1.2 Область действия.....	3
1.3 Определения и сокращения.....	3
1.4 История изменений.....	3
1.1 Ссылки.....	5
2. Общее описание INMan-a.....	5
2.1 Описание требований.....	5
2.1.1 Интеграция с SMSC/USSDC.....	5
2.1.1 Взаимодействие с IN-платформой.....	5
2.1.1 Режимы тарификации.....	6
2.2 Ограничения и расхождения.....	6
2.1 Протоколы, используемые при тарификации SMS/USSD сообщений.....	6
2.1.1 Обзор протокола взаимодействия INMan-IN.....	7
2.1.2 Сценарий взаимодействия INMan – IN (оказание услуги разрешено).....	7
2.1.3 Сценарий взаимодействия INMan – IN (оказание услуги запрещено).....	7
2.1.4 Общий сценарий взаимодействия Отправитель – USSDC – INMan – IN - Получатель.....	8
2.2 Принципы тарификации.....	8
2.2.1 Понятие ‘политики абонентов’, ‘провайдер абонента’.....	8
2.2.1 Логика и режимы тарификации.....	9
2.2.1 Тарификация SMS Extra услуг.....	9
2.2.1 Генерация CDR записей.....	10
3. Протоколы взаимодействия клиентов с INMan-ом.....	10
3.1 Общий формат клиентских протоколов INMan-a.....	10
3.1.1 Используемые типы данных и их формат передачи.....	10
3.1 Обзор протокола тарификации SMS/USSD сообщений (USSDC-INMAN).....	11
3.1.1 Описание взаимодействия.....	11
3.1.1.1 Сценарий взаимодействия (оказание услуги разрешено).....	12
3.1.1.2 Сценарий взаимодействия (оказание услуги запрещено).....	12
3.1.1.3 Сценарий взаимодействия (протарифицировать уже оказанную услугу).....	12
3.1.2 Асинхронность протокола.....	12
3.1.3 Описание формата PDU.....	12
3.1 Структура PDU протокола тарификации SMS/USSD сообщений.....	13
3.1.1 Структура ChargeSms PDU (тэг 0x01).....	13
3.1.2 Структура ChargeSmsResult PDU (тэг 0x02).....	14
3.1.3 Структура DeliverySmsResult PDU (тэг 0x03).....	14
3.1.4 Структура DeliveredSmsData PDU (тэг 0x04).....	15
3.2 Обзор протокола определения контракта абонента (USSDC-INMAN).....	16
3.2.1 Асинхронность протокола.....	17
3.2.2 Описание формата PDU.....	17
3.1 Структура PDU протокола определения контракта абонента.....	17
3.1.1 Структура AbntContractRequest PDU (тэг 0x06).....	17
3.1.2 Структура AbntContractResult PDU (тэг 0x07).....	17
4. Структура CSV файла для генерации CDR записей.....	18
4.1 Типы и формат данных полей CSV файла.....	18
4.1 Поля данных CSV файла.....	18
5. Описание параметров файла конфигурации INMan-a.....	20
5.1 Конфигурируемые сервисы.....	20

5.1 Сводный список секций файла конфигурации.....	21
5.2 Структура секции Services.....	22
5.1 Параметры секции Host (сервис icsTCPSTServer).....	23
5.2 Параметры секции AbonentsCache (сервис icsAbntCache).....	23
5.3 Параметры секции SS7 (сервис icsTCAPDisp).....	23
5.4 Параметры секции SS7_HD (сервис icsTCAPDisp).....	23
5.4.1 Параметры подсекции SS7Units.....	24
5.4.2 Параметры подсекции Layouts.....	24
5.5 Шаблон секции TCAPUser.....	25
5.6 Параметры секции AbonentDetector (сервис icsAbntDetector).....	25
5.7 Параметры секции Billing (сервис icsSmBilling).....	26
5.7.1 Параметры подсекции BillingModes.....	27
5.8 Параметры секции AbonentPolicies (сервис icsIAPManager).....	28
5.1 Структура и параметры секции IN-Platforms (сервис icsIAPManager).....	29
5.1.1 Параметры конфигурации в формате 'параметры диалога'.....	29
5.1.1.1 Параметры подсекции ServiceKeys.....	30
5.1.1.1 Параметры конфигурации в формате 'alias-платформы'.....	31
5.2 Параметры секции AbonentProviders (сервис icsIAPManager).....	32
5.2.1 Параметры подсекции Config для провайдера icsIAPrvdSRI.....	32
1. Описание файла конфигурации SMS Extra услуг.....	33
2. Информация о компании.....	34
3. Контактная информация.....	34

1. Введение

1.1 Цель

В данном документе описывается предназначение, структура и логика работы модуля тарификации. Также приводится перечень требований, проектных ограничений и других аспектов, необходимых для наиболее полного и всестороннего понимания предлагаемого решения.

1.2 Область действия

Документ подготовлен в рамках проекта 23.10.2009 и предназначен для использования участниками и аудиторами данного проекта.

1.3 Определения и сокращения

CAP – CAMEL Application Part

MSC – Mobile Switching Centre

IN – Intelligent Network

USSD – Unstructured Supplementary Service Data

1.4 История изменений

Дата	Версия	Описание	Автор
23-Марта-2006	0.1.0	структура документа, описание взаимодействия	Виктор Рыжков
04-Апреля-2006	0.1.0	описание PDU протокола, параметров config.xml	Роман Герасимов
29-Мая-2006	0.4.1	поддержка внешних модулей AbonentProvider-a	Роман Герасимов
20-Сентября-2006	0.5.0	базовая поддержка “ комплексных политик абонентов ”	Роман Герасимов
26-Октября-2006	0.5.1	поддержка тарификации SMS Extra услуг.	Роман Герасимов
26-Октября-2006	0.5.2	изменения транспортного протокола SMSC-INMan	Роман Герасимов
06-Декабря-2006	0.5.4	добавлен протокол определения контракта абонента	Роман Герасимов
07-Декабря-2006	0.5.5	добавлен код ошибки INMan-a в AbntContractResult	Роман Герасимов
19-Декабря-2006	0.5.7	новый параметр IDPLocationInfo в конфигурации IN-платформ	Роман Герасимов
23-Января-2007	0.5.9	добавлен IMSI в AbntContractResult	Роман Герасимов
04-Апреля-2007	0.6.0	Изменена логика тарификации SMS Extra услуг, введен новый формат их файла конфигурации.	Роман Герасимов
10-Мая-2007	0.6.1	Добавлена строка описания кода ошибки в ChargeSmsResult и AbntContractResult	Роман Герасимов
15-Мая-2007	0.6.2	Введена возможность задать режим тарификации для каждого типа сообщения отдельно. Соответствующие изменения в секции Billing .	Роман Герасимов
17-Мая-2007	0.6.3	Новый параметр abonentTypeRequest в секции Billing . Описание CSV файла .	Роман Герасимов
18-Мая-2007	0.6.4	Введено понятие типа тарификации (ON_SUBMIT, ON_DELIVERY). Расширение протокола тарификации - новое PDU DeliveredSmsData	Роман Герасимов
22-Мая-2007	0.6.4	Скорректировано описание CSV файла	Роман Герасимов

28-Мая-2007	0.6.5	Поддержка вторичного режима тарификации в случае ошибки взаимодействия с IN. Соответствующие изменения в секции Billing.BillingModes файла конфигурации.	Роман Герасимов
31-Мая-2007	0.6.6	Поддержка МТ тарификации. Соответствующие изменения в PDU: ChargeSms и DeliveredSmsData	Роман Герасимов
04-Июня-2007	0.6.7	Введена возможность конфигурации режима тарификации не только в зависимости от типа сообщения но и от типа тарификации МО или МТ. Соответствующие изменения в секции BillingModes	Роман Герасимов
22-Июня-2007	0.6.10	Введена возможность конфигурации тарификации сообщения-носителя для SMS Extra сообщений. Изменения в формате файла конфигурации SMS Extra услуг .	Роман Герасимов
06-Июля-2007	0.6.11	В конфигурацию IN-платформ введены новые возможности: 1) задать алгоритм для вычисления сервис-ключей. 2) указать alias-платформу Изменены Параметры секции IN-Platforms .	Роман Герасимов
14-Сентября-2007	0.6.20	Введена поддержка IN-платформ фирмы Commverse. Изменены параметры диалога секции IN-Platforms : 1) новый параметр IDPReqMode 2) новый параметр RPCList_retry 3) исключен устаревший RPCList_postpaid	Роман Герасимов
11-Марта-2008	0.7.0	Добавлен тип контракта абонента в ChargeSmsResult . INMap переведен на технологию конфигурируемых сервисов. Изменена структура документа. Значительные изменения в параметрах файла конфигурации .	Роман Герасимов
16-Октября-2008	0.7.3	В протокол тарификации SMS/USSD сообщений добавлена возможность указания клиентом SMPP service type для сохранения его в записи CSV файла . Соответствующие изменения в PDU ChargeSms и DeliveredSmsData .	Роман Герасимов
13-Апреля-2009	0.7.7	Введена поддержка EIN HD SS7 стэка. Новая секция файла конфигурации SS7_HD .	Роман Герасимов
05-Мая-2009	0.7.8	Изменено задание адреса Comman Parts Manager-а в секции файла конфигурации SS7_HD .	Роман Герасимов
24-Сентября-2009	0.7.10	Добавления в протоколе тарификации: – введена новая политика тарификации ON_SUBMIT_COLLECTED – введена возможность принудительно задать режим тарификации CDR независимо от настроек в файле конфигурации, изменения в PDU ChargeSms и DeliveredSmsData	Роман Герасимов
23-Октября-2009	0.7.11	Исключена поддержка ORACLE DB в качестве провайдера абонентов . Провайдер абонентов использующий HLR сервис SEND_ROUTING_INFO реализован как отдельный конфигурируемый сервис. Изменения в секции файла конфигурации AbonentProviders	Роман Герасимов

1.1 Ссылки

3GPP TS 22.078 version 6.7.0 Release 6 - CAMEL Service description
3GPP TS 23.078 version 6.6.0 Release 6 - CAMEL Technical realization
3GPP TS 29.078 version 6.4.0 Release 6 - CAMEL CAP specification

2. Общее описание INMan-a

Операторы мобильной связи все шире предлагают свои услуги на базе предоплаченной модели оказания услуг, повсеместно происходит установка IN платформ. В этой связи возникает необходимость on-line тарификации услуг на базе USSD технологии. Однако, на текущий момент организация взаимодействия USSD центров или MSC и IN платформы стандартом не регламентируется. Именно этот пробел и восполняет *Модуль тарификации INMan*.

2.1 Описание требований

Эта секция перечисляет в табличной форме требования к модулю тарификации и описывает способы удовлетворения этим требованиям.

2.1.1 Интеграция с SMSC/USSDC.

Требование	Способ достижения
1. Модуль должен самостоятельно и в on-line режиме тарифицировать через IN платформу USSD сообщения, проходящие через Sibinco USSDC	Трактовать USSD сообщение как SMS сообщение. Эмулировать взаимодействие по протоколу CAP между MSC и IN при обработке SMS сообщения.
2. Модуль должен быть необязательным компонентом Sibinco USSDC	Реализовать модуль в виде внешнего приложения.
3. Модуль должен взаимодействовать не только с Sibinco USSDC, но и с любым другим клиентом, которому необходима тарификация USSD сообщений	Реализовать модуль в виде внешнего приложения, предоставляющего услугу тарификации по определенному протоколу взаимодействия.
4. Модуль должен обеспечивать удаленное взаимодействие с клиентами	Реализовать протокол с использованием TCP/IP транспорта
5. Протокол взаимодействия модуля и клиента должен быть простым и открытым	Предоставить описание команд, их формат и последовательность взаимодействия
6. Модуль должен обеспечивать одновременное подключение нескольких клиентов	Разрешить одновременное открытие нескольких транспортных соединений.
7. Модуль должен обеспечивать высокую пропускную способность обработки заявок на тарификацию	Реализовать асинхронный протокол общения с клиентами
8. Модуль должен снять с Sibinco USSDC обязанность по генерации CDR файлов	Реализовать поддержку генерации CDR файлов

2.1.1 Взаимодействие с IN-платформой.

Требование	Способ достижения
------------	-------------------

1.Протокол взаимодействия модуля с клиентами должен учитывать логику взаимодействия с IN платформой	Команды протокола должны следовать логике CAP SMS диалога .
2.Модуль должен взаимодействовать с IN платформами разных поставщиков	Обеспечить возможность внешней конфигурации параметров взаимодействия с разными IN платформами
3.Модуль должен взаимодействовать с несколькими IN платформами одновременно	Реализовать возможность одновременного запуска нескольких подмодулей для конкретных IN платформ
4.Модуль должен самостоятельно решать, с какой IN платформой взаимодействовать на основании MSISDN абонента	Реализовать возможность внешней конфигурации стратегии вычисления номера IN платформы и параметров для взаимодействия с ней на основании MSISDN. Например, AnyTimeSubscriptionInterrogation, запрос в БД.
5.Модуль не должен взаимодействовать с IN платформой для postpaid абонентов	Реализовать подмодуль вычисления типа контракта абонента. Например, AnyTimeSubscriptionInterrogation, запрос в БД.
6.Модуль не должен вычислять тип контракта абонента, каждый раз обращаясь к внешним системам	Реализовать кэширование типа контракта абонента
7.Модуль не должен заполнять кэш типов контрактов каждый раз при запуске	Кэш должен иметь возможность сохраняться на файловой системе.

2.1.1 Режимы тарификации.

Требование	Способ достижения
1.Модуль должен поддерживать выборочную тарификацию SMS и/или USSD сообщений	Обеспечить возможность внешней конфигурации режима тарификации.
2.Модуль должен поддерживать тарификацию сообщений, использующих дополнительные услуги (SMS Extra)	Обеспечить возможность внешней конфигурации дополнительных услуг (SMS Extra)
3.Модуль должен поддерживать раздельную тарификацию как посылки, так и доставки сообщений	Реализовать такую возможность в протоколе тарификации.

2.2 Ограничения и расхождения

- Динамическая реконфигурация не поддерживается.
- Для работы с элементами сигнальной сети требуется установленное TCAP API компании TietoEnator.

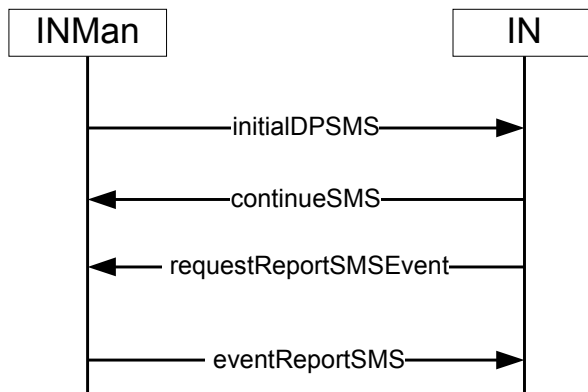
2.1 Протоколы, используемые при тарификации SMS/USSD сообщений.



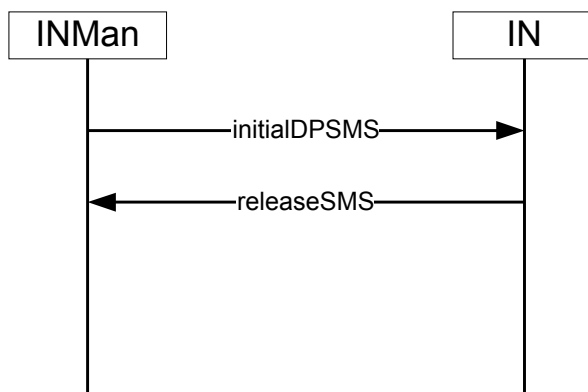
2.1.1 Обзор протокола взаимодействия INMan-IN

Для взаимодействия с IN платформой используется протокол CAP3 (CAMEL phase 3) в той его части, которая описывает взаимодействие MSC и IN платформы в момент обработки MO SMS. Детальное описание можно найти в спецификациях **3GPP TS 29.078**. Примерные сценарии взаимодействия описаны ниже.

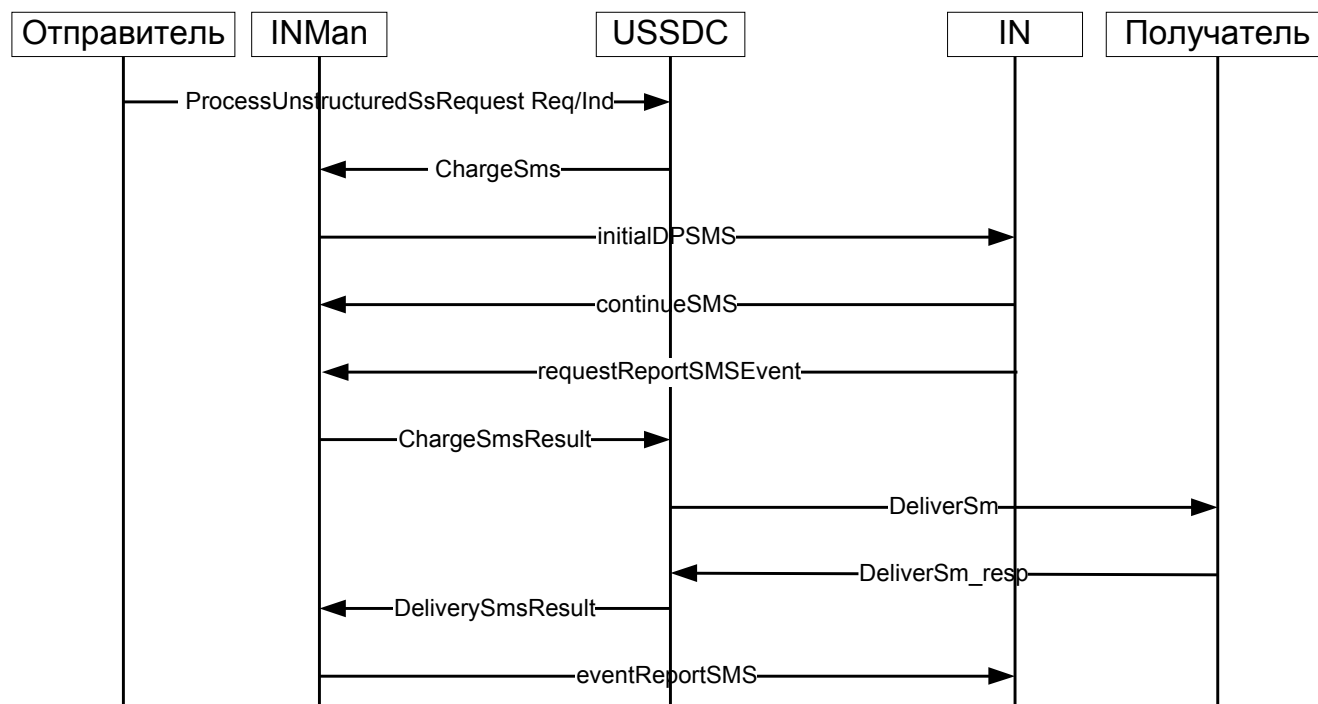
2.1.2 Сценарий взаимодействия INMan – IN (оказание услуги разрешено)



2.1.3 Сценарий взаимодействия INMan – IN (оказание услуги запрещено)



2.1.4 Общий сценарий взаимодействия Отправитель – USSDC – INMan – IN - Получатель



2.2 Принципы тарификации.

2.2.1 Понятие ‘политики абонентов’, ‘провайдер абонента’.

Для выполнения условий из пп. 10-13 [требований к функциональности](#) INman-а вводится понятие ‘политики абонента’, задающей стратегию определения контракта абонента по его MSISDN и определения параметров последующего диалога с IN-платформой если необходимо.

Логически, при тарификации услуги, действия INMan-а состоят из трех фаз:

1. обращение к некоему внешнему для INMan-а приложению/модулю, способному определить тип контракта абонента
2. в случае preraid абонента определение адреса IN-платформы и параметров для диалога с ней либо от внешнего приложения/модуля, либо из конфигурации INMan-а
3. диалог с IN-платформой

В терминах реализации INMan-а внешний модуль из пп.1 называется ‘провайдер абонента’. Его природа и функциональность может быть совершенно различной (база данных, MAP сервис HLR-а, etc), Поэтому подмодули INMan-а взаимодействующие с ‘провайдерами абонентов’ реализованы в виде динамических библиотек, конфигурируемых при загрузке. На данный момент в INMan-е реализована работа со следующими типами ‘провайдеров абонентов’:

- ORACLE база данных - возвращает только тип контракта абонента, **не поддерживается начиная с версии 0.7.11**
- HLR Сервис SEND_ROUTING_INFO – возвращает тип контракта абонента, а для preraid абонента также и ISDN адрес IN-платформы (без параметров для диалога)
- HLR Сервис ANY_TIME_SUBSCRIPTION_INTERROGATION - возвращает тип контракта абонента, а для preraid абонента также и ISDN адрес IN-платформы с параметрами для диалога

Как видно, не всегда можно получить всю необходимую информацию для полноценного диалога с IN-платформой от 'провайдера абонента'. Поэтому предусмотрена возможность задания минимально необходимых, а также дополнительных, в зависимости от производителя, параметров в файле конфигурации INMan-a.

Таким образом, 'политика абонента' состоит из:

- конфигурации для 'провайдера абонента'
- конфигурации для одной или нескольких IN-платформ

В общем случае 'политика абонентов' привязывается к определенному пулу адресов абонентов, позволяя гибко настроить стратегию тарификации для разных номерных серий.

2.2.1 Логика и режимы тарификации.

Для обеспечения гибкой системы тарификации в INMan-е введена возможность задания режима тарификации на каждый тип тарифицируемого сообщения.

Под понятием 'тип сообщения' подразумевается совокупность типа носителя (DP bearer type) и запрошенных особенностей тарификации (к примеру, использовать SMS Extra сервисы).

На данный момент различаются следующие типы тарифицируемых сообщений:

- SMS – обычный SMS пакет
- USSD – обычный пакет с данными USSD
- XSMS – SMS пакет использующий SMS Extra сервисы

На каждый тип сообщения можно задать следующие **режимы тарификации**:

- OFF – тарификация запрещена (по умолчанию),
- CDR – тарифицировать через создание CDR записей
- IN – тарифицировать через обращение к IN-платформе, в случае невозможности - использовать вторичный режим (CDR или OFF) в соответствии с заданными установками в файле конфигурации

Задание **режимов тарификации** производится в секции [Billing.BillingModes](#) файла конфигурации INMan-a. Если какой-либо тип сообщения не указан в конфигурации, то для него используется режим тарификации по умолчанию (OFF).

NOTE: SMSC имеет возможность запросить у INMan-a безусловную тарификацию через IN-платформу посредством поля smsXSrvsId в [ChargeSms PDU](#).

Изначально модуль INMan поддерживал только один **тип тарификации** - MO, то есть тарификацию отправителя сообщения, но начиная с версии 0.6.6 введена поддержка: также и **типа тарификации** MT, то есть тарификацию получателя. Требуемый **тип тарификации** задается соответствующими полями PDU протокола тарификации [ChargeSms](#) или [DeliveredSmsData](#).

Для обеспечения раздельной тарификации как отправки так и доставки сообщений поддерживаются следующие **политики тарификации**:

- ON_SUBMIT – тарификация в момент отправки сообщения с проверкой платежеспособности
- ON_DELIVERY – тарификация в момент доставки сообщения, с предварительной проверкой платежеспособности в момент отправки,
- ON_DATA_COLLECTED – тарификация уже после доставки сообщения без проверки платежеспособности (тарификация производится только через CDR).
- ON_SUBMIT_COLLECTED – тарификация в момент отправки сообщения без проверки платежеспособности (тарификация производится только через CDR).

2.2.1 Тарификация SMS Extra услуг.

Для обеспечения тарификации SMS Extra услуг, между SMSC/USSDC и INMan-ом вводится соглашение на идентификацию соответствующих сервисов посредством битовой маски, в которой каждому единичному сервису выделяется один бит. Со стороны SMSC значения этих битов жестко прописаны в исполняемом коде, однако INMan получает их из специального [файла конфигурации SMS Extra услуг](#).

В случае поддержки SMSC дополнительных услуг SMS Extra, он добавляет в запрос [CHARGE_SMS](#) маску пакета услуг для тарификации. Если [режим тарификации](#) INMan-а, заданный в его файле конфигурации, разрешает тарифицирование, то по этой маске INMan определяет из [файла конфигурации SMS Extra услуг](#) сервис-код и короткий номер, соответствующие пакету услуг. Тарификация базового SMS сообщения и пакета SMS Extra услуг проводится в зависимости от типа контракта абонента:

- для postpaid – создается CDR запись по базовому SMS сообщению, дополнительно содержащая сервис-код пакета услуг
- для prepaid – по базовому SMS ничего не делается, (подразумевается, что оно было протарифицировано коммутатором), кроме случая, когда в маске пакета услуг SMSC выставил бит безусловной тарификации через IN платформу, а для SMS Extra услуг инициируется диалог с IN-платформой, в котором короткий номер используется в качестве dstSubscriberNumber. NOTE: начиная с версии 0.6.10 появилась возможность задать в [файле конфигурации SMS Extra услуг](#) признак тарификации и для базового SMS.

2.2.1 Генерация CDR записей.

Так как формат CDR(charging data record) записей не стандартизован и различается для различных биллинговых систем, INMan не создает их непосредственно. Вместо этого он создает так называемые CSV файлы, содержащие транзакции INMan-а с необходимыми данными, которые конвертируются в конечный CDR сторонними утилитами. Формат CSV файла описан в главе '[Структура CSV файла для генерации CDR записей](#)'.

3. Протоколы взаимодействия клиентов с INMan-ом

3.1 Общий формат клиентских протоколов INMan-а.

Все клиентские протоколы, поддерживаемые INMan-ом, основаны на обмене протокольными единицами (PDU) с использованием транспортных возможностей TCP/IP и имеют единый формат пакетов. Все пакеты состоят из двух частей: служебного заголовка и содержимого PDU сериализованных в следующем порядке:

UInt32	UInt16	UInt16		
Packet length	PDU tag	Header format	Header content	PDU content

где:

1. Packet length – длина всего пакета в октетах
2. PDU tag – идентификатор PDU, определяющий как выбранный протокол общения(тарификация к примеру), так и дальнейшее содержимое пакета
3. Header format – формат служебного заголовка
4. Header content – содержимое служебного заголовка, сериализованное в соответствии с форматом
5. PDU content - сериализованное содержимое PDU

3.1.1 Используемые типы данных и их формат передачи.

Следующие типы данных используются в TCP/IP протоколах, поддерживаемых INMan-ом

- Boolean - передается как один октет, ноль означает false, любое ненулевое значение – true; далее упоминается как: Bool
- Unsigned Integers (8, 16, 32, 64 bits) – беззнаковые целые с различными максимальными значениями, передаются как определенное количество октетов (1, 2, 4, 8) в сетевом порядке байт; далее упоминается как: UInt8, UInt16, UInt32, UInt64
- Signed Integers (16, 32 bits) – знаковые целые с различными максимальными/минимальными значениями, передаются как определенное количество октетов (2, 4) в сетевом порядке байт; далее упоминается как: Int16, Int32
- TimeT - знаковое целое, хранящее время в секундах, передается как signed integer (32bit)
- Variable length strings – последовательность октетов произвольной длины, передаются с префиксом длины, кодируемым следующим образом: длина, как беззнаковое целое представляется минимальным числом бит, кратным 7-ми, разбивается на группы по 7бит, которые передаются от младших к старшим в октетах, где взведенный старший бит является признаком, того что это не последний октет префикса длины:
таким образом длины до 127 кодируются как один октет с нулевым старшим битом, длины от 128 до 16383, как два октета, со взведенным 8-м битом в первом октете и нулевым во втором, и т.д; соответственно пустая строка передается как один нулевой октет; далее упоминается как: VarString

3.1 Обзор протокола тарификации SMS/USSD сообщений (USSDC-INMAN).

Данный протокол предназначен для взаимодействия между INMan и его клиентами, которым необходимо осуществлять тарификацию услуг по отсылке/доставке SMS/USSD сообщений. Протокол использует единый формат TCP/IP соединений INMan-a.

Клиентам для взаимодействия с INMan-ом необходимо инициировать транспортное соединение, после установления которого, можно начинать обмен PDU.

Протоколом определены следующие протокольные единицы

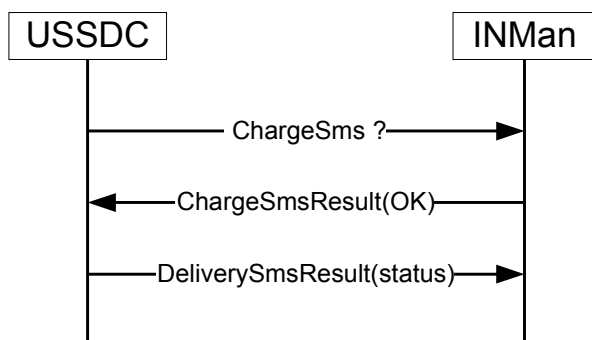
Наименование	Направлении передачи	Описание
ChargeSms	от клиента в сторону INMan	Запрашивает возможность оказания услуги
ChargeSmsResult	от INMan в сторону клиента	Разрешить/запретить оказание услуги
DeliverySmsResult	от клиента в сторону INMan	Информирует о статусе оказания услуги
DeliveredSmsData	от клиента в сторону INMan	Сообщает данные об уже оказанной услуге для проведения тарификации.

3.1.1 Описание взаимодействия

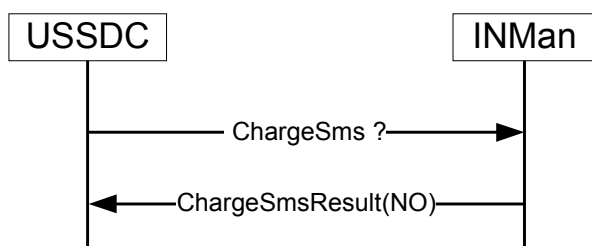
Перечисленные протокольные единицы предназначены для организации диалога в процессе оказания услуги для её тарификации. В начале диалога клиент запрашивает возможность оказания услуги, и после получения разрешения производит оказание услуги (например, доставку сообщения) с последующим уведомлением о статусе. Если в оказании услуги отказано, то уведомление производить не нужно.

В случае, если одна из сторон столкнулась с ошибкой при приеме PDU, препятствующей дальнейшему взаимодействию, то эта сторона должна проиницировать закрытие текущего соединения и открыть новое.

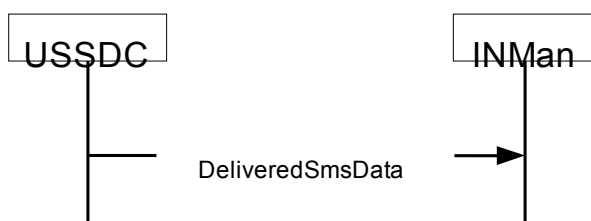
3.1.1.1 Сценарий взаимодействия (оказание услуги разрешено)



3.1.1.2 Сценарий взаимодействия (оказание услуги запрещено)



3.1.1.3 Сценарий взаимодействия (протарифицировать уже оказанную услугу)



3.1.2 Асинхронность протокола

Протокол регламентирует последовательность операций только в рамках одного диалога. Клиент может инициировать несколько параллельных диалогов. Принадлежность протокольной единицы тому или иному диалогу отслеживается с помощью специального поля в служебном заголовке пакета, значение которого задается клиентом при старте диалога с INMan-ом.

3.1.3 Описание формата PDU

Все пакеты протокола тарификации имеют единый служебный заголовок, с идентификатором формата 0x01 и содержащим [идентификатор диалога с INMan-ом](#). Таким образом, все пакеты, в соответствии с [общим форматом пакетов](#), имеют следующую структуру:

UInt32	UInt16	UInt16	UInt32	
Packet length	PDU tag	0x01	Dialog Id	PDU content

где:

1. Packet length – длина всего пакета в октетах
2. PDU tag – тэг PDU протокола тарификации:
 - 0x01 - [ChargeSms](#)
 - 0x02 - [ChargeSmsResult](#)

- 0x03 - [DeliverySmsResult](#)
- 0x04 - [DeliveredSmsData](#)

3. Dialog Id – уникальный идентификатор диалога SMSC ↔ INMan, назначаемый SMSC для обеспечения асинхронности его запросов.

Содержимое всех PDU подробно описано в главе [‘Структура PDU протокола тарификации SMS/USSD сообщений.’](#)

3.1 Структура PDU протокола тарификации SMS/USSD сообщений.

3.1.1 Структура ChargeSms PDU (тэг 0x01)

Данное PDU содержит данные для инициации диалога с IN платформой и данные для генерации CDR.

Поле	Тип	Длина	Описание
Служебные поля			
chrgFlags	UInt8	1	флаги определяющие режим и тип тарификации : bit0 — тип тарификации: 0 — MO (тарифицировать отправителя), 1 — MT (тарифицировать получателя) bit1 – выбор режима тарификации 0 – в соответствии с контрактом тарифицируемого абонента и типом сообщения 1 - принудительный режим тарификации CDR
smsXSrvsId	UInt32	4	совместная маска использованных SMS Extra услуг и флагов спец. тарификации: 0x40000000 – безусловная тарификация через IN-платформу, независимо от параметра файла конфигурации Billing.billMode
Данные для диалога с IN платформой			
dstSubscriberNumber	VarString	1 – 21	адрес получателя
callingPartyNumber	VarString	1 – 21	адрес отправителя
callingImsi	VarString	1 – 21	IMSI отправителя
smscAddress	VarString	1 – 21	SMSC адрес получателя
submitTimeTZ	TimeT	4	время посылки сообщения
tpShortMessageSpecificInfo	UInt8	1	первый октет short message TPDU (3GPP TS 23.040)
tpProtocolIdentifier	UInt8	1	идентификатор протокола из short message TPDU
tpDataCodingScheme	UInt8	1	DCS из short message TPDU
tpValidityPeriod	TimeT	4	период жизни сообщения в секундах
locationInformationMSC	VarString	1 – 21	номер коммутатора, в зоне действия которого находился отправитель
Данные для генерации CDR			
callingSMEid	VarString	1 - ...	идентификатор службы, через которую сообщение принято

routeId	VarString	1 - ...	идентификатор маршрута
serviceId	Int32	4	идентификатор услуги (из маршрута)
userMsgRef	Int32	4	пользовательский идентификатор сообщения (SMPP)
msgId	UInt64	8	системный идентификатор сообщения
ussdServiceOp	Int32	4	код USSD операции
partsNum	UInt8	1	число частей для multipart сообщения
msgLen	UInt16	2	общая длина сообщения (все части)
chargeMode	UInt8	1	тип тарификации: 0 - ON_SUBMIT – тарифицировать посылку сообщения, 1 - ON_DELIVERY (по умолчанию) – тарифицировать доставку сообщения
dsmSrvType	VarString	1 - ...	тип SMS сервиса, ассоциированного с сообщением (SMPP)

3.1.2 Структура ChargeSmsResult PDU (тэг 0x02)

Данное PDU содержит ответ INMap-а на запрос услуги.

Поле	Тип	Длина	Описание
value	UInt16	2	0x00 – услуга доступна, иначе - недоступна
errorcode	UInt32	4	Универсальный код ошибки INMap-а NOTE: ненулевой код возможен и при разрешении услуги, это означает, что несмотря на ошибку взаимодействия с IN, конфигурация INMap-а позволяет протарифицировать услугу через CDR.
errMsg	VarString	1 .. N	расшифровка кода ошибки
cntrType	UInt8	1	тип контракта: 0x00 – unknown или контракт не удалось определить из-за ошибки (см. код в errorcode) 0x01 – postpaid, 0x02 – prepaid(обслуживается IN-платформой)

3.1.3 Структура DeliverySmsResult PDU (тэг 0x03)

Данное PDU содержит данные для завершения диалога с IN платформой и данные для генерации CDR.

Поле	Тип	Длина	Описание
Данные для диалога с IN платформой			
value	UInt32	2	0x00 – доставка успешна, иначе код ошибки

Данные для генерации CDR			
final	Bool	1	true – последняя попытка доставки
destImsi	VarString	1 – 21	IMSI получателя если была доставка, иначе пустая строка
destMSC	VarString	1 – 21	номер коммутатора, в зоне действия которого находился получатель, если была доставка, иначе пустая строка
destSMEid	VarString	1 - ...	идентификатор службы, через которую сообщение доставлено получателю, если была доставка, иначе пустая строка
divertedAdr	VarString	1 - 21	Изначальный адрес получателя, если доставка была произведена на другой адрес, иначе пустая строка
finalTimeTZ	TimeT	4	Время последней доставки или попытки доставки

3.1.4 Структура DeliveredSmsData PDU (тэг 0x04)

Данное PDU содержит все данные об уже доставленном сообщении, достаточные для инициации диалога с IN платформой и генерации CDR. На данный момент это PDU используется только для проведения тарификации типа ON_DATA_COLLECTED (см. [Логика и режимы тарификации](#).)

Поле	Тип	Длина	Описание
Служебные поля			
chrgFlags	UInt8	1	флаги определяющие режим и тип тарификации : bit0 — тип тарификации: 0 — МО (тарифицировать отправителя), 1 — МТ (тарифицировать получателя) bit1 – выбор режима тарификации 0 – в соответствии с контрактом тарифицируемого абонента и типом сообщения 1 - принудительный режим тарификации CDR
smsXSrvsId	UInt32	4	совместная маска использованных SMS Extra услуг и флагов спец. тарификации: 0x40000000 – безусловная тарификация через IN-платформу, независимо от параметра файла конфигурации Billing.billMode
Данные для диалога с IN платформой			
dstSubscriberNumber	VarString	1 – 21	адрес получателя
callingPartyNumber	VarString	1 – 21	адрес отправителя
callingImsi	VarString	1 – 21	IMSI отправителя
smscAddress	VarString	1 – 21	SMSC адрес получателя
submitTimeTZ	TimeT	4	время посылки сообщения
tpShortMessageSpecificInfo	UInt8	1	первый октет short message TPDU (3GPP TS 23.040)
tpProtocolIdentifier	UInt8	1	идентификатор протокола из short message TPDU

tpDataCodingScheme	UInt8	1	DCS из short message TPDU
tpValidityPeriod	TimeT	4	период жизни сообщения в секундах
locationInformationMSC	VarString	1 – 21	номер коммутатора, в зоне действия которого находился отправитель
Данные для генерации CDR			
callingSMEid	VarString	1 - ...	идентификатор службы, через которую сообщение принято
routeId	VarString	1 - ...	идентификатор маршрута
serviceId	Int32	4	идентификатор услуги (из маршрута)
userMsgRef	Int32	4	пользовательский идентификатор сообщения (SMPP)
msgId	UInt64	8	системный идентификатор сообщения
ussdServiceOp	Int32	4	код USSD операции
partsNum	UInt8	1	число частей для multipart сообщения
msgLen	UInt16	2	общая длина сообщения (все части)
dsmSrvType	VarString	1 - ...	тип SMS сервиса, ассоциированного с сообщением (SMPP)
dlvrRes	UInt32	2	0x00 – доставка успешна, иначе код ошибки
destImsi	VarString	1 – 21	IMSI получателя если была доставка, иначе пустая строка
destMSC	VarString	1 – 21	номер коммутатора, в зоне действия которого находился получатель, если была доставка, иначе пустая строка
destSMEid	VarString	1 - ...	идентификатор службы, через которую сообщение доставлено получателю, если была доставка, иначе пустая строка
divertedAdr	VarString	1 - 21	Изначальный адрес получателя, если доставка была произведена на другой адрес, иначе пустая строка
finalTimeTZ	TimeT	4	Время последней доставки или попытки доставки

3.2 Обзор протокола определения контракта абонента (USSDC-INMAN).

Данный протокол предназначен для взаимодействия между INMan и его клиентами, которым необходимо определить тип контракта абонента - postpaid или prepaid, а для последнего типа также и адрес обслуживающей IN-платформы.

Протокол использует единый формат TCP/IP соединений INMan-а. Клиентам для взаимодействия с INMan-ом необходимо инициировать транспортное соединение, после установления которого, можно начинать обмен PDU.

Протоколом определены следующие протокольные единицы

Наименование	Направлении передачи	Описание
AbntContractRequest	от клиента в сторону INMan-а	Запрос типа контракта и параметров обслуживающей IN-платформы (если есть)
AbntContractResult	от INMan-а в сторону клиента	Ответ с типом контракта и параметрами обслуживающей IN-платформы (если есть)

3.2.1 Асинхронность протокола

Протокол регламентирует последовательность операций только в рамках одного диалога. Клиент может инициировать несколько параллельных диалогов. Принадлежность протокольной единицы тому или иному диалогу отслеживается с помощью специального поля в служебном заголовке пакета, значение которого задается клиентом при старте диалога с INMan-ом.

3.2.2 Описание формата PDU

Все пакеты протокола определения контракта абонента имеют единый служебный заголовок, с идентификатором формата 0x01 и содержащим [идентификатор диалога с INMan-ом](#). Таким образом, все пакеты, в соответствии с [общим форматом пакетов](#), имеют следующую структуру:

UInt32	UInt16	UInt16	UInt32	
Packet length	PDU tag	0x01	Dialog Id	PDU content

где:

1. Packet length – длина всего пакета в октетах
2. PDU tag – тэг PDU протокола тарификации:
 - 0x06 - AbntContractRequest
 - 0x07 - AbntContractResult
3. Dialog Id – уникальный идентификатор диалога Клиент \leftrightarrow INMan, назначаемый клиентом для обеспечения асинхронности его запросов

Содержимое всех PDU подробно описано в главе '[Структура PDU протокола определения контракта абонента](#)'.

3.1 Структура PDU протокола определения контракта абонента.

3.1.1 Структура AbntContractRequest PDU (тэг 0x06)

Данное PDU содержит идентификатор абонента и параметры, влияющие на определения его контракта.

Поле	Тип	Длина	Описание
useCache	Bool	1	флаг, разрешающий INMan-у взять информацию о контракте из кэша. При false значении, INMan определит контракт в соответствии с подходящей ' политикой абонента ' (см. AbonentPolicies)
subscriberNumber	VarString	1 – 21	номер абонента

3.1.2 Структура AbntContractResult PDU (тэг 0x07)

Данное PDU содержит ответ INMan-а с контрактом абонента, и параметрами обслуживающей IN-платформы, если таковая есть.

Поле	Тип	Длина	Описание
nmPolicy	VarString	1 .. N	имя 'политики абонента', которое было использовано INMan-ом для определения контракта; нулевая длина означает, что был использован кэш.
cntrType	UInt8	1	тип контракта: 0x00 – unknown или контракт не удалось определить из-за ошибки (см. код в serviceKeyOrError) 0x01 – postpaid, 0x02 – prepaid(обслуживается IN-платформой)
gsmSCFAddress	VarString	1 – 21	ISDN адрес IN-платформы в виде "Ton . Npi. Address" для postpaid абонента – нулевое значение(один нулевой байт).
serviceKeyOrError	UInt32	4	если gsmSCFAddress задан, то идентификатор сервиса IN-платформы для отправки SMS, иначе код ошибки INMan-a
abImsi	VarString	1 -- 17	IMSI абонента, если определен, иначе один нулевой байт
errMsg	VarString	1 .. N	описание ошибки или один нулевой байт

4. Структура CSV файла для генерации CDR записей.

CSV (от англ. comma separated values) представляет из себя обыкновенный текстовый файл, в котором существует два типа строк:

- строка заголовка – это первая строка файла, содержит перечисление через запятую имен полей данных, такая строка в файле может быть только одна
- строка данных – содержит перечисление через запятую значений в текстовом виде полей данных, таких строк может быть произвольное количество (даже 0)

4.1 Типы и формат данных полей CSV файла.

Для полей CSV файла используются следующие типы данных

- UInt[8|32|64] – беззнаковые целые с различным максимальными значениями (8, 32, 64 bits), записываются как строка содержащая только десятичные символы ('0' – '9')
- DateTimeString – дата и время, записывается как строка в виде DD.MM.YYYY HH:MM:SS например 17.05.2007 09:53:28
- NumString – просто строка десятичных символов('0' – '9') в кавычках, например "250013901464251"
- AddrString – ISDN адрес, записывается как строка в кавычках в виде: ".Ton.Npi.Signals", например: ".1.1.79139343290"
- CString – просто строка в кавычках, содержащая ASCII символы, например "sibinco.sms > plmn.kem"

4.1 Поля данных CSV файла.

В CSV файле определены следующие поля данных:

Поле	Соотв.Тип	Описание
MSG_ID	UInt64	системный идентификатор сообщения
RECORD_TYPE	UInt8	тип CDR записи: 0 (ordinary)– обыкновенная доставка, 1 (diverted)-сообщение доставлено с переадресацией (см DIVERTED_FOR)
MEDIA_TYPE	UInt8	тип содержимого сообщения: 0 – текст, 1 – бинарные данные
BEARER_TYPE	UInt8	тип носителя сообщения: 0 – SMS, 1 - USSD
SUBMIT	DateString	время отправки сообщения
FINALIZED	DateString	время доставки сообщения
STATUS	UInt32	статус доставки: 0 – успешно, иначе - код ошибки
SRC_ADDR	AddrString	адрес отправителя
SRC_IMSI	NumString	IMSI отправителя
SRC_MSC	AddrString	номер коммутатора, в зоне действия которого находился отправитель
SRC_SME_ID	CString	идентификатор службы, через которую сообщение принято
DST_ADDR	AddrString	адрес получателя
DST_IMSI	NumString	IMSI получателя
DST_MSC	AddrString	номер коммутатора, в зоне действия которого находился получатель, если была доставка, иначе пустая строка
DST_SME_ID	CString	идентификатор службы, через которую сообщение доставлено получателю, если была доставка, иначе пустая строка
DIVERTED_FOR	AddrString	реальный адрес получателя если сообщение доставлено с переадресацией, иначе пустая строка
ROUTE_ID	CString	идентификатор маршрута
SERVICE_ID	UInt32	идентификатор услуги (из маршрута), в случае SMS Extra - сервис код пакета услуг SMS Extra
SERVICE_TYPE	CString	тип SMS сервиса, ассоциированного с сообщением (SMPP)
USER_MSG_REF	UInt32	пользовательский идентификатор сообщения (SMPP)
DATA_LENGTH	UInt32	общая длина сообщения в байтах для бинарного, в символах для текстового
PARTS_NUM	UInt8	число частей сообщения
SMSX_SRV	UInt32	маска пакета услуг SMS Extra
MT	UInt8	1 – тарификация получателя, 0 – тарификация отправителя
CONTRACT	UInt8	тип контракта абонента: 0x00 – unknown или контракт не удалось определить из-за ошибки 0x01 – postpaid, 0x02 – prepaid(обслуживается IN-платформой)
CHARGE	UInt8	тип тарификации (см. Логика и режимы тарификации): 0 - ON_SUBMIT – тарификация отправки сообщения, 1 - ON_DELIVERY (по умолчанию) – тарификация доставки сообщения, 2 - ON_DATA_COLLECTED – тарификация доставки сообщения, у которого предварительно была протарифицирована отправка.
IN_BILL	UInt8	флаг проведения тарификации через IN платформу:

		<p>0 – нет,</p> <p>1 – да</p>
--	--	-------------------------------

5. Описание параметров файла конфигурации INMan-a.

При старте INMan считывает параметры инициализации из файла конфигурации. Этот файл является xml документом, структура которого позволяет создавать иерархическое пространство имен для параметров и указывать тип параметров. Иерархия задается тэгами *section* внутри документа. Ниже приведен пример конфигурационного файла с одним корневым параметром *version* и параметром *maxBillings* внутри секции *Billing*.

Начало файла

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE config SYSTEM "file://configuration.dtd">
<config>
  <param name="version" type="string">0.7.0</param>
  <section name="Billing">
    <param name="maxBillings" type="int">500</param>
  </section>
</config>
```

Конец файла

Логически INMan состоит из нескольких составляющих (конфигурируемых сервисов), каждая из которых, как правило, конфигурируется одной или несколькими секциями файла конфигурации.

5.1 Конфигурируемые сервисы.

Эксплуатация модуля тарификации выявила, что его функциональность востребована в различных вариантах на инсталляциях с различными как ПО так и оборудованием. Основные варианты использования INMan-a следующие:

- сервис тарификации (только тарификация SMS/USSD сообщений, работа с сигнальной сетью)
- сервис определителя абонентов (только определение данных контракта абонента, работа с сигнальной сетью)
- backup архиватор CDR-ов (работа без сигнальной сети, создание CDR записей по оказанным услугам в случае сбоев сигнальной сети или для последующего использования в системах статистики)

Для обеспечения возможности гибкой настройки INMan-a под конкретные нужды, начиная с версии 0.7.0 INMan был перестроен по технологии ‘конфигурируемых динамически подгружаемых сервисов’, далее упоминающиеся как просто ‘конфигурируемые сервисы’. Эта технология позволяет при старте INMan-a, в соответствии с настройками файла конфигурации, определить востребованные сервисы и загрузить и проинициализировать для дальнейшей работы только их.

Каждый сервис идентифицируется уникальным идентификатором (UId) и может быть либо встроен в INMan, либо динамически загружен при его старте.

Функционально все сервисы подразделяются на две группы :

- сервисы верхнего уровня (topLevel) – непосредственно обслуживают клиентов INMan-a
- разделяемые сервисы (shared)– обслуживают другие сервисы.

Сервисы обеих групп могут быть конфигурируемыми. Таким сервисам назначена по умолчанию секция в файле конфигурации, которая может быть переназначена в специальной секции этого файла.

Для выбора нужных сервисов верхнего уровня а также для переназначения секций конфигурации для конфигурируемых сервисов служит головная секция файла конфигурации с зарезервированным именем [‘Services’](#).

В данной версии, для настройки посредством файла конфигурации доступны следующие сервисы:

Таблица 1: Конфигурируемые сервисы

UID (тип сервиса)	Имя секции по умолчанию (+ допол.-е)	Тип загрузки (имя динамической библиотеки)	Описание
icsTCPServer (shared)	Host	Prelinked	Обслуживает TCP соединения клиентов INMan-a, в соответствии с правилами, описанными в главе “Протоколы взаимодействия клиентов с INMan-ом”
icsAbntCache (shared)	AbonentsCache	dynamic (libinman_ics_abcach.so)	Кэш данных об абонентах (контракт, IN-платформы, etc)
icsTCAPDisp (shared)	SS7 либо SS7_HD	dynamic (libinman_ics_tcapdsp.so)	Предоставляет TCAP API для взаимодействия с элементами сигнальной сети, интегрируется с SS7 стэкком TietoEnator, обычным или HD в зависимости от варианта сборки INMan-a
icsIAPManager (shared)	AbonentPolicies (+ IN-platforms , AbonentProviders)	dynamic (libinman_ics_iapmgr.so)	Реестр ‘политик’ и ‘провайдеров’ абонентов.
icsSmBilling (topLevel)	Billing	dynamic (libinman_ics_smbill.so)	Сервис тарификации SMS/USSD сообщений, в соответствии с протоколом, описанным в главе Обзор протокола тарификации (USSDC-INMAN) .
icsAbntDetector (topLevel)	AbonentDetector	dynamic (libinman_ics_abdtcr.so)	Сервис определения контракта абонентов, в соответствии с протоколом, описанным в главе Обзор протокола определения контракта абонента (USSDC-INMAN) .
icsIAPrvdSRI (shared)	Подсекция в AbonentProviders	dynamic (libinman_iap_sri.so)	Сервис ‘провайдера абонентов’ использующий HLR сервис SEND_ROUTING_INFO

5.1 Сводный список секций файла конфигурации.

В нижеследующей таблице перечислены все секции верхнего уровня, определенные в файле конфигурация INMan-a текущей версии.

Имя секции (по умолчанию)	Наличие	Краткое описание
Services	опциональная	задаёт требуемые topLevel сервисы
Host	обязательная	
Billing	опциональная	тарификация SMS/USSD сообщений
AbonentDetector	опциональная	определение контракта абонентов
SS7	опциональная	интеграция с SS7 стэкком TietoEnator
SS7_HD	опциональная	интеграция с HD версией SS7 стэка TietoEnator

TCAPUser	опциональная	параметры работы с TCAP API
AbonentsCache	опциональная	кэш данных об абонентах
AbonentPolicies	опциональная	определения 'политик абонентов'
IN-platforms	опциональная	конфигурации известных IN-платформ
AbonentProviders	опциональная	конфигурации 'провайдеров абонентов'

5.2 Структура секции Services.

Данная секция является главной и служит для указания требуемых сервисов верхнего уровня, а также для переназначения секций конфигурации для конфигурируемых сервисов, перечисленных в разделе [Конфигурируемые сервисы](#).

Секция может содержать только строковые параметры вида:

```
<param name="SectionName" type="string">ICSIdent</param>
```

где

- SectionName – имя секции файла конфигурации (не обязательно верхнего уровня), которая содержит настройки для сервиса, идентифицируемого значением данного строкового параметра ICSIdent
- ICSIdent – строка, идентифицирующая сервис в одном из следующих форматов:
 1. [uid:] icsUId – где icsUId уникальный идентификатор сервиса указанный в таблице из раздела [Конфигурируемые сервисы](#), если сервис является динамически подгружаемым, то INMan загрузит соответствующую библиотеку.
 2. dll: libname.so – где libname.so имя динамической библиотеки желаемого сервиса.

NOTE: Для единообразия файлов конфигурации на введенных в эксплуатацию инсталляциях рекомендуется использовать ICSIdent вида 1, так как режим загрузки сервисов (prelinked/dynaptic) может различаться для разных инсталляций.

Пример:

```
<section name="Services">
  <!-- Abonent contract type and gsmSCF params determination -->
  <param name="AbonentDetector" type="string">uid: icsAbntDetector</param>
  <param name="HostWAN" type="string">uid: icsTCPSTServer</param>
</section>
```

В данном варианте INMan запустит сервис определения контракта абонентов, а настройки для сервиса icsTCPSTServer возьмет из секции 'HostWAN', вместо секции по умолчанию 'Host'.

Несмотря на то, что эта секция главная, она опциональна. Если она отсутствует или не имеет параметров, то считается, что секция имеет вид:

```
<section name="Services">
  <param name="Billing" type="string">uid: icsSmBilling</param>
</section>
```

Это сделано для совместимости с файлами конфигурации предыдущих версий INMan-a.

5.1 Параметры секции Host (сервис icsTCPSTServer).

Данная секция является обязательной. В ней определены следующие параметры

Имя параметра	Тип	Наличие	Описание
Host	строка	обязательный	DNS имя или адрес хоста INMan-a
Port	целое	обязательный	Служебный порт INMan-a
maxClients	целое	опциональный	Максимальное количество клиентских соединений, по умолчанию: 3

5.2 Параметры секции *AbonentsCache* (сервис *icsAbntCache*).

Данная секция опциональна. Она используется только если один из сервисов востребовал функциональность кэша абонентов. В секции определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
cacheRAM	целое (Mb)	обязательный	максимальный размер ОЗУ под кэш абонентов
cacheDir	строка	обязательный	директория для хранения файлов кэша абонентов, если значение пустое, то кэшэм будет использоваться только ОЗУ
cacheExpiration	целое (минуты) [1 .. 65535]	опциональный (default)	время жизни записи кэша абонентов в минутах, по умолчанию: 20
cacheRecords	целое	опциональный (default)	начальное число записей файла кэша абонентов, при его создании; по умолчанию: 10000

5.3 Параметры секции *SS7* (сервис *icsTCAPDisp*).

Данная секция опциональна. Она используется только если один из сервисов нуждается во взаимодействии с элементами сигнальной сети.

Секция предназначена для настройки подключения к SS7 стэку фирмы TietoEnator. В секции определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
ss7UserId	целое ([1..20])	обязательный	идентификатор пользователя для использования TietoEnator TCAP API, рекомендуется: 3 (USER03_ID)
maxMsgNum	целое ([1..65535])	опциональный (default)	максимальное число сообщений в буфере входящих SS7 пакетов, по умолчанию: 512

5.4 Параметры секции *SS7_HD* (сервис *icsTCAPDisp*).

Данная секция опциональна. Она используется только если один из сервисов нуждается во взаимодействии с элементами сигнальной сети.

Секция предназначена для настройки подключения к HD (horizontal distribution) версии SS7 стэка фирмы TietoEnator. В секции определены параметры, необходимые для соединения с удаленным SS7 HD CommonParts manager-ом (см. сопроводительную документацию TietoInator), а также две подсекции:

- **SS7Units** - описывает модули HD SS7 стэка, доступные для взаимодействия
- **Layouts** - описывает варианты взаимодействия с модулями стэка.

Параметры секции:

Имя параметра	Тип	Наличие	Описание
cpMgrHosts	строка (CSVList)	обязательный	CSV список адресов хостов, на которых работают требуемые SS7 HD CommonParts manager-ы. Каждый адрес представляется в формате: "DNS_имя : порт "

			Пример: 'localhost:6666 , localhost: 6667'
cpMgrInstId	целое ([0..255])	опциональн й (default)	Номер экземпляра SS7 HD CommonParts manager-a, по умолчанию: 0
mp7UserId	целое ([1..20])	обязательный	идентификатор пользователя для использования TietoEnator TCAP API, рекомендуется: 3 (USER03_ID)
maxMsgNum	целое ([1..65535])	опциональн й (default)	максимальное число сообщений в буфере входящих SS7 пакетов, по умолчанию: 512
dfltLayout	строка	обязательный	Имя подсекции из секции Layouts , которая описывает вариант интеграции с модулями HD SS7 стэка.

5.4.1 Параметры подсекции SS7Units.

Данная подсекция обязательная. Она содержит только подсекции, описывающие модули HD SS7 стэка , доступные для взаимодействия. Каждая подсекция может иметь произвольное имя, но должна обязательно иметь следующие параметры:

Имя параметра	Тип	Наличие	Описание
unitId	строка	обязательный	идентификатор модуля HD SS7 стэка, как указано в файле portss7.h
instancesList	строка (CSV list)	опциональн й (default)	список номеров ([1..255]) экземпляров модуля, разделенных запятой, по умолчанию: 1

NOTE: на данный момент распознаются только следующие идентификаторы модулей HD SS7 стэка:

- **TCAP_ID**
- **SCCP_ID**

Например:

```
<section name="SS7Units"> <!-- configuration of known SS7 Units →
  <section name="TCAP_BE_HD2"> <!-- TCAP unit distributed by two instances -->
    <param name="unitId" type="string">TCAP_ID</param>
    <param name="instancesList" type="string">1, 2</param>
  </section>
  <section name="SCCP_BE_1"> <!-- single instance SCCP unit -->
    <param name="unitId" type="string">SCCP_ID</param>
    <param name="instancesList" type="string">1</param>
  </section>
</section> <!-- SS7_HD.SS7Units -->
```

5.4.2 Параметры подсекции Layouts.

Данная подсекция обязательная. Она содержит только подсекции, описывающие варианты интеграции с модулями HD SS7 стэка, описанными в секции **SS7Units**.

Каждая подсекция может иметь произвольное имя, но, как минимум, одна подсекция должна быть определена и иметь следующие параметры:

Имя параметра	Тип	Наличие	Описание
instanceId	целое ([1..255])	опциональный (default)	локальный номер экземпляра приложения, взаимодействующего с модулями HD SS7 стэка, по умолчанию: 1
ss7units	строка (CSV list)	обязательный	описывает используемые модули HD SS7 стэка. содержит список имен подсекций из секции SS7Units , разделенных запятой

Например:

```
<section name="Layouts">
  <section name="ComplexApplication"> <!-- complex application that uses HD TCAP BE
                                     (distributed by two instances) and single SCCP BE -->
    <param name="instanceId" type="int">4</param>
    <param name="ss7Units" type="string">TCAP_BE_HD2, SCCP_BE_1</param>
  </section>
</section> <!-- SS7_HD.Layouts ->
```

5.5 Шаблон секции TCAPUser

Для инициализации работы с TCAP API существует определенный набор параметров. Каждый конфигурируемый компонент INMan-а, использующий TCAP API, должен в своей секции конфигурации либо иметь подсекцию либо ссылаться на внешнюю секцию с этими параметрами. Имя секции может быть произвольно, а параметры должны быть следующие:

Имя параметра	Тип	Наличие	Описание
ownAddress	строка	обязательный	ISDN адрес используемый данным компонентом
ownSsn	целое ([1..255])	обязательный	номер сабсистемы TCAP используемой данным компонентом
fakeSsn	целое ([1..255])	опциональный (default)	номер сабсистемы TCAP, указываемой в originating address, значение по умолчанию: 0 – то есть указывать ownSsn
rosTimeout	целое (секунды: [1..65535])	опциональный (default)	таймаут на выполнение MAP/CAP операций, по умолчанию: 20
maxDialogs	целое ([2..65530])	опциональный (default)	максимальное количество TCAP диалогов на подсистему, по умолчанию 2000

5.6 Параметры секции AbonentDetector (сервис *icsAbntDetector*).

Данная секция является опциональной. Она предназначена для настройки сервиса определения контракта абонентов. В секции определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
abonentPolicy	строка	обязательный	задает имя 'политики абонента' по умолчанию (название подсекции в секции AbonentPolicies), она применяется если MSISDN адрес абонента не соответствует ни одному из пулов адресов заданных

			политик
abonentTypeTimeout	целое (секунды: [1 .. 65535])	опциональн й (default)	Таймаут, ограничивающий время на определение контракта абонента, по умолчанию: 8 сек.
maxRequests	целое [1 .. 100000]	опциональн й (default)	Максимальное количество одновременно обрабатываемых запросов на одно клиентское соединение, по умолчанию: 1000
useCache	bool	опциональн й (default)	Включает использование кэша абонентов, по умолчанию: true
cacheExpiration	целое (минуты) [1 .. 65535]	опциональн й (default)	Время легитимности данных из кэша абонентов, по умолчанию: 20, значение 0 означает использовать cacheExpiration из настроек кэша.

5.7 Параметры секции Billing (сервис *icsSmBilling*).

Данная секция является опциональной. Она предназначена для настройки сервиса тарификации SMS/ussd сообщений. В секции определены подсекция **BillingModes**, задающая режимы тарификации для различных типов сообщений, а также следующие параметры.

Имя параметра	Тип	Наличие	Описание
abonentTypeRequest	строка	опциональн й (default)	режим определения типа котракта абонента (запрос к провайдеру абонентов): <ul style="list-style-type: none"> • onDemand – по требованию, если необходимо для тарификации (по умолчанию), • always - всегда
abonentPolicy	строка	опциональн й	задает имя 'политики абонента' по умолчанию (название подсекции в секции AbonentPolicies), она применяется если MSISDN адрес абонента не соответствует ни одному из пулов адресов заданных политик требуется, если в подсекции BillingModes присутствует режим тарификации IN или abonentTypeRequest выставлен в always
abonentTypeTimeout	целое (секунды: [1 .. 65535])	опциональн й (default)	Таймаут, ограничивающий время на определение контракта абонента, по умолчанию: 8 сек. требуется, если в подсекции BillingModes присутствует режим тарификации IN или abonentTypeRequest выставлен в always
maxBillings	целое [1 .. 100000]	опциональн й (default)	Максимальное количество одновременно обрабатываемых запросов на одно клиентское соединение, по умолчанию: 1000
maxTimeout	целое (секунды: [5 .. 65535])	опциональн й	Максимальный таймаут на промежуточные операции (взаимодействие с SMSC, etc); по его истечению INMan прерывает биллинг; по умолчанию: 180 сек
cdrMode	строка	опциональн й (default)	Определяет режим записи CSV файлов для генерации CDR записей. Варианты:

			<ul style="list-style-type: none"> none – отключить генерации CSV; all - создавать CSV на любой запрос; billMode - создавать CSV в зависимости от заданного режима тарификации (по умолчанию)
cdrDir	строка	опциональн й	директория для хранения файлов с CDR записями, требуется если cdrMode не none
cdrInterval	целое (секунды: [1 .. 65535])	опциональн й	Период накопления записей в CSV файл , по истечению создается следующий файл, требуется если cdrMode не none
useCache	bool	опциональн й (default)	Включает использование кэша абонентов, по умолчанию: true
cacheExpiration	целое (минуты) [1 .. 65535]	опциональн й (default)	Время легитимности данных из кэша абонентов, по умолчанию: 20, значение 0 означает использовать cacheExpiration из настроек кэша.
CAP3Sms	строка	опциональн й	Имя секции вида Шаблон секции TCAPUser с настройками для диалога с IN-платформами. требуется, если в подсекции BillingModes присутствует режим тарификации IN
smsExtraConfig	строка	опциональн й	имя файла конфигурации SMS Extra услуг.

5.7.1 Параметры подсекции BillingModes.

Данная подсекция задает режимы тарификации для типов сообщений описанных в [Логика и режимы тарификации](#). Секция может содержать только подсекции с именами, соответствующими типам сообщений. В каждой такой подсекции определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
MO	строка	обязательный	режим тарификации для тарификации отправителя сообщения
MT	строка	обязательный	режим тарификации для тарификации получателя сообщения

NOTE: Для режима 'IN', возможно задание через запятую вторичного режима тарификации: **CDR** или **OFF**. Вторичный режим будет использоваться в случае если по какой-либо причине (отказ сигнальной сети, недостаточно данных в файле конфигурации) невозможно инициировать диалог с IN-платформой. По умолчанию используется **OFF**.

NOTE: В случае сбоя, после того как IN-платформа уже дала разрешение на оказание услуги, ВСЕГДА формируется CDR запись.

Пример:

```
<section name=" BillingModes ">
  <section name="SMS">
    <param name="MO" type="string">OFF</param>
    <param name="MT" type="string">IN,CDR</param>
  </section>
```

```
<section name="XSMS">
  <param name="MO" type="string">IN,OFF</param>
  <param name="MT" type="string">IN,CDR</param>
</section>
</section>
```

NOTE: если хотя бы один режим тарификации выставлен в *IN*, то в файле конфигурации INMan-а должна быть заполнена секция SS7 для обеспечения взаимодействия с элементами сигнальной сети .

5.8 Параметры секции AbonentPolicies (сервис *ics/APManager*).

Данная секция является обязательной при режимах тарификации, отличных от *none*.. Она определяет ‘*политики абонентов*’ в виде подсекций фиксированного формата. К примеру:

```
<section name="AbonentPolicies">
  <section name="MTC_Nsk_Default">
    <!-- abonent policy parameters -->
  </section>
</section>
```

Здесь MTC_Nsk_Default есть имя политики.

NOTE: Как минимум одна политика должна быть определена, и ее имя указано в качестве значения параметра [Billing.abonentPolicy](#).

Параметры ‘политики абонентов’

Имя параметра	Тип	Наличие	Описание
policy	строка (CSV list)	обязательный	<p>список из двух опциональных частей:</p> <p>первая – имя ‘провайдера абонента’, (имя подсекции) из секции AbonentProviders,</p> <p>вторая – CSV list имен IN-platform (имена подсекций) из секции IN-Platforms.</p> <p>NOTE: здесь можно использовать символ ‘*’ (asterisk) для обозначения всех IN-platform из секции IN-Platforms.</p> <p>NOTE: см. примеры ниже.</p>
addrPool	строка (format)	опциональный	<p>маска пула адресов в виде формата sscanf(),</p> <p>если отсутствует, то данная политика исключается при определении политики по адресу абонента, и может быть только указана в Billing.abonentPolicy.</p> <p>пример: 7913%7[0-9]</p>
active	boolean	опциональный	<p>флаг использования политики, по умолчанию true</p> <p>(полезно для временного отключения политики)</p>

NOTE: CSV list означает строку, содержащую список слов, разделенных запятой.

Примеры значений параметра policy:

- пусть в секции [AbonentProviders](#) определена подсекция *HLR_Nsk_SRI*, задающая параметры для ‘провайдера абонентов’ определяющего контракт абонента через SEND_ROUTING_INFO сервис HLR-a, а в секции [IN-Platforms](#) определена подсекция *IN_Nsk*, задающая serviceKey для диалога с IN-платформой, тогда политика может иметь значение: “*HLR_Nsk_SRI, IN_Nsk*”
- пусть в секции [AbonentProviders](#) определена подсекция *HLR_Msk_SRI*, задающая параметры для ‘провайдера абонентов’ определяющего контракт абонента через SEND_ROUTING_INFO сервис главного HLR-a, а в секции [IN-Platforms](#) определены несколько подсекций для региональных IN-платформ, тогда политика может иметь значение: “*HLR_Msk_SRI, **”(обратите внимание на asterisk!!)
- пусть в секции [IN-Platforms](#) определена подсекция *IN_Nsk_Huawei*, задающая serviceKey для диалога с IN-платформой, тогда политика может иметь значение: “*, IN_Nsk_Huawei*” (обратите внимание на лидирующую запятую!!), в этом случае абонент будет считаться как prepaid только в случае успешного диалога с IN-платформой (rpCause == (0 || RPCList_reject)).

NOTE: если политики абонентов используют IN платформы или провайдера абонентов, взаимодействующего с элементами сигнальной сети, то будет автоматически загружен сервис [icsTCAPDisp](#), для которого в файле конфигурации INMan-а должна быть заполнена секции [SS7](#) или [SS7_HD](#), в зависимости от сборки INMan-а, для обеспечения взаимодействия с HLR сигнальной сети.

5.1 Структура и параметры секции IN-Platforms (сервис *icsIAPManager*).

Данная секция является опциональной. Она содержит конфигурации IN-платформ, содержащих параметры, необходимые для диалога с ними. Данная секция необходима, если используемые ‘политики абонентов’ основаны на ‘провайдерах абонентов’, которые не способны предоставить все параметры для диалога с IN-платформой, или если требуется проводить тарификацию не по параметрам, предоставленным провайдером.

Секция может содержать только подсекции, содержащие конфигурации отдельных IN-платформ, в виде:

```
<section name="IN-platforms"> <!-- known IN-platforms with serviceKeys specified -->
  <section name="IN_Nsk_Huawei">
    .. параметры IN-платформы ..
  </section>
</section>
```

Здесь "IN_Nsk_Huawei" есть имя IN-платформы, которое может быть использовано в значении ‘политики абонента’ а также идентифицирует данную IN-платформу в журнальных записях INMan-а.

В эксплуатации встречаются особые конфигурации сигнальной сети, в которых после определения обслуживающей абонента IN-платформы, реальный диалог надо произвести с другой IN-платформой (к примеру, один из узлов группы). Для описания таких ситуаций вводится понятие ‘*alias-платформа*’. Соответственно, секция конфигурации IN-платформы может иметь один из двух форматов:

- формат ‘*alias-платформы*’, содержащий ссылку на другую IN-платформу
- формат ‘*параметры диалога*’, содержащая данные для диалога с данной IN-платформой.

5.1.1 Параметры конфигурации в формате ‘параметры диалога’

В данном формате секция содержит обязательные и опциональные параметры для диалога с IN-платформой, а также опциональную подсекцию [ServiceKeys](#), задающую сервис-ключи.

NOTE: *параметры IN-платформ, указанные в данной секции имеют приоритет, перед параметрами, полученными из других источников.*

Определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
scfAddress	целое	обязательный	ISDN адрес IN платформы
RPCList_reject	строка (CSV list)	опциональный	список дополнительных кодов отказа в обслуживании, возвращаемых IN платформой в случае недостаточного

			баланса; на данный момент реализовано в IN платформе фирмы Alcatel: 16
RPCList_retry	строка (CSV list)	опциональный	<p>список кодов отказа в обслуживании, возвращаемых IN платформой, в случае получения которых, запрос тарификации повторится указанное число раз, <i>Рекомендуется использовать кода 41 (temporary failure) и 42(congestion) определенные в 3GPP TS 24.011 Annex E-2:</i> <i>RP-cause definition for mobile originating SM-transfer,</i></p> <p>элемент CSV списка должен иметь следующий формат: <i>rpCause[:attempNum]</i></p> <p>если <i>attempNum</i> не указан, по умолчанию используется значение 1</p> <p>пример: 41, 42:2</p> <p>один повтор, при получении кода 41, два - для 42</p>
IDPLocationInfo	строка	опциональный (default)	<p>определяет тип адреса , подставляемого в LocationInformationMSC при диалоге с данной IN-платформой, варианты:</p> <ul style="list-style-type: none"> • MSC – адрес коммутатора сети (default), • SMSC – адрес SMS центра, запросившего обслуживание, • SSF – адрес INMan-a
IDPReqMode	строка	опциональный (default)	<p>определяет режим отправки нескольких запросов на тарификацию одного абонента в IN-платформу, варианты:</p> <ul style="list-style-type: none"> • MT - многопоточный режим (default) • SEQ – последовательный <p>NOTE: последовательный режим значительно снижает пропускную способность модуля тарификации, поэтому рекомендуется его выставлять только для IN-платформ , заведомо не имеющих способности к многопоточной обработке запросов. На данный момент этот недостаток присущ IN-платформам фирмы Commverse.</p>

NOTE: CSV list означает строку, содержащую список целых значений, разделенных запятой.

5.1.1.1 Параметры подсекции **ServiceKeys**.

Одним из обязательных параметров диалога с IN-платформой является serviceKey. Вследствие особенностей реализации INMan подразделяет сервис-ключи на категории, в зависимости от типа предоставляемого сервиса. На данный момент для задания в конфигурации IN-платформ доступны следующие категории:

- **MO-BC** – Basic Call handling , mobile originating
- **MO-SM** – Short Message transfer, mobile originating

Данная подсекция опциональна и может содержать только строковые параметры с именами, соответствующими указанным выше категориям сервис-ключей.

Значением параметров является строка, описывающая алгоритм вычисления сервис ключа в следующем формате:

[algorithm_type :] algorithm_argument [:algorithm_data_section]

Поддерживаются следующие алгоритмы:

- **val** – аргументом является строка, представляющая целое беззнаковое число, algorithm_data_section не определен, алгоритм просто возвращает свой аргумент как UInt32
- **map** - аргументом является строка, содержащая имя категории сервис-ключей, в случае этого алгоритма INMan пытается определить сервис-ключ данной категории и далее произвести маппинг по таблице соответствия, заданной в секции, на которую указывает algorithm_data_section; algorithm_data_section имеет вид секции с целыми параметрами, где имена параметров соответствуют значениям algorithm_argument, а значения параметров – результату.

NOTE: если algorithm_type не указан, то подразумевается алгоритм **val**

Пример 1: Использование алгоритма **val**

```
<section name="ServiceKeys">
  <param name="MO-SM" type="string">2 </param>
</section>
```

Пример 2: Использование алгоритма **map**

```
<section name="ServiceKeys">
  <param name="MO-SM" type="string">map: MO-BC: mapMOBC-MOSM</param>
</section>

<section name="mapMOBC-MOSM">
  <param name="80" type="int">1</param>
  <param name="82" type="int">2</param>
  <param name="84" type="int">3</param>
  <param name="86" type="int">4</param>
  <param name="88" type="int">5</param>
</section>
```

5.1.1 Параметры конфигурации в формате '*alias-платформы*'

Данная секция содержит следующие параметры:

Имя параметра	Тип	Наличие	Описание
scfAddress	целое	обязательный	ISDN адрес IN платформы
aliasFor	строка	обязательный	имя секции описывающей alias-платформу

Пример: пусть определена секция конфигурации IN-платформы IN_Ksd_Ericsson

```
<section name="IN_Ksd_Ericsson_1">
  <param name="scfAddress" type="string">79184790291</param>
  <param name="aliasFor" type="string">IN_Krd_Ericsson</param>
</section>
```


5.2 Параметры секции AbonentProviders (сервис *icsIAPManager*).

Данная секция содержит конфигурации для подгружаемых модулей ‘провайдеров абонентов’. Она является опциональной, так как возможны варианты установки INMan-а, при которых ‘провайдеры абонентов’ не используются.

Секция может содержать только подсекции в виде:

```
<section name="AbonentProviders"> <!-- abonent contract/gsmSCF params providers -->
  <section name="HLR_Nsk_SRI">
    <!-- конфигурация 'провайдера абонентов' -->
  </section>
</section>
```

Здесь "HLR_Nsk_SRI" есть имя провайдера, которое может быть использовано в значении ‘политики абонета’.

Каждая подсекция имеет полуфиксированный формат. Это значит, что она должна иметь подсекцию “Config”, содержимое которой различно для разных типов ‘провайдеров абонентов’, и параметры, не зависящие от типа провайдера. Определены следующие параметры:

Имя параметра	Тип	Наличие	Описание
loadup	строка	обязательный	Идентификатор требуемого 'конфигурируемого сервиса' (ICSIdent), реализующего данный вид провайдера.

Пример:

```
<section name="AbonentProviders"> <!-- abonent contract/gsmSCF params providers -->
  <section name="HLR_Nsk_SRI">
    <param name="loadup" type="string">icsIAPrvdSRI</param>
    <section name="Config">
      ... параметры
    </section>
  </section>
</section>
```

Сборка INMan-а версии 0.7.11 распознаёт следующие виды ‘провайдеров абонентов’:

- uid: icsIAPrvdSRI (dll: libinman_iap_sri.so) - использует SEND_ROUTING_INFO сервис HLR-а, определяет контракт абонента и ISDN адрес IN-платформы; (для диалога с IN дополнительно нужен [serviceKey](#))

5.2.1 Параметры подсекции Config для провайдера icsIAPrvdSRI.

Провайдер icsIAPrvdSRI реализует функциональность ‘провайдера абонентов’ посредством использования сервиса SEND_ROUTING_INFO, предоставляемого HLR-ом сигнальной сети. Библиотека позволяет определить тип контракта абонента, а для prepaid абонента также и ISDN адрес IN-платформы (без параметров для диалога)

Следующие параметры определены:

Имя параметра	Тип	Наличие	Описание
tcapUser	строка	обязательный	имя секции вида TCAPUser с параметрами для данного провайдера

NOTE: при использовании данной библиотеки будет автоматически загружен сервис [icsTCAPDisp](#), для которого в файле конфигурации INMan-а должна быть заполнена секция [SS7](#) для обеспечения взаимодействия с HLR сигнальной сети.

1. Описание файла конфигурации SMS Extra услуг.

Файл конфигурации SMS Extra услуг устроен по тому же принципу что и файл конфигурации INMan-а, и содержит только одну секцию "SMSEExtra" :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE config SYSTEM "file://configuration.dtd">
<config>
  <section name="SMSEExtra">
    ....
  </section>
</config>
```

Каждая услуга, а вернее пакет услуг, описывается подсекцией со следующими параметрами:

Имя параметра	Тип	Наличие	Описание
serviceCode	целое	обязательный	сервис-код пакета услуг
serviceMask	целое	обязательный	маска пакета услуг
serviceAdr	строка	опциональный	короткий номер (dstSubscriberNumber) для обращения к IN-платформе, его отсутствие означает тарификацию услуг через создание CDR
chargeBearer	bool	опциональный (default)	признак тарификации сообщения-носителя (базового SMS), по умолчанию: false . При значении true INMan, будет инициировать для prepaid абонентов два диалога с IN-платформой: один для тарификации базового сообщения до получателя, второй - для тарификации SMS Extra услуги. Для postpaid абонентов по прежнему генерится одна CSV запись со всей информацией.

Пример:

```
<section name="SMS-Flash">
  <param name="serviceCode" type="int">1</param>
  <param name="serviceMask" type="int">2</param>
  <param name="serviceAdr" type="string">smsx:7961</param>
</section>
<section name="SMS-Calendar">
```

```
<param name="serviceCode" type="int">2</param>
<param name="serviceMask" type="int">4</param>
<param name="serviceAdr" type="string">smsx:7962</param>
</section>
<section name="SMS-Flash-Calendar">
  <param name="serviceCode" type="int">5</param>
  <param name="serviceMask" type="int">6</param> <!-- 4+2 -->
  <param name="serviceAdr" type="string">smsx:7965</param>
</section>
<section name="SMS-Flash-Group">
  <param name="serviceCode" type="int">7</param>
  <param name="serviceMask" type="int">18</param> <!-- 16+2 -->
  <param name="serviceAdr" type="string">smsx:7965</param>
  <param name="chargeBearer" type="bool">true</param>
</section>
```

Здесь описывается два единичных сервиса SMS-Flash и SMS-Calendar, их комбинация SMS-Flash-Calendar (см. serviceMask), а также комбинация SMS-Flash-Group с выставленным признаком тарификации сообщения-носителя.

2. Информация о компании

СибИнфоКонсалтинг является поставщиком интегрированных технологических решений и услуг в области мобильных коммуникаций и мультимедиа. Мы работаем с такими известными компаниями как Мобильные ТелеСистемы, Panasonic, Hewlett Packard, Sun Microsystems.

3. Контактная информация

Мы будем рады ответить на Ваши вопросы.

Контактная информация:

www: www.eyeline.mobi
e-mail: info@eyeline.mobi

Россия
630055, Новосибирск
ул. М. Джалиля 3/1, оф. 324
Факс: +7 (383) 332-88-17
Телефон: +7 (383) 363-01-39
Временной пояс: GMT +6