

**Multimedia Messaging
Service Center**

m.MMSC

Установка
и
настройка

Этот документ предназначен для использования при настройке серверных продуктов компании ООО «*М.Технолоджис*». Копирование, публикация, модификация и передача третьим лицам данного документа или любой из его частей в какой-либо форме без письменного разрешения от ООО «*М.Технолоджис*» запрещена.

m.MMSC является собственностью ООО «*М.Технолоджис*».

Другие продукты и имена компаний, использованные в этом документе, могут быть собственностью и торговыми марками соответствующих владельцев.

Компания ООО «*М.Технолоджис*» оставляет за собой право изменения продукта, описанного в этом документе, без предварительных уведомлений.

Содержимое этого документа предоставляется «КАК ЕСТЬ». Компания ООО «*М.Технолоджис*» не несёт ответственности за какие-либо убытки и ущерб (в том числе, убытки в связи с недополученной коммерческой прибылью, прерыванием коммерческой или производственной деятельности, утратой деловой информации и иной имущественный ущерб), возникающие в связи с использованием или невозможностью использования программного обеспечения, описанного в данном документе. Компания ООО «*М.Технолоджис*» оставляет за собой право вносить изменения в документ и отзываться его без предварительных уведомлений.

1. Установка программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений m.MMSC»

Установка программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**» (**m.MMSC**) и подготовка его к работе разделяется на три этапа:

1. Установка и настройка среды исполнения;
2. Установка ядра системы из поставляемого дистрибутива;
3. Создание базы данных;
4. Настройка конфигурации сервис-центра.

1.1. Установка и настройка среды исполнения

Перед установкой программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**» необходимо произвести установку среды исполнения **Java**, сервера приложений **Apache Tomcat**, препроцессора **PHP** и т.д., в соответствии со списком, приведенным в [разделе 4.1](#).

Инструкции по установке и настройке этих программных продуктов находятся на официальных сайтах соответствующих компаний и сообществ разработчиков.

1.2. Установка ядра системы

Программное обеспечение платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**» поставляется в виде дистрибутива, представляющий собой запакованный архив. Дистрибутив **m.MMSC** имеет следующую структуру.

Таблица 5.1 - Структура дистрибутива программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**»

Название	Тип	Описание
/mmmc.v1.2	Каталог	Общий каталог дистрибутива
/bin	Каталог	Каталог, содержащий исполняемые модули и скрипты запуска сервера
/m.mmmc.sh	Файл	Скрипт запуска и остановки сервера
/data	Каталог	Каталог служебных данных
/docs	Каталог	Каталог документации
/etc	Каталог	Каталог, содержащий файлы конфигурации
/keystore	Каталог	Каталог ключей, используемых при установке SSL-соединений

/lib	Каталог	Каталог служебных библиотек
/logs	Каталог	Каталог по умолчанию для хранения журналов
/sql	Каталог	Каталог, содержащий скрипт создания базы данных

Архив дистрибутива следует распаковать в каталог **/usr/local/mmsc/**, который следует создать при необходимости.

1.3. Создание базы данных хранилища сообщений

Для создания базы данных необходимо выполнить скрипт **create_database.sql**, поставляемый в дистрибутиве, на командном интерпретаторе сервера **MySQL**. Структура базы данных соответствует описанию в [разделе 4.4](#).

2. НАСТРОЙКА КОНФИГУРАЦИИ

Конфигурация программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**» состоит из одного или более файлов. Ссылка на корневой файл конфигурации указывается в параметрах запуска системы (см. [раздел 6.8](#)). Все конфигурационные файлы используют синтаксис XML.

Изменение конфигурации любого модуля сервиса может быть выполнено ручным редактированием конфигурационных файлов (при помощи любого текстового редактора), или с помощью специального инструмента редактирования конфигураций **m.Control** разработки компании ООО «М.Технолоджис». После изменения конфигурации требуется рестарт соответствующих модулей.

Расположение и имена файлов конфигурации не имеют значения. По умолчанию файлы конфигурации располагаются в подкаталоге **/etc/config**, находящемся в главном каталоге сервиса (**/usr/local/mmsc/**).

Основной файл конфигурации (как правило, имеет имя **m.MMSC.config.xml**), располагается в каталоге **/etc/config/main** и содержит корневой документ **<MMSC>**, который состоит из частей, приведенных в таблице 6.1.

Таблица 6.1 - Элементы, вложенные в корневой элемент **<MMSC>**

Элемент XML	Описание
<COMMON>	Секция описания общих параметров окружения и системы
<DATABASE>	Конфигурация хранилища сообщений
<MSGSTORAGE>	Путь хранения сообщений
<MYSQL>	Интерфейс взаимодействия с БД
<NOTIFICATORS>	Интерфейсы межмодульного взаимодействия
<SMSGATE>	Интерфейс взаимодействия с SMS-гейтом
<PPG>	Интерфейс взаимодействия с Push Proxy Gateway
<PARAMS>	Общие параметры и ограничения для сообщений
<STATISTICS>	Элемент включения статистики
<MONITORING>	Настройки модуля мониторинга доставки
<INTERFACES>	Спецификации реализаций интерфейсов
<INTERFACE>	Именованная спецификация реализации
<MO_SIDE>	Конфигурация модуля взаимодействия с мобильными

Элемент XML	Описание
	терминалами
<SERVICES>	Конфигурация интерфейсов взаимодействия
<FILTERS>	Конфигурации процессоров событий обработки сообщений
<FILTER>	Конфигурация процессора обработки сообщения
<FROM>	Фильтр по отправителю сообщения для обработки
<TO>	Фильтр по адресатам сообщения подлежащего обработке
<IMPLEMENTATION>	Конфигурация реализации типа
<MODIFY_ADDRESSES>	Список модификаций адресов
<MODIFICATION>	Реализация модификация адресов
<LEGACY>	Общая конфигурация процессора «зависших» сообщений
<IMPLEMENTATION>	Типизированная конфигурация
<SP_SIDE>	Конфигурация модуля взаимодействия с сервис провайдерами
<MMSC_SIDE>	Конфигурация модуля взаимодействия с внешними MMSC
<EMAIL_SIDE>	Конфигурация модуля передачи сообщений электронной почты
<SMTP>	Конфигурация интерфейса с сервером отправки сообщения
<IMAP>	Конфигурация интерфейса взаимодействия с сервером входящих сообщения
<PAYMENT>	Конфигурация ссылки на реализацию интерфейса тарификации для тех модулей, которые не имеют собственного интерфейса
<ROUTES>	Конфигурация межмодульной маршрутизации сообщений
<TARIFICATIONS>	Конфигурация системы тарификации
<NODES>	Перечень узлов, образующих кластер
<NODE>	Описание узла, входящего в кластер

Элемент XML	Описание
<ADMIN>	Конфигурация управляющего интерфейса для отладки и настройки сервиса.

Элемент **<ADMIN>** настоящей версии не используется, оставлен для обратной совместимости. Этот элемент всегда должен присутствовать в файле конфигурации, и иметь следующий вид:

```
<ADMIN host="localhost" port="8765" login="admin" password="adm"/>
```

2.1. Секция описания общих параметров окружения и системы

Секция описания общих параметров окружения и системы задается элементом **<COMMON>** и состоит из следующих элементов:

- **<DATABASE>** – Элемент задает конфигурацию хранилища сообщений;
- **<MSGGATE>** – Настройки соединения с SMS-гейтом (*SMPP-сервером*), элемент может отсутствовать;
- **<PPG>** – Настройки соединения с *Push Proxy Gateway* и отсылки уведомлений получателям;
- **<PARAMS>** – Общие параметры сервиса;
- **<STATISTICS>** – Параметры ведения статистики работы.
- **<MONITORING>** – Параметры мониторинга доставки сообщений.

Элемент описания конфигурации хранилища **<DATABASE>** является обязательным, и содержит следующие дочерние элементы:

- **<MYSQL>** – Обязательный элемент. Задает конфигурацию взаимодействия с сервером СУБД *MySQL*. Имеет следующие атрибуты:
 - **url** – спецификация адреса подключения (обязательный атрибут);
 - **user** – имя пользователя БД;
 - **password** – пароль.
- **<MSGSTORAGE>** – Задает место хранения сообщений на диске (в файловом виде). Имеет единственный обязательный атрибут **path**, который указывает путь к каталогу хранения.

По пути, указанному в атрибуте **path**, создаются подкаталоги /email_msg, /immsc_msg, /mobile_msg, /sp_msg, предназначенные для хранения сообщений, обрабатываемых соответствующими модулями. В каждом из этих каталогов создаются подкаталоги, соответствующие времени создания сообщения. Например, сообщения, направленные модулю взаимодействия с сервис-провайдерами (**SP_SIDE**) 13 мая 2007 года в 17 часов, будут расположены в каталоге /sp_msg/2007.05.13/17/.

При записи сообщений в хранилище в виде файлов, в таблицу 'msg' базе данных (в поле 'mms') записывается строка ссылки на файл, помеченная префиксом **FILE:..** Например, «FILE:/incomming/mmsmsgs/sp_msg/2007.05.13/17/1234.mms». Если элемент <MSGSTORAGE> отсутствует, то сообщения будут храниться непосредственно в этом поле таблицы.

- **<NOTIFICATORS>** – Задаёт конфигурацию межмодульной коммуникации, которая используется, если модули обработки сообщений разнесены на разные сервера в сети. В этом случае в конфигурации, размещаемой на каждом сервере, прописывается адрес, на котором будет открыт прослушивающий порт UDP и адреса других модулей. Элемент является обязательным.

Адрес каждого модуля задается вложенным элементом **<NOTIFICATOR>** (по одному на каждый модуль). Элемент имеет следующие обязательные атрибуты:

- **type** – тип модуля обработки. Возможны варианты: MMSC, SP, MO, EMAIL. В конфигурации каждого сервера должны быть перечислены адреса всех четырех типов модулей.
- **host** – задает имя или IP-адрес хоста, на котором расположен модуль обработки, ожидающий пакеты уведомления.
- **port** – номер порта, на который следует посылать пакеты уведомления.

Элемент **<MSGGATE>**, задающий конфигурацию взаимодействия с **SMPP-сервером**, не обязателен. Он может быть включен в секцию **<COMMON>**, если модуль **Legacy** использует режим доставки web/WAP, поскольку в этом случае SMS-гейт или PPG используются для отправки абоненту URL на web/WAP-представление MMS-сообщения. Кроме этого, SMS-гейт используется модулем мониторинга для отправки абонентам диагностических SMS.

Если в секции **<COMMON>** определен элемент **<MSGGATE>**, то для доставки будет использован он, а в противном случае – **<PPG>**. Элемент **<MSGGATE>** имеет следующие атрибуты:

- **url** – URL для передачи SMPP-запросов;
- **username** – Имя пользователя;
- **password** – Пароль для авторизации подключения.

Элемент **<PPG>**, задающий конфигурацию взаимодействия с **Push Proxy Gateway**, обязателен, если на сервере реализован модуль взаимодействия с мобильными терминалами (MO), а также для нужд модулей **Legacy** и **DM**. Элемент имеет следующие атрибуты, являющиеся обязательными:

- **defaultMSISDNPrefix** – Префикс MSISDN, который поставляется в начало номера, если номер адресата короче, чем длина номера по умолчанию. Сумма длины номера и префикса по умолчанию должна быть равна длине номера по умолчанию.
- **defaultMSISDNLength** – Длина MSISDN по умолчанию.
- **url** – Адрес PPG.
- **reportURL** – Адрес и порт **m.MMSC**, на котором модуль мониторинга ожидает статус доставки отправленного через PPG уведомления.
- **deliveryDays** – Время жизни (актуальности) Push-сообщения в сутках. Не должно быть больше, чем время срабатывания LEGACY-обработки.

Обязательный элемент **<PARAMS>**, задающий общие настройки сервиса, содержит следующие дочерние элементы, настраивающие конфигурационные значения параметров:

- **<MESSAGE>** – задает параметры обработки MMS-сообщений сервером. Имеет следующие атрибуты:
 - **maxsize** – Максимально допустимый размер сообщения в КБ.

- **transactionPostfix** – Постфикс идентификатора транзакции сообщения. Используется при маршрутизации отчетов о доставке.
- **maxexp** – Максимальное время жизни сообщения на сервис-центре в сутках. Если параметр сообщения, определяющий время жизни, не задан или больше максимального, то устанавливается максимум сервис-центра.
- **closeOnResponse** – поведение MMSC при ответе на запрос. Если «true», то после каждого ответа соединение с клиентом закрывается, иначе оно держится открытым до закрытия с клиентской стороны.
- **duplicate_timeout** – Максимальное время (в секундах), в течение которого пришедшее повторно от отправителя сообщение считается дубликатом и не обрабатывается, а отправителю повторно отправляется уведомление о успешном получении сообщения. Если параметр не задан или равен 0, обработка дубликатов не производится, каждое сообщение считается уникальным.

Элемент **<STATISTICS>**, если присутствует, задает конфигурацию ведения статистики работы MMS-центра. Элемент имеет следующие обязательные атрибуты:

- **filename** – путь и имя файла, в котором сохранять работу MMSC.
- **period** – период сбора статистических данных (в секундах).

2.2. Конфигурация модуля мониторинга доставки

Параметры модуля мониторинга доставки сообщений описываются в главном файле конфигурации в секции **<COMMON>**. Все параметры мониторинга располагаются внутри элемента **<MONITORING>**. Элемент имеет следующую структуру:

```
<MONITORING>
  <GET times="num" roaming_only="true|false" message="sms_text"/>
  <CONFIRM times="num" roaming_only="true|false" message="sms_text"/>
  <ROAMING header="REGION_HEADER_NAME" valuemask="regexp"/>
  <LBS url="url"/>
  <VRADIUS IPattribute="attr_name" MSISDNattribute="attr_name"
    GGSNattribute="attr_name" SGSNattribute="attr_name">
    <PRIMARY host="ip" port="port" secret="secret" user="username"/>
    <SECONDARY host="ip" port="port" secret="secret" user="username"/>
  </VRADIUS>
</MONITORING>
```

```
</VRADIUS>  
<OPTIONS locateRecipient="onLegacy|onPushDelivered"  
         locateSender="true|false" />  
</MONITORING>
```

Элементы **<GET>** и **<CONFIRM>**, имеющие одинаковую структуру, настраивают поведение модуля мониторинга на отправку абоненту (получателю) указанного SMS при обнаружении множественных попыток от его мобильного терминала получить сообщение или передать уведомление о его получении, соответственно. Элементы имеют следующие атрибуты:

- **times** – количество повторных попыток мобильного терминала, приводящих к отправке SMS.
- **roaming_only** – Указывает условие отправки SMS. Если **true**, то SMS будет отправляться только для абонентов в роуминге (для них должно быть истинным условие, указанное в элементе **<ROAMING>**, см. ниже). В противном случае сообщение будет отправлено без анализа нахождения абонента в роуминге.
- **message** – Текст отправляемого сообщения.

Примечание. Если в секции **<COMMON>** указан **<MSGGATE>**, то сообщение абоненту будет отправлено через него, в противном случае оно будет отправлено через **<PPG>**.

Элементы **<GET>** и **<CONFIRM>** не являются обязательными. Если они не указаны, подсчет числа попыток загрузки и подтверждения загрузки сообщения не ведется. Если атрибут **message** не указан, сообщение отправлено не будет.

Элемент **<ROAMING>** указывает правило определения нахождения абонента-получателя в роуминге (в сети другого оператора). Для корректного определения нахождения абонента в роуминге WAP GW, обслуживающий абонента, должен подставлять в HTTP-запросы, идущие от абонента, специальный заголовок, содержащий название региона, в котором находится абонент. Элемент имеет следующие атрибуты:

- **header** – название HTTP-заголовка из запроса, содержащего название региона, в котором находится абонент.
- **valuemask** – регулярное выражение, которому должно удовлетворять значение заголовка, чтобы абонент считался находящимся в роуминге.

Элемент **<LBS>** описывает параметры подключения к платформе определения местоположения абонента **m.LBSP** и имеет всего один атрибут – URL, запрашиваемый у платформы. В ответе от **m.LBSP** всегда возвращается статус успешности обработки запроса и данные о местоположении абонента (*LAC, CID, долгота, широта*), если они установлены. Подробнее формат URL см. в документе «Платформа определения и расчета географических координат абонентов **m.LBSP**. *Описание интерфейса внешних клиентов LBS API*».

Параметры установления географического местоположения абонента-отправителя и получателя посредством обращений к платформе **m.LBSP** задаются атрибутами элемента **<OPTIONS>**. Элемент имеет следующие атрибуты:

- **locateRecipient** – Условия, при которых определять местоположение абонента-получателя. Имеет два возможных значения. Значение `onLegacy` предписывает определять местоположение получателя только в случае постановки предназначенного ему сообщения на Legacy-доставку, а значение `onPushDelivered` – при получении от мобильного терминала подтверждения доставки.
- **locateSender** – Требуется или нет определять местоположение абонента-отправителя (`true` или `false`).

Результаты определения местоположения абонентов, полученные от **m.LBSP**, сохраняются в базе данных.

Элемент **<VRADIUS>** описывает параметры подключения к AAA-серверу **m.VRadius** для установления факта активации PDP-контекста мобильным терминалом абонента. Элемент имеет следующие атрибуты:

- **IPAttribute** – название атрибута, в котором будет получен IP-адрес мобильного терминала абонента.
- **MSISDNAttribute** – название атрибута, в котором будет передан MSISDN-адрес мобильного терминала абонента.
- **GGSNAttribute** – название атрибута, в котором будет получен GGSN, используемый мобильным терминалом абонента для доступа в сеть.

- **SGSNAttribute** – название атрибута, в котором будет получен SGSN, используемый мобильным терминалом абонента для доступа в сеть.

В теле элемента **<VRADIUS>** содержатся два одинаковых по структуре вложенных элемента **<PRIMARY>** и **<SECONDARY>**, описывающих подключение к основному и вспомогательному VRADIUS-серверам. Элементы имеют следующие атрибуты:

- **host** – IP-адрес хоста, на котором функционирует сервер **m.VRADIUS**.
- **port** – номер порта, прослушиваемый сервером (обычно – 3813).
- **secret** – значение секрета, разделяемого **m.MMSC** и сервером **m.VRADIUS**.
- **user** – имя пользователя для подключения.

Подробнее описание API VRADIUS см. в документе «Сервер авторизации и аккаунтинга **m.VRADIUS**. Интерфейс приложений (API)».

Пример конфигурации общих параметров сервиса и модуля мониторинга:

```
<COMMON>

<STATISTICS filename="/usr/local/mmsc/statistics.log" period="60"/>

<DATABASE>
  <MYSQL url="jdbc:mysql://localhost:3306/mmsc"
    user="mms" password="mms"/>

  <MSGSTORAGE path="/usr/local/mmsc/data/mmsmsgs">

    <NOTIFICATORS>
      <NOTIFICATOR type="MMSC" port="63210" host="localhost"/>
      <NOTIFICATOR type="SP" port="62211" host="localhost"/>
      <NOTIFICATOR type="MO" port="62212" host="localhost"/>
      <NOTIFICATOR type="EMAIL" port="62213" host="localhost"/>
    </NOTIFICATORS>
  </DATABASE>

  <PPG defaultMSISDNPrefix="7812" defaultMSISDNLength="11"
    url="http://10.255.20.99:8880/PGW"
    reportURL="http://10.255.20.99:9990/ppgReport" deliveryDays="3"/>
  <MSGATE url="http://10.20.30.250:13013/cgi-bin/sendsms"
    password="xxx" username="user"/>

  <PARAMS>
    <MESSAGE maxsize="1000" transactionPostfix="_MOBITECH" maxexp="3"
      closeOnResponse="true"/>
  </PARAMS>

  <MONITORING>
    <GET times="2" roaming_only="false" message="Перезагрузите телефон"/>
```

```

<CONFIRM times="2" roaming_only="false"
    message="Перезагрузите телефон"/>
<!-- Абонент будет считаться находящимся в роуминге, если заголовок
    «X_GAPS_REGION» будет пуст-->
<ROAMING header="X_GAPS_REGION" valuemask="^$"/>

<LBS url="https://10.255.2.99:8989/internal/app.php?uid=mmsc&pwd=password&mode=7"/>

<VRADIUS IPAttribute="IP" MSISDNAttribute="MSISDN"
    GGSNAttribute="GGSNAddress" SGSNAttribute="SGSNAddress">
    <PRIMARY host="10.255.2.9" port="3813" secret="123" user="mmsc"/>
    <SECONDARY host="10.255.2.7" port="3813" secret="123" user="mmsc"/>
</VRADIUS>
<OPTIONS locateRecipient="onLegacy" locateSender="true" />
</MONITORING>

</COMMON>

```

2.3. Конфигурация интерфейсов взаимодействия с внешними источниками сообщений

Этот набор параметров предназначен для связи интерфейсов, которые программное обеспечение платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**» использует при взаимодействии с внешними источниками сообщений, с классами, реализующими функциональность этих интерфейсов. Для каждого интерфейса определен свой формат кодирования MMS-сообщений для передачи через интерфейс.

Каждый сервер модуля обработки сообщений должен ссылаться на один из идентификаторов, которые перечислены в списке интерфейсов.

Секция конфигурации интерфейсов взаимодействия задается элементом **<INTERFACES>**. Описание реализации каждого интерфейса описывается вложенным элементом **<INTERFACE>**. Этот элемент содержит два обязательных атрибута:

- **id** – Идентификатор реализации, на который будут ссылаться реализации серверов модулей обработки.
- **class** – Имя Java-класса, реализующего интерфейс.

Стандартный пример описания реализаций выглядит следующим образом.

```

<INTERFACES>
  <INTERFACE id="EAIF"
    class="com.mobitech.mmsc.interfaces.eaif.eaifInterfaceImpl"/>
  <INTERFACE id="WWW"
    class="com.mobitech.mmsc.interfaces.mobile.mobileInterfaceImpl"/>
  <INTERFACE id="MM4"
    class="com.mobitech.mmsc.interfaces.eaif.eaifInterfaceImpl"/>
  <INTERFACE id="MM7"
    class="com.mobitech.mmsc.interfaces.mm7.VASPIInterface"/>
</INTERFACES>

```

Эта конфигурация должна использоваться во всех стандартных реализациях сервисов.

2.4. Конфигурация кластеризации

Конфигурация кластера по обработке сообщений, организованного из нескольких MMSC, задается необязательным элементом **<NODES>**, в котором перечисляются узлы, образующие кластер. Каждый узел описывается элементом **<NODE>**, имеющим три обязательных атрибута:

- **id** – Идентификатор узла. Произвольная текстовая строка, записываемая в базу данных для каждого сообщения, принятого узлом к обработке. Предназначена для идентификации сообщений, обрабатываемых узлом кластера.
- **ip** – IP-адрес сервера, на котором функционирует узел кластера.
- **port** – Порт для связи с узлом кластера (для проверки его работоспособности).

Если используется кластеризация, то в кластер рекомендуется включать не более двух узлов во избежание конфликтов при обработке сообщений отказавшего узла.

Если кластер не организуется, элементом **<NODES>** задавать не требуется (в этом случае идентификаторы узлов для сообщений не задаются).

2.5. Конфигурация модулей обработки сообщений

Модули обработки имеют схожую реализацию и внутреннюю структуру (см. [раздел 2.3](#)). Поэтому их конфигурация имеет одинаковую структуру.

Конфигурация каждого модуля определяется элементом **<XX_SIDE>**, где префикс «XX» зависит от типа модуля. Допустимы следующие элементы конфигурации:

- **<SP_SIDE>** – конфигурация модуля обработки сообщений от/для контент-провайдеров;
- **<MO_SIDE>** – конфигурация модуля обработки сообщений от/для мобильных терминалов;
- **<MMSC_SIDE>** – конфигурация модуля обработки сообщений от/для внешних MMSC;
- **<EMAIL_SIDE>** – конфигурация модуля обработки сообщений от/для электронной почты.

Описание конфигурации каждого модуля обработки, за исключением **<EMAIL_SIDE>**, в свою очередь, состоит из трех разделов:

- **<SERVICES>** – Наличие этого элемента обязательно, поскольку он задает конфигурацию сервиса приема и передачи сообщений модулем. Элемент имеет атрибут **routing**, указывающий на имя файла конфигурации маршрутизации сообщения между провайдерами внутри модуля (см. [раздел 6.8](#)).

Раздел может включать в себя конфигурацию нуля или более серверов модуля, задаваемую элементом **<SERVER>**. Элемент имеет следующие атрибуты:

- **id** – строка, задает уникальный идентификатор сервера. Используется для привязки клиентских интерфейсов (см. [раздел 6.8.2](#));
 - **interface** – строка, ссылка на идентификатор интерфейса, реализующего протокол работы сервера. Должна соответствовать id одного из интерфейсов, перечисленных в секции **<INTERFACES>** (см. [раздел 6.3](#));
 - **host** – строка, имя или IP-адрес интерфейса привязки сервера;
 - **port** – число (от 1 до 65535), порт привязки сервера;
 - **baseURL** – строка в формате URL, задающая внешний адрес сервера. Значение используется в ссылках, отправляемых клиентам.
 - **useSSL** – логическое значение, определяющее, следует ли использовать шифрованное соединение по SSL.
- **<LEGACY>** – Задает конфигурацию LEGACY-процессора альтернативной доставки сообщений. Каждый модуль обработки сообщений может иметь один процессор альтернативной доставки сообщений. Текущая реализация содержит LEGACY-процессоры двух типов: WEB и REROUTER. Конфигурация LEGACY-процессора зависит от его типа, но у всех вариантов есть общая часть – задающий элемент **<LEGACY>** и вложенный элемент **<IMPLEMENTATION>**.

Элемент конфигурации **<LEGACY>** имеет следующие атрибуты:

- **deliveryTimeout** – число, задающее время (в минутах) пребывания сообщения в одном из заданных статусов, перед началом его отправки через LEGACY.

- **status** – строка, задающая список из нуля или более значений статусов, разделенный запятыми. Процессор отслеживает только те сообщения, статус которых совпадает с перечисленными^{1*}. Список возможных значений статуса:

- wait – сообщение ожидает отправки
- notified – оповещение о сообщении
отправлено
- failed – ошибка доставки сообщения
- sent – сообщение отправлено, но его
доставка не подтверждена
- delivered – сообщение доставлено,
доставка подтверждена адресатом
- legacy – сообщение отправлено на
доставку через LEGACY.

- **type** – строка, определяющая тип реализации LEGACY-процессора. Возможные значения – WEB, REROUTER.

Для всех типов LEGACY-процессора возможно выполнение фильтрации сообщений и модификации адресов и статуса сообщения.

Параметры фильтрации сообщения описываются вложенным элементом **<FILTER_ADDRESSES>**. Этот элемент содержит набор (нуль или более) условий сравнения адреса отправителя (поля **from:**) и адреса получателя (поля **to:**, **cc:**, **bcc:**). Условия фильтрации по отправителю задаются элементом **<FROM>**, а по получателю – элементом **<TO>**. Синтаксис задания условий фильтрации адресов см. в [разделе 6.6](#).

Сообщения проходят фильтр и поступают в Legacy-обработку, если фильтр отсутствует (не задан), или если указанные адреса отправителя и получателя соответствуют заданным выражениям. Если условия **<FROM>** или **<TO>** не заданы, то соответствующее поле сообщения не проверяется.

^{1*} В качестве индикатора проблемы прямой доставки рекомендуется использовать статусы: wait, notified и, возможно, sent. Использование статуса failed может привести к множественной повторной отправке.

Если какое-либо сообщение, отправленное для legacy-обработки, не прошло фильтрацию, соответствующее уведомление фиксируется в журнале **m.MMSC**.

Пример настройки фильтра сообщений для legacy-обработки:

```
<FILTER_ADDRESSES>
  <FROM masktype="REGEXP" type="PLMN"
    value="^[+]? (7|8) (812|921) ([0-9]{7}) $" />
  <TO masktype="REGEXP" type="PLMN"
    value="^[+]? (7|8) (812|921) ([0-9]{7}) $" />
</FILTER_ADDRESSES>
```

В данном случае на обработку попадут только те сообщения, которые отправлены с MSISDN *СЗФ ОАО МегаФон* на MSISDN *СЗФ ОАО МегаФон* (сообщения **MOMT**). Остальные сообщения будут помечены как **legacy**, но обработаны не будут.

Параметры модификации сообщения, принятого в legacy-обработку задаются с помощью вложенных элементов **<MODIFY_ADDRESSES>** и входящего в него списка операций **<MODIFICATION>**.

Элемент **<MODIFY_ADDRESSES>** имеет два атрибута: **addr_type**, определяющий тип модифицируемого адреса (возможные значения – **to** и **from**), а также атрибут **msg_states**, указывающий, в каком состоянии должно находиться сообщение, чтобы подвергнуться модификации (указывается подмножество статусов, перечисленных в атрибуте **status** родительского элемента **<LEGACY>**) Атрибут **msg_states** не обязательный. Если не указан, модификации применяются ко всем сообщениям.

Вложенный элемент **<MODIFICATION>** описывает модификацию, производимую с адресом сообщения указанного типа. Имеет следующие атрибуты:

- **msg_status** – строка, определяющая, в каком состоянии должно находиться сообщение (из тех, что перечислены в атрибуте **msg_states** родительского элемента **<MODIFY_ADDRESSES>**), чтобы к нему была применена данная модификация адреса. Атрибут не обязательный. Если не указан, модификация применяется ко всем сообщениям.
- **pattern** – строка, определяющая маску – регулярное выражение, по которому происходит поиск в строковом представлении адреса. Выражение может быть сгруппировано (при помощи скобок) для выделения необходимых частей адреса, которые будут использованы для подстановки их в результат модификации.

- **replace** – строка, в которую подставляется найденное выражение. Должна содержать плейсхолдеры **\$N**, соответствующие по номеру указанной группы регулярного выражения **pattern**.

Процедура модификации извлекает из адреса выражение, соответствующее заданной регулярной маске **pattern**, выполняет поиск всех ее частей, и подставляет их вместо соответствующих плейсхолдеров **\$1** в строке **replace**.

Пример описания процедуры модификации адреса:

```
<MODIFY_ADDRESSES addr_type="to" msg_states="notified">
  <MODIFICATION msg_status="notified"
    pattern="^[+]? (7|8) (812|921) ([0-9]{7}) $" replace="+7$2$3"/>
  <MODIFICATION msg_status="notified"
    pattern="^[+]? (7|8) (812) ([0-9]{7}) $" replace="+7921$3"/>
  <MODIFICATION msg_status="notified"
    pattern="^[+]? [0-9]{7} $" replace="+7921$1/TYPE=PLMN"/>
</MODIFY_ADDRESSES>
```

Первая инструкция **<MODIFY_ADDRESSES>** выделяет из адреса получателя код страны, код региона и семизначный номер абонента, и формирует MSISDN, состоящий из +7, кода региона и семизначного номера абонента. Вторая инструкция формирует MSISDN, начинающийся с +7921 и семизначного номера абонента. Третья инструкция аналогична второй, но добавляет в конец модифицированного адреса получателя тип адреса (TYPE=PLMN).

Реализация вложенного элемента **<IMPLEMENTATION>**, описывающего поведение процессора, зависит от типа процессора (WEB или REROUTER).

LEGACY-процессор с типом «WEB» осуществляет формирование web/WAP-представления сообщения и публикацию его на web-сервере с отсылкой уведомления получателю в виде SMS. Для этого используется **<MSGGATE>** или **<PPG>**, описанный в секции **<COMMON>**.

Конфигурация процессора преобразования LEGACY-WEB включает в себя элемент **<IMPLEMENTATION>**, содержащий атрибуты:

- **storePath** – адрес хранения контекста web-сервера, в который размещаются подкаталоги web- и WAP-представления MM-сообщения.
- **baseURL** – внешний адрес контекста web-сервера в котором располагаются web-/WAP-представления MMS-сообщений.

LEGACY-процессор с типом «REROUTER» осуществляет передачу сообщения на другой модуль обработки.

Конфигурация процессора преобразования LEGACY-REROUTER включает в себя элемент **<IMPLEMENTATION>**, содержащий атрибут **targetSide**, указывающий тип модуля обработки, на который будет переправлено сообщение. Возможны следующие варианты передачи: “EMAIL_SIDE”, “SP_SIDE”, “MO_SIDE” и “MMSC_SIDE”.

- **<FILTERS>** – задает конфигурацию фильтров обработки событий, генерируемых процессором сообщений модуля обработки сообщений.

Механизм реализации фильтров позволяет включать по одному фильтру на событие каждого типа. Если в конфигурации встречается несколько фильтров для одного события – будет использован только последний.

Элемент конфигурации **<FILTERS>** имеет следующие атрибуты:

- **type** – строка, задающая тип фильтра. Доступны два типа фильтров – генератор ответа (RESPONDER), и фильтр перенаправления обработки (REROUTER), аналогичный LEGACY-продюсеру LEGACY-REROUTER.
- **event** – строка, определяющая тип события, которое будет обрабатываться фильтром. В текущей версии сервера фильтры могут быть подключены к событиям двух типов:

- **ROUTE_ERROR** –

адресат сообщения указан неверно, в конфигурации не определен путь доставки, соответствующий параметрам сообщения.

- **REJECTED** –

сообщение отклонено получателем.

Конфигурация фильтров включает в себя список адресов получателей (**<TO>**) и отправителей (**<FROM>**). Фильтр будет применен к событию, только если адреса сообщения попадают в набор **<TO>** и **<FROM>** или не задано ни одного **<TO>** или **<FROM>** соответственно. Синтаксис задания списков адресов см. в [разделе 6.6](#).

Фильтры, как и LEGACY-процессоры, имеют механизм модификации адреса **<MODIFY_ADDRESSES>**. Специфика реализации фильтра определенного типа задается

элементом **<IMPLEMENTATION>**, синтаксис которого зависит от типа фильтра (REROUTER или RESPONDER).

Для фильтров типа REROUTER элемент **<IMPLEMENTATION>** имеет единственный атрибут **targetSide**, указывающий тип модуля обработки, на который будет переправлено сообщение. Возможны следующие варианты передачи: "EMAIL_SIDE", "SP_SIDE", "MO_SIDE" и "MMSC_SIDE".

Для фильтров автоматического ответа типа RESPONDER элемент **<IMPLEMENTATION>** имеет следующие атрибуты:

- **subject** – строка, заголовок сообщения автоответа;
- **from** – строка, адрес отправителя. Возможно указание адреса в виде произвольной строки в соответствии со стандартом *MMS Addressing Model*. Например, возможны следующие варианты задания адреса:

from = "Megafon Pro"

from = "0401234567/TYPE=PLMN"

from = "+358501234567/TYPE=PLMN"

from = "Joe User <joe@user.org>"

(без учета кватирования спецсимволов XML)

from = "FEDC:BA98:7654:3210:FEDC:BA98:7654:3210/TYPE=IPv6"

from = "195.153.199.30/TYPE=IPv4"

- **file** – строка, задающая имя файла, в котором хранится текст сообщения
- **encoding** – строка, указывающая название кодировки содержания файла. Названия кодировок должны соответствовать *IANA Charset Registry*.

Пример конфигурации модуля обработки MMS-сообщений от (для) мобильных терминалов:

```
<MO_SIDE active="true">
  <SERVICES routing="/etc/config/providers/mo/mo.configuration.xml">
    <SERVER id="MO_WWW" interface="WWW" host="0.0.0.0" port="19531"
      baseURL="http://84.17.12.42:19531/mms/msg" useSSL="false"/>
  </SERVICES>
  <LEGACY deliveryTimeout="1" status="wait,notified" type="WEB">
    <IMPLEMENTATION storePath="/opt/htdocs/webapps/legacy/"
      baseURL="http://121.121.121.50:9999/legacy/" />
  </LEGACY>
  <FILTERS>
    <FILTER type="RESPONDER" event="ROUTE_ERROR">
      <FROM masktype="REGEXP" type="PLMN" value="[+]?79219343811"/>
    </FILTER>
  </FILTERS>
</MO_SIDE>
```

```

<FROM masktype="REGEXP" type="PLMN" value="[+]?79217975200"/>
<FROM masktype="REGEXP" type="PLMN" value="[+]?79219145097"/>
<TO masktype="REGEXP" type="PLMN" value="[+]?[0-9]{7}$"/>
<IMPLEMENTATION subject="Короткий номер не пройдет" from="Мегафон"
file="response.txt" encoding="cp1251"/>
</FILTER>
<FILTER type="REROUTER" event="REJECTED">
<FROM masktype="REGEXP" type="PLMN" value="[+]?79219361646"/>
<FROM masktype="REGEXP" type="PLMN" value="[+]?79219345813"/>
<TO masktype="REGEXP" type="PLMN" value="[+]?[0-9]{11}$"/>
<IMPLEMENTATION targetSide="EMAIL_SIDE">
<MODIFY_ADDRESSES addr_type="to">
<MODIFICATION pattern="[0-9]{7,11}$"
replace="&quot;Forwarding to Mobile phone
$1&quot;&lt;$1@nwgsm.ru&gt;"/>
</MODIFY_ADDRESSES>
<MODIFY_ADDRESSES addr_type="from">
<MODIFICATION pattern="[0-9]{7,11}$" replace="$1@nwgsm.ru"/>
</MODIFY_ADDRESSES>
</IMPLEMENTATION>
</FILTER>
</FILTERS>
</MO_SIDE>

```

В отличие от других модулей **<XX_SIDE>**, настройка модуля **<EMAIL_SIDE>** содержит в своем теле следующие элементы:

- **<SMTP>** – задает конфигурацию подключения к почтовому серверу для отсылки MMS на e-mail. Элемент содержит следующие атрибуты:
 - **host** – имя или IP-адрес хоста, на котором работает SMTP-сервер;
 - **port** – порт, используемый SMP-сервером;
 - **domain** – доменное имя SMTP-сервера.
- **<IMAP>** – задает конфигурацию подключения к почтовому серверу для получения MMS, полученных по e-mail. . Элемент содержит следующие атрибуты:
 - **host** – имя или IP-адрес хоста, на котором работает POP3/IMAP-сервер;
 - **port** – используемый сервером порт;
 - **user** – имя пользователя для полключения;
 - **password** – пароль пользователя для подключения;
 - **protocol** – используемый протокол (*pop* или *imap*);
 - **mbox** – имя папки ходящих сообщений.
- **<PAYMENT>** – задает ссылку на правила тарификации MMS-сообщений, отправленных (полученных) на e-mail. Атрибуты элемента аналогичны рассмотренным в [разделе 6.8.2](#).

Элемент **<EMAIL_SIDE>** не содержит ссылок на файл внутримодульной маршрутизации и список подключаемых провайдеров.

Пример:

```
<EMAIL_SIDE>
  <SMTP host="mail" port="25" domain="server.com"/>
  <IMAP host="10.255.2.245" port="110" user="user" password="pwd"
    protocol="pop3|imap" mbox="INBOX"/>
  <PAYMENT ref="mail" base="222.0" discount="byContent"
    origID="MOBILE" termID="EMAIL"/>
</EMAIL_SIDE>
```

2.6. Конфигурация межмодульной маршрутизации

Маршрутизация сообщений между модулями центра задается элементом **<ROUTES>**, который может быть реализован в виде ссылки на внешний файл с помощью атрибута **cfg**, или в случае отсутствия ссылки, должен иметь тело, имеющее следующую структуру:

```
<ROUTES>
  <ROUTE side="side_type" default="true|false" type="address"
    source="side_type">

    <TO masktype="VALUE|REGEXP" type="PLMN|IPv4|IPv6|EMAIL"
      value="to_value"/>

    ...
    <FROM masktype="VALUE|REGEXP" type="PLMN|IPv4|IPv6|EMAIL"
      value="from_value"/>

    ...
  </ROUTE>
  ...
</ROUTES>
```

Корневой элемент **<ROUTES>** включает в свой состав один или более элементов **<ROUTE>**, указывающих модуль, в котором должна выполняться обработка сообщения. Атрибуты элемента:

- **side** – строка, задающая имя модуля обработки;
- **default** – логическое значение, определяющее, указывает ли элемент на модуль обработки по умолчанию;
- **type** – строка, тип маршрутизации. Возможны следующие варианты типа маршрутизации:

- **"address"** – маршрутизация по адресу

- **"source"** – маршрутизация в соответствии с типом сообщения и модулем,

сформировавшим текущее или родительское сообщение.

В текущей версии реализован только тип **"address"**.

- **source** – необязательный параметр. Строка, задающая имя модуля, откуда должно поступить сообщение, чтобы к нему применялось это правило. Набор возможных значений совпадает с набором значений атрибута **side**.

Модуль назначения (источника), который указывается в атрибуте **side (source)** может иметь одно из нижеперечисленных значений:

- **"SP"** – модуль подключения сервис-провайдеров;
- **"EMAIL"** – модуль передачи сообщений электронной почты;
- **"MOBILE"** – модуль взаимодействия с мобильными терминалами;
- **"interMMSC"** – модуль взаимодействия с внешними MMSC-серверами;
- **"ERROR"** – модуль формирования ошибки маршрутизации (рекомендуется использовать в качестве направления по умолчанию).

Элемент с указанием одного и того же типа модуля может встречаться неограниченное число раз. Определяющую роль играет последовательность перечисления модулей в списке. Алгоритм выбора маршрута определяет первое правило, подходящее по источнику (если указан) и пулам адресов.

Если ни один модуль не определен, то используется модуль по умолчанию, который должен быть определен в любом случае. Модуль «по умолчанию» определяется значением **"true"** атрибута **default**.

При маршрутизации сообщения по адресу, сообщение проверяется на соответствие адресному пулу по отправителю (задается элементами **<FROM>**) и получателю (задается элементами **<TO>**). Элементы **<TO>** и **<FROM>**, задающие адреса, имеют следующие атрибуты:

- **masktype** – строка, задает тип определения адреса.

Возможны варианты:

- **"VALUE"** – при определении значения адреса;

- **"REGEXP"** – при определении регулярной маски адреса.

- **type** – строка, определяет тип адреса. Возможные варианты значений соответствуют стандартам определения адресов (**"PLMN"**, **"IPv4"**, **"IPv6"** или **"EMAIL"**). Может отсутствовать.

- **value** – строка, определяет значение адреса или регулярную маску, в зависимости от значения атрибута **masktype**. Для сервис-провайдера, если не задан атрибут **type**, в качестве значения можно указывать не только его короткий номер или IP-адрес, но и его **VASID** (идентификатор услуги), задаваемый в описании провайдера (см. [раздел 6.8](#)).

Пример настроек межмодульной маршрутизации:

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- Таблица маршрутизации верхнего уровня -->
<ROUTES>
  <!-- Адреса сервис-провайдеров (сообщения, посланные «с» или «на» такие
        адреса, будут обрабатываться модулем SP) -->
  <ROUTE side="SP" default="false" type="address">
    <TO masktype="REGEXP" type="PLMN" value="^[+]?0009013"/>
    <TO masktype="REGEXP" type="PLMN" value="^[+]?00090250"/>
    <TO masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{6}"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{6}"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{11}"/>
  </ROUTE>

  <!-- Адреса отправки MMS на e-mail -->
  <ROUTE side="EMAIL" default="false" type="address">
    <TO masktype="REGEXP" type="EMAIL" value="^[a-zA-Z][\w\.-]*[a-zA-Z0-9]@[a-zA-Z0-9][\w\.-]*[a-zA-Z0-9]\.[a-zA-Z][a-zA-Z\.-]*[a-zA-Z$]"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{11}"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{7}"/>
  </ROUTE>

  <!-- Все сообщения от "SP" передавать на доставку на "MOBILE" -->
  <ROUTE side="MOBILE" default="false" type="address" source="SP" />

  <!-- Мобильные адреса локального региона -->
  <ROUTE side="MOBILE" default="false" type="address">
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{11}"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{7}"/>
    <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{6}"/>
    <FROM masktype="REGEXP" type="EMAIL" value=".*"/>
    <TO masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{11}"/>
  </ROUTE>

  <!-- Все остальные адреса передавать для других MMSC -->
```

```

<ROUTE side="interMMSC" default="false" type="address">
  <TO masktype="REGEXP" type="PLMN" value="^[+]?[1-9]{1}[0-9]{6}" />
  <TO masktype="REGEXP" type="PLMN" value="^[+]?[1-9]{1}[0-9]{10}" />
  <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{11}" />
  <FROM masktype="REGEXP" type="PLMN" value="^[+]?[0-9]{7}" />

  <FROM masktype="REGEXP" type="EMAIL" value="^[a-zA-Z][\w\.-]*[a-zA-Z0-9]@[a-zA-Z0-9][\w\.-]*[a-zA-Z0-9]\.[a-zA-Z][a-zA-Z\.-]*[a-zA-Z]$" />
</ROUTE>

<!-- Некорректный адрес. Перенаправить на ошибку -->
<ROUTE side="ERROR" default="true" type="address" />
</ROUTES>

```

2.7. Конфигурация тарификации сообщений

Конфигурация тарификации обработки сообщений модулями центра задается элементом **<TARIFICATIONS>**, который может быть реализован в виде ссылки на внешний файл с помощью атрибута **cfg**, или в случае отсутствия ссылки, должен иметь тело, имеющее следующую структуру:

<TARIFICATIONS> – корневой элемент конфигурации системы тарификации

<METHODS> – набор методов тарификации – типов систем тарификации

<METHOD> – описание условий выбора метода тарификации в зависимости от нижестоящих параметров сообщения

<DIRECTION> – направление сообщения

<TO> – адрес получателя сообщения

<FROM> – адрес отправителя сообщения

<DISCOUNTS> – список коэффициентов тарификации сообщения

<DISCOUNT> – описание условий применения коэффициента тарификации в зависимости от параметров сообщения

<PAYMENTS> – список интерфейсов взаимодействия с системами тарификации

<PAYMENT> – конфигурация интерфейса

<CDR> – конфигурация CDR интерфейса

<FILE> – описание параметров файла

<FIELD> – описание поля CDR файла

<INTERFACE> – описание реализации интерфейса взаимодействия с системой тарификации

В секции **<METHODS>** задаются варианты используемых методов тарификации.

Каждый метод тарификации описывается вложенным элементом **<METHOD>**. Элемент описывает параметры, определяющие выбор типа интерфейса, используемого для выполнения тарификации. Имеет следующие атрибуты:

- **name** – строка, имя метода;
- **default** – логическое значение (**true** или **false**), указывающее, используется ли этот метод тарификации по умолчанию.
- **interface** – строка, указывающая на имя интерфейса тарификации, используемого данным методом. Должно совпадать с именем (**id**) одного из интерфейсов, описанных при помощи элементов **<INTERFACE>** (см. выше).

Каждое описание метода тарификации содержит в своем теле следующие дочерние элементы:

- **<DIRECTION>** – указывает направление сообщения. Имеет единственный атрибут **type**, соответствующий одному из направлений (МОМТ, МОАТ, АОМТ, АОАТ, см. [раздел 2.5](#));
- **<FROM>** – адресный пул отправителей;
- **<TO>** – адресный пул получателей.

Каждый элемент может присутствовать в количестве одного или более экземпляров. Формат элементов **<FROM>** и **<TO>** описывается в [разделе 6.5](#).

Направление сообщения определяется при вызове интерфейса тарификации модулем провайдера при обработке сообщения. Направление определено для каждого модуля провайдера в конфигурации ссылки на интерфейс.

Элемент **<DISCOUNTS>** задает перечень используемых коэффициентов тарификации. Этот элемент не имеет атрибутов.

Каждый коэффициент описывается дочерним элементом **<DISCOUNT>**. Коэффициент тарификации определяет процентное значение изменения базовой тарификации в зависимости от содержания сообщения. Коэффициент определяется минимальным значением из всех подходящих по параметрам сообщения, при условии, что выполняются условия фильтрации по адресному пулу.

Элемент **<DISCOUNT>** имеет обязательный атрибут **id**, задающий уникальное имя коэффициенту дисконта, который используется провайдером для ссылки на

коэффициент при создании индивидуального набора коэффициентов провайдера. Элемент содержит следующие вложенные элементы:

- **<CONTENT>** – элемент, задающий критерии вычисления коэффициента тарификации сообщения в зависимости от его содержимого.

Элемент имеет следующие атрибуты:

- **type** – строка, тип содержания сообщения.

Возможные типы сообщения:

- **text** – сообщение содержит только текстовые части;

- **video** – сообщение содержит только текстовые и видео части (Content-Type=video/*);

- **audio** – сообщение содержит только текстовые и аудио части (Content-Type=audio/*);

- **image** – сообщение содержит только текстовые и картинки части (Content-Type=image/*);

- любое другое значение – смешанное содержание.

- **size_limit** – число, ограничение суммарного объема частей по размеру, в килобайтах

- **value** – число, значения коэффициента применяемого к базовой тарификации, в процентах.

- **<FROM>** – адресный пул отправителей;

- **<TO>** – адресный пул получателей.

Формат элементов **<FROM>** и **<TO>** описывается в [разделе 6.5](#). Элементы задают пулы адресов отправителей и получателей, сообщения которых будут тарифицироваться с учетом этого коэффициента.

Элемент **<PAYMENTS>** содержит набор именованных групп интерфейсов взаимодействия с платежными системами. Элемент не имеет атрибутов. Для осуществления тарификации должен иметь внутри не менее одного элемента описания

группы интерфейсов взаимодействия с платежной системой **<PAYMENT>**, а также перечень используемых реализаций **<interface>**.

Именованная группа интерфейсов тарификации должна включать себя реализацию конфигурации или ссылку на ранее описанную реализацию, для каждого типа интерфейсов, которые указываются в наборе методов (**<METHODS>**, см. выше).

Атрибуты корневого элемента именованной группы **<PAYMENT>**:

- **id** – строка, уникальный идентификатор группы;
- **default** – логическое значение, определяет, будет ли интерфейс использоваться по умолчанию.

В текущей версии сервиса используются следующие варианты интерфейсов с системами тарификации:

- **CDR** – интерфейс формирования текстовых CDR-файлов;
- **PPS** – интерфейс взаимодействия с сервером *Converse PPS*.

Каждый вариант интерфейса взаимодействия с системами тарификации задается соответствующим элементом XML.

Элемент **<INTERFACE>** определяет тип взаимодействия с системой тарификации и задает класс его реализации. Он описывается следующим набором атрибутов:

- **id** – строка, имя (уникальный идентификатор) интерфейса;
- **type** – строка, уникальное имя типа взаимодействия, используется для связи с форматами вывода платежной информации **<PAYMENT>** (см. ниже);
- **implementation** – имя класса, реализующего данной взаимодействие. В данной версии реализованы следующие типы интерфейсов:

- `com.mobitech.mmsc.payment.CDRPayment` – реализует текстовый CDR-файл;

- `com.mobitech.mmsc.payment.NokiaCDRPayment` – реализует бинарный CDR-формат в соответствии со спецификацией *Nokia MMS Center 2.1/3.0 CDR*;

- `com.mobitech.mmsc.payment.PPSComverse` – реализует взаимодействие с *PPS Comverse*;

• com.mobitech.mmsc.payment.Nokia

aIACC – интерфейс к тарификационной системе **Nokia IACC**.

Необходимо перечислить столько элементов **<INTERFACE>**, сколько интерфейсов используется при тарификации.

2.7.1. Конфигурация интерфейса формирования текстовых CDR

Текстовые файлы CDR располагаются в каталоге, определенном атрибутом **path**. Имя файла определяется как **<prefix><дата>_<номер>**, где:

- Поле **<prefix>** определяется атрибутом **prefix**;
- Поле **<дата>** формируется в формате **YYYYMMDD**;
- Поле **<номер>** – число в формате **XXXX**, циклический счётчик в диапазоне значений **0000 – 9999**.

В конфигурации **m.MMSC** для файла CDR определяются два ограничения:

- **release_time** – максимальное время использования файла;
- **recordsLimit** – максимальное количество строк в файле.

При достижении любого из ограничений текущий файл с записями закрывается, и создается новый файл, в который продолжается запись событий. Копия закрытого файла перемещается в каталог, доступный для ИБС.

Параметры формирования текстового CDR-файла описываются вложенным элементом, имя которого соответствует типу интерфейса, имплементируемого классом **com.mobitech.mmsc.payment.CDRPayment** (см. выше описание элемента **<INTERFACE>**, атрибут **type**).

Например, если имплементация интерфейса описана элементом

```
<INTERFACE id="CDR_id" type="CDR"
    implementation="com.mobitech.mmsc.payment.CDRPayment"/>
```

то конфигурация может иметь следующий вид:

```
<CDR ref="">
  <FILE path="./cdrs_SPMOMT/" out_path="./cdrs_SPMOMT/OUT/"
    prefix="mmsi1" recordsLimit="100" delim="," release_time="10">
    <FIELD name="TIMESTAMP"/>
    <FIELD name="ORIG_MSISDN"/>
    <FIELD name="TERM_MSISDN"/>
    <FIELD name="DISCOUNT"/>
    <FIELD name="VALUE"/>
  </FILE>
</CDR>
```

Атрибуты:

- **ref** – строка, ссылка на другую группу **<PAYMENT>**, содержащую группу CDR того же типа, из которой будет взята реализация данного интерфейса.

Правила формирования и содержание файла описываются вложенным элементом **<FILE>**, имеющим атрибуты:

- **path** – строка, путь к каталогу, в котором будут формироваться файлы. Если каталога не существует, он будет создан;
- **out_path** – строка, указывающая путь, по которому будет перемещена копия CDR файла после закрытия. Если пути не существует, то он будет создан;
- **prefix** – строка, префикс имени файла CDR (содержание строки не должно включать символов, недопустимых в именах файлов);
- **release_time** – число, задающее максимальное время использования одного файла CDR. По умолчанию не ограничено;
- **nameType** – строка, определяющая тип формирования последовательности имен файлов. В текущей версии **m.MMSC** не используется;
- **recordsLimit** – число, определяющее максимальное количество записей в одном файле. По умолчанию – 1000;
- **delim** – строка, символ, или последовательность символов, используемых в качестве разделителя полей файла. По умолчанию используется символ **" , "**;
- **direction** – строка, определяющая направление, для которого может использоваться данная реализация. Если направление указано для CDR-файла, а направление сообщения ему не соответствует, запись сформирована не будет.

Содержание и последовательность полей CDR-файла соответствует последовательности вложенных элементов **<FIELD>**. Элемент **<FIELD>** имеет следующие атрибуты:

- **name** – строка, имя параметра тарификации, записываемого в файл;

- **value** – строка, значение параметра, которое необходимо вывести в файл.

Атрибут **value** используется только для параметров, помеченных символом *****.

Определены имена параметров тарификации, перечисленные в таблице 6.2.

Таблица 6.2 - Имена параметров тарификации

Параметр	Тип	Описание
DISCOUNT	Число	Величина коэффициента тарификации
VALUE	Число	Величина тарификации с учетом коэффициента
MSG_HEADER *	Зависит от значения value	Вывод значения заголовка MMS-сообщения, имя которого указано в атрибуте value . Имя заголовка не чувствительно к регистру и соответствует спецификации <i>OMA-MMS-ENC-V1_2-20040323-C</i>
TIMESTAMP	Дата/время	Время формирования записи для несения в CDR-файл
ORIG_MSISDN	Строка	Стандартизованная форма адреса отправителя
ORIG_TYPE	Цифра	Тип адреса отправителя: 0 – Unknown; 1 – MSISDN; 2 – IPv4; 3 – IPv6; 4 – E-mail
ORIG_ID	Строка	Идентификатор отправителя в соответствии с конфигурацией модуля обработки сообщения
TERM_MSISDN	Строка	Стандартизованный номер получателя сообщения
ORIG_TERM_MSISDN	Строка	Оригинальное (не форматированное) значение адреса получателя сообщения
TERM_TYPE	Цифра	Тип адреса получателя (значения аналогичны ORIG_TYPE)
TERM_ID	Строка	Идентификатор получателя в соответствии с конфигурацией модуля обработки сообщения
EVENT_TYPE	Число в формате XX	Тип события, по которому была сформирована запись: 00 (RECIEVED) – сообщение получено от абонента. Для каждой копии формируется отдельная запись. Т.е.

Параметр	Тип	Описание
		при получении одного сообщения с 3-мя адресатами, в CDR-файл будут внесены три строки, отличающиеся только адресами получателей; 01 (SUBMITTED) – сообщение принято на обработку; 02 (ERROR) – отказ в обработке, сообщение не может быть доставлено или произошла ошибка доставки; 03 (SENT) – передано по назначению (на терминал получателя отправлена нотификация о сообщении); 04 (UPLOADED) – сообщение загружено на терминал абонента; 05 (REJECTED) – сообщение отвергнуто абонентом или фильтрами m.MMSC; 06 (CONFIRMED) – терминал абонента подтвердил получение; 07 (DELIVERED) – терминал получателя прислал подтверждение корректности MMS-сообщения, а отправителю (при установленном запросе) отправлен отчет о доставке сообщения; 08 (READ) – сообщение прочитано получателем; 09 (LEGACY) – сообщение передано на Legacy-терминал для альтернативной доставки; 10 (EXPIRED) – срок доставки сообщения истек; 11 (FORWARDED) – сообщение переадресовано без загрузки на MT; 12 (MMSTORED) – сообщение сохранено на сервере; 13 (MMDELETED) – сообщение удалено с сервера; 14 (MMUPLOADED) – сообщение загружено для редактирования; 15 (CHECK_AMOUNT) – проверка аккаунта на допустимость выполнения действия (для совместимости);

Параметр	Тип	Описание
		16 (APPLY_PAYMENT) – служебное событие; 17 (REVERSE_PAYMENT) – сообщение отклонено получателем, запрос возврата тарификации; 18 (UNKNOWN) – неизвестно системное событие
MSG_ID	Строка	Уникальный внутренний идентификатор MMS-сообщения
PARENT_MSG_ID	Строка	Идентификатор родительского сообщения
TOTAL_RCPTS	Число	Количество получателей сообщения
MSG_SIZE	Число	Размер сообщения в байтах
MSG_TYPE	Строка	Тип содержания сообщения: ТЕХТ, IMAGE, AUDIO, VIDEO, MIXED
SENT_DATE	Строка	Время отправки сообщения получателю в формате <code>yyyyMMddHHmmss</code>
INCOMING_DATE	Строка	Время поступления сообщения на сервер в формате <code>yyyyMMddHHmmss</code>
CNT_SIZE_VIDEO	Число	Суммарный размер видео частей в MMS-сообщении
CNT_NUMBER_VIDEO	Число	Количество видео частей в MMS -сообщении
CNT_SIZE_AUDIO	Число	Суммарный размер аудио частей в MMS-сообщении
CNT_NUMBER_AUDIO	Число	Количество аудио частей в MMS-сообщении
CNT_SIZE_IMAGE	Число	Суммарный размер частей с изображениями в MMS
CNT_NUMBER_IMAGE	Число	Количество частей с изображениями в MMS
CNT_SIZE_TEXT	Число	Суммарный размер текстовых частей в MMS
CNT_NUMBER_TEXT	Число	Количество текстовых частей в MMS-сообщении
CNT_SIZE_OTHER	Число	Суммарный размер прочих частей в MMS-сообщении
CNT_NUMBER_OTHER	Число	Количество прочих частей в MMS-сообщении
DELIVERY_REPORT	Цифра	Наличие запроса отчета о доставке: 0 – отчет не запрошен, 1 – отчет запрошен
READ_REPLY	Цифра	Наличие запроса отчета о прочтении: 0 – отчет не запрошен, 1 – отчет запрошен
RECIPTS_NO	Число	Количество получателей сообщения
REC_ID	Число в	Номер записи в текущем файле

Параметр	Тип	Описание
	формате XXXXXX	
MSG_CLASS	Цифра	Класс сообщения: 0 – Personal 1 – Advertisement, 2 – Informational 3 – Auto.
FIXED_VALUE *	Строка	Вывод в файл значения, заданного в атрибуте value
KKZ_EVENT_TYPE	Число	Формируется только для событий DELIVERED и RECIEVED и имеет значения 52 и 51 соответственно
KKZ_CNT_TYPE	Число	Обозначает тип контента MMS-сообщения. Может принимать значения, соответствующие различным значениям поля MSG_TYPE: Для типа "TEXT" – 0 "IMAGE" – 1 "AUDIO" – 2 "VIDEO" – 4 "MIXED" – 8
KKZ_TARIFF_CLASS	Число	Класс тарификации. В зависимости от размера MMS-сообщения имеет следующие значения: от 0 до 5119 байт 1 от 5120 до 10239 байт 2 от 10240 до 46079 байт 3 от 46080 байт 4
KKZ_STATUS	Строка	Вывод статуса, в котором находится MMS-сообщение («wait», «notified» и т.п.)

2.7.2. Конфигурация интерфейса взаимодействия с Comverse PPS

Реализация взаимодействия с **Comverse PPS** поддерживает работу с основным и резервным PPS-сервером. Параметры взаимодействия описываются вложенным элементом **<PPS>**, который имеет следующие атрибуты:

- **ref** – строка, ссылка на другую группу, из которой будет взята реализация данного интерфейса. В случае если ссылка указана, тело элемента не используется;
- **terminatorPrefix** – строка, задающая префикс, устанавливаемый перед адресами получателей;
- **originatorPrefix** – строка, задающая префикс перед адресами отправителей;
- **heartBeatPeriod** – число, равное интервалу отправки запроса *HeartBeat* для поддержки соединения, в секундах;
- **waitResponseTimeout** – число, задающее максимальный интервал ожидания ответа перед повтором запроса, в секундах;
- **debugOutput** – строка, указывающая имя файла вывода журналов отладочной информации;
- **traceOutput** – строка, указывающая имя файла журнала событий;

Дочерние элементы описывают параметры подключения к первичному (<PRIMARY_HOST>) и вторичному (<SECONDARY_HOST>) серверам PPS. Оба элемента имеют следующие атрибуты:

- **host** – строка, задающая имя или IP-адрес сервера;
- **port** – число, задающее порт подключения к серверу PPS.

Пример задания конфигурации подключения к серверу PPS:

```
<PPS terminatorPrefix="" originatorPrefix="" ref="" heartBeatPeriod="30"
waitResponseTimeout="10" debugOutput="./logs/PPS/debug.log"
traceOutput="/logs/PPS/trace.log">

<PRIMARY_HOST host="hostname" port="5467"/>
<SECONDARY_HOST host="hostname" port="5467"/>

</PPS>
```

Пример задания конфигурации тарификации:

```
<TARIFICATIONS>
<METHODS>
<METHOD name="" interface="PPS" default="false">
<DIRECTION type="MOMT"/>
<DIRECTION type="MOAT"/>
<TO type="PLMN" masktype="REGEXP" value="."/>
<FROM type="PLMN" masktype="REGEXP" value="77.*"/>
</METHOD>
<METHOD name="" interface="CDR" default="true">
<DIRECTION type="AOMT"/>
<DIRECTION type="MOMT"/>
<DIRECTION type="MOAT"/>
</METHOD>
```

```

</METHODS>
<DISCOUNTS>
  <DISCOUNT id="contentText">
    <CONTENT type="text" size_limit="100" value="10"/>
    <TO type="PLMN" masktype="VALUE" value="[+]?[0-9]{7,13}$"/>
    <TO type="EMAIL" masktype="VALUE" value="test@mobitech.com"/>
    <FROM type="PLMN" masktype="REGEXP" value="[+]?77[0-9]{9}$"/>
  </DISCOUNT>
</DISCOUNTS>

<PAYMENTS>
  <PAYMENT id="DEFAULT" default="true">
    <CDR>
      <FILE path="/var/CDRS/cdrs_MO/" nameType="DATE" delim=","
        recordsLimit="1000" out_path="./cdrs_MOMT/_OUT"
        release_time="10">
        <FIELD name="TIMESTAMP"/>
        <FIELD name="ORIG_MSISDN"/>
        <FIELD name="ORIG_ID"/>
        <FIELD name="TERM_MSISDN"/>
        <FIELD name="TERM_ID"/>
        <FIELD name="DISCOUNT"/>
        <FIELD name="VALUE"/>
      </FILE>
    </CDR>
    <PPS terminatorPrefix="" originatorPrefix="" ref=""
      heartBeatPeriod="30" waitResponseTimeout="10"
      debugOutput="./logs/PPS/debug.log"
      traceOutput="./logs/PPS/trace.log">
      <PRIMARY_HOST host="hostname" port="5467"/>
      <SECONDARY_HOST host="hostname" port="5467"/>
    </PPS>
  </PAYMENT>
  <PAYMENT id="SP" default="false">
    <CDR ref="">
      <FILE path="./cdrs_SPMOMT/" out_path="./cdrs_SPMOMT/OUT/"
        prefix="mmsi1_" recordsLimit="100" delim=","
        release_time="10" >
        <FIELD name="TIMESTAMP"/>
        <FIELD name="ORIG_MSISDN"/>
        <FIELD name="TERM_MSISDN"/>
        <FIELD name="DISCOUNT"/>
        <FIELD name="VALUE"/>
      </FILE>
    </CDR>
    <PPS ref="DEFAULT"/ >
  </PAYMENT>
  <PAYMENT id="InterMMSC" default="false">
    <CDR ref="DEFAULT"/>
    <PPS ref="DEFAULT"/>
  </PAYMENT>

  <INTERFACE id="CDR" type="CDR"
    implementation="com.mobitech.mmsc.payment.CDRPayment"/>
  <INTERFACE id="PPS" type="PPS"
    implementation="com.mobitech.mmsc.payment.PPSComverse"/>
</PAYMENTS>

</TARIFICATIONS>

```

2.8. Конфигурация внутримодульной маршрутизации сообщений

Конфигурация внутренней модульной маршрутизации сообщений описывается для каждого модуля в отдельном файле, на который ссылается атрибут **routing** элемента **<XX_SIDE>**, задающий конфигурацию этого модуля в файле основной конфигурации (см. [раздел 6.5](#)).

Файл содержит конфигурацию общих параметров и правил обработки сообщений, а также список провайдеров со ссылками на файлы их конфигураций.

Конфигурация внутренней маршрутизации имеет следующую структуру

```
<PROVIDERS>
  <COMMON defaultLogFile="log_file" defaultLogSize="size">
    <RULES>
      <RULE id="rule_id" action="reject|change" class="rule_class_name">
        <PARAM type="string" default="def_value"/>
        ...
      </RULE>
      ...
    </RULES>
  </COMMON>

  <PROVIDER id="provider_id" cfgfile="cfg_file" loglevel="level"/>
  ...
</PROVIDERS>
```

Корневой элемент **<PROVIDERS>** атрибутов не имеет.

2.8.1. Общая конфигурация маршрутизации

Общая конфигурация маршрутизации задается элементом **<COMMON>**, и содержит описание параметров маршрутизации по умолчанию. Элемент имеет следующие атрибуты:

- **defaultLogFile** – строка, задающая имя файла журнала для всех провайдеров. Используется для журналирования информации по сообщениям, для которых не определена дальнейшая маршрутизация (в текущей версии **m.MMSC** не поддерживается, оставлено для обратной совместимости);
- **defaultLogSize** – число, задающее ограничение размера файла журнала в килобайтах, используется при ротации журналов (в текущей версии **m.MMSC** не используется).

Внутри тела элемента **<COMMON>** располагается список правил маршрутизации, задаваемый элементом **<RULES>**, не имеющим атрибутов. Если правила

маршрутизации у модуля отсутствуют, следует использовать элемент списка правил с пустым телом **<RULES/>**.

Каждое правило внутри списка задается элементом **<RULE>**, задающим описание шаблона или реализации правила. В секции **<COMMON>** могут описываться только шаблоны правил. Каждый шаблон должен иметь уникальный идентификатор, определять количество параметров и их значения по умолчанию, а так же определять действие по умолчанию, и реализующий правило класс *Java*.

Эти параметры правила задаются следующими атрибутами элемента **<RULE>**:

- **id** – строка, задающая уникальный, в рамках модуля обработки, идентификатор шаблона правила. Используется в ссылках (**ref**) в конфигурации провайдеров (см. ниже, [раздел 6.8.2](#));

- **action** – строка, определяющая тип правила. Используются следующие типы правил:

- **reject** – сообщение, не удовлетворяющее правилу, должно быть заблокировано;

- **change** – сообщение, не удовлетворяющее правилу, должно быть модифицировано в соответствии с параметрами реализации правила. Не для всех правил возможна реализация данного метода;

- **class** – имя Java-класса, реализующего правило.

Набор параметров, которые использует правило, а также значение этих параметров по умолчанию задается набором вложенных элементов **<PARAM>**. Элемент имеет два атрибута:

- **type** – строка, задающая тип параметра. Допустимы следующие типы: целое число (**int**), строка (**string**), логическое значение (**bool**);

- **default** – значение параметра правила по умолчанию.

Все Java-классы реализации правил находятся в пространстве имен `com.mobitech.mmsc.provider.rules`, поэтому все значения атрибутов **class** имеют одинаковый вид `com.mobitech.mmsc.provider.rules.<имя класса реализации>`.

Набор правил, реализованных в текущей версии **m.MMSC**, описан в таблице

6.3.

Таблица 6.3 - Конфигурация правил обработки и контроля параметров сообщений

Имя класса реализации	Описание	Реализация действий – action		Параметры – PARAM		
		change	reject	№ п/п	Тип	Значение
TrafficShaper	Ограничение интенсивности запросов отправки со стороны провайдера	Задержка внеочередного запроса	Отклонение	1	строка	Единицы времени Варианты: sec, min, hour, day
				2	число	Ограничение количества сообщений в единицу времени
contentIdsChecker	Проверка уникальности идентификаторов вложений MMS-сообщения	Модификация неуникальных идентификаторов	Отклонение	нет		
recipientsCount	Контроль количества адресатов	Доставка только к первым адресатам в соответствии с ограничением	Отклонение	1	число	Максимальное количество адресатов
reportsRequest	Проверка или изменения запроса отчета при отправке сообщения	Установка флага запроса в заданное значение	Отклонение	1	логическое значение	Тип отчета. Возможные варианты: RequestDeliveryReport – отчет о доставке; RequestReadReport – отчет о прочтении.
sizeLimit	Ограничение размера сообщения	В сообщении остаются только первые части, которые вписываются в установленные ограничения. Презентационная часть остается всегда	Отклонение	1	число	Размер сообщения, в байтах

tariffClasses	Проверка принадлежности значения заголовка MMS-сообщения X-Nokia-MMSC-Charging числовому набору	Невозможна	Отклонение	1	строка	Тип ограничения. Варианты: between – диапазон тарифов one_of – список разрешенных тарифов no_one_of – список запрещенных тарифов
				2	число	Номер тарифа или нижняя граница диапазона
				3	число	Следующий номер в списке, или верхняя граница диапазона

```
<PROVIDERS>
  <COMMON defaultLogFile="./logs/commonDelivery.log"
    defaultLogSize="1000000">
    <!-- Описание правил обработки сообщений и ограничения доступа -->
    <!-- Параметры фиксированы, а значения параметров могут меняться -->
    <RULES>
      <!-- Запрашивать отчет о прочтении сообщения -->
      <RULE id="reports" action="change"
        class="com.mobitech.mmsc.provider.rules.reportsRequest">
        <PARAM type="string" default="RequestReadReport"/>
        <PARAM type="string" default="Yes"/>
      </RULE>

      <!-- Проверять, чтобы было не более 2 получателей. Отклонить
        сообщение, если больше -->
      <RULE id="recCount" action="reject"
        class="com.mobitech.mmsc.provider.rules.recipientsCount">
        <PARAM type="int" default="2"/>
      </RULE>

      <!-- Отклонять сообщения размером более 1024 байт -->
      <RULE id="sizeLimit" action="reject"
        class="com.mobitech.mmsc.provider.rules.sizeLimit">
        <PARAM type="int" default="1024"/>
      </RULE>
    </RULES>
  </COMMON>

  <!-- Определение списка провайдеров -->
  <PROVIDER id="MOBILE1" cfgfile="./etc/mobile/mobile.cfg.xml"
    loglevel="10"/>
</PROVIDERS>
```

2.8.2. Конфигурация провайдеров

Список провайдеров модуля определяется набором элементов **<PROVIDER>** внутри секции **<PROVIDERS>**. Элемент **<PROVIDER>** может описывать ссылку на файл конфигурации провайдера или непосредственно реализовывать его конфигурацию.

Элемент **<PROVIDER>**, описывающий ссылку на файл, имеет следующие атрибуты:

- **id** – строка, задающая идентификатор провайдера (VASPID), уникальный в рамках модуля. Используется при ведении журналов;
- **cfgfile** – строка, указывающая на имя файла, содержащего конфигурацию провайдера;
- **loglevel** – число в диапазоне 0 – 10, определяющее уровень детализации журналов (в текущей версии **m.MMSC** не используется, оставлено для обратной совместимости).

Элемент **<PROVIDER>**, описывающий конфигурацию, а не ссылку, имеет следующие атрибуты:

- **id** – строка, уникальный, в рамках модуля, идентификатор провайдера (VASPID). Используется при ведении журналов;

- **name** – строка, произвольное имя провайдера.

Используется при ведении журналов;

- **active** – логическое значение, индикатор активности провайдера. При маршрутизации и анализе поступивших запросов используются только активные провайдеры. По умолчанию – истина;

- **loglevel** – число, в диапазоне 0 – 10, определяющее уровень детализации логов (в текущей версии не используется, оставлено для обратной совместимости);

- **logfile** – строка, имя файла журнала детальной информации по сообщениям (не используется);

- **logsize** – число, максимальный размер файла журнала, в байтах. Используется при ротации (не используется в текущей версии).

Описание конфигурации провайдера содержит следующие дочерние элементы:

- **<ACCESS>** – элемент конфигурации списков доступа. Содержит перечень списков доступа, каждый из которых задается элементом **<ACCESS_LIST>**. Дочерние элементы **<ACCESS_SLIST>** могут описывать ссылку на файл конфигурации списка доступа или полностью включать в себя список. Несколько различных элементов **<ACCESS>** могут содержать элементы **<ACCESS_LIST>**, ссылающиеся на один и тот же файл.

При ссылке на файл элемент **<ACCESS_LIST>** имеет единственный атрибут **file**, указывающий имя файла списка доступа.

Элемент **<ACCESS_LIST>**, описывающий конфигурацию доступа, содержит черный и белый список адресов (как соответствующие дочерние элементы) и атрибуты, определяющие, какой из списков используется. Атрибуты элемента **<ACCESS_LIST>**:

- **usewhite** – логическое значение, указывает, использовать ли белый список;
- **useblack** – логическое значение, указывает, использовать ли черный список.

Оба списка (черный и белый) одновременно, в рамках одного элемента **<ACCESS_LIST>**, использоваться не могут. Белый и черный списки задаются соответствующими элементами, не имеющими атрибутов:

- **<WHITE>** – элемент, содержащий адреса белого списка;
- **<BLACK>** – элемент, содержащий адреса черного списка.

Каждый список состоит из перечня дочерних элементов **<TO>** и **<FROM>**, определяющих адреса получателей и отправителей соответственно. Формат задания адресов см. в [разделе 6.5](#).

Пример задания списка доступа:

```
<ACCESS_LIST usewhite="0" useblack="1">
  <WHITE>
    <TO masktype="VALUE" type="PLMN" value="+78129345813"/>
    <FROM masktype="REGEXP" type="PLMN" value="[+]?[0-9]*/>
  </WHITE>
  <BLACK>
    <TO masktype="REGEXP" type="PLMN" value="[+]?7911[0-9]*/>
  </BLACK>
</ACCESS_LIST>
```

▪ **<CLIENT>** – элемент конфигурации клиентского интерфейса отправки сообщений. Клиентский интерфейс используется в модулях **SP** и **EMS** (см. [раздел 2.2](#)) при отправке сообщений от **m.MMSC**. Конфигурация клиентского интерфейса модуля определяется значениями атрибутов элемента **<CLIENT>**:

- **ref** – строка, определяет ссылку на элемент **<SERVER>**, описывающий парный серверный интерфейс взаимодействия. Значение атрибута **ref** должно совпадать со значением атрибута **id** элемента **<SERVER>** (см. [раздел 6.4](#));
- **url** – строка, описывающая адрес подключения в стандарте URL;
- **maxconnections** – число, определяющее ограничение числа соединений, используемых для подключения к серверу при отправке;
- **contimeout** – число, определяющее максимальное время простоя соединения до отключения. В течение данного периода времени соединение может использоваться повторно.

Пример задания конфигурации клиента:

```
<CLIENT ref="first" url="http://localhost:50703/mobitech/eaif"
  maxconnections="10" contimeout="60"/>
```

▪ **<RULE>** – конфигурация правил обработки. Каждый элемент **<RULE>** в конфигурации провайдера описывает реализацию шаблона правила, заданного в модуле (см. [раздел 6.8.1](#)), для провайдера. Возможны два варианта реализации:

- *по умолчанию*, с указанием только ссылки на правило, определенное в конфигурации модуля;
- *с переопределением параметров* родительского шаблона правила.

Элемент **<RULE>** имеет единственный атрибут – **ref**, задающий ссылку на идентификатор родительского шаблона правила (значение атрибута **id**). В качестве параметров правила, будут использованы параметры родительского модуля.

В случае, когда применяется реализация правила по умолчанию (без переопределения параметров), элемент **<RULE>** не имеет тела.

Во втором случае для переопределения значений параметров используются вложенные элементы **<PARAM>**, имеющие синтаксис, совпадающий с тем, который используется при определении шаблонов.

При перегрузке параметров шаблонного правила возможна только поочередная перегрузка. Число параметров при переопределении может быть меньшим, чем при описании шаблона по умолчанию. В этом случае для недостающих параметров будут использованы значения по умолчанию из шаблона.

Конфигурация провайдера может включать в себя несколько различных реализаций одного и того же шаблона правила. Например, это может использоваться при ограничении интенсивности запросов, как за секунду, так и за час или сутки.

Пример переопределения правила для провайдера:

```
<!-- Проверять, чтобы было не более 4 получателей -->  
<RULE ref="recCount" action="reject">  
  <PARAM type="int" value="4"/>  
</RULE>
```

- **<IPSOURCE>** – элемент конфигурации списка доступа на уровне IP.

Используется для идентификации и ограничения прав доступа провайдера на уровне IP-адресов. Элемент определяет IP-адреса провайдера, которые могут использоваться для подключения к программному обеспечению платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**». Все адреса, которые не удовлетворяют списку, будут отброшены. Атрибутов этот элемент не имеет.

Каждый элемент списка адресов может быть определен в виде маски или конкретного значения, и задаётся при помощи вложенного элемента **<IADDRESS>**, который имеет следующие атрибуты:

- **masktype** – строка, определяющая тип задания адреса.

Возможны варианты **"VALUE"** или **"REGEXP"**, при определении адреса в виде конкретного значения или регулярного выражения, соответственно;

- **value** – строка, значение адреса или регулярной маски набора адресов (в зависимости от значения атрибута **masktype**).

Пример переопределения правила для провайдера:

```
<IPSOURCE>  
  <IADDRESS masktype="VALUE" value="192.168.0.1"/>  
  <IADDRESS masktype="REGEXP" value="127.0.0.[0-9]"/>  
</IPSOURCE>
```

- **<ADDRESSES>** – элемент конфигурации, задающий пулы адресов провайдера. Адресный пул провайдера используется для определения принадлежности сообщения провайдеру при его отправке и/или получении.

Вторичное предназначение пула адресов – ограничения диапазона адресов, на которые может отправлять провайдер и адресов, от имени которых может осуществляться отправка.

Пул адресов определяется перечнем вложенных элементов **<TO>** – определяющих адреса, которые принадлежат провайдеру для входящих сообщений и **<FROM>** – определяющих набор адресов, которые могут быть использованы провайдером в качестве исходящих.

Элементы **<TO>** и **<FROM>** имеют одинаковые наборы атрибутов. Формат элементов см. в разделе 5.4.

Пример задания пулов адресов провайдера:

```
<ADDRESSES>
  <TO masktype="REGEXP" type="PLMN" value="[+]?[0-9]*/>
  <!-- MMS, отправленные на короткий номер 00019, придут сюда -->
  <TO masktype="VALUE" type="PLMN" value="00019"/>
  <FROM masktype="REGEXP" type="PLMN" value="[+]?[0-9]*/>
  <FROM masktype="VALUE" type="EMAIL" value="nh@nokia.com"/>
  <FROM masktype="VALUE" type="IPv4" value="127.0.0.1"/>
  <FROM masktype="VALUE" value="127.0.0.1"/>
  <!-- MMS, отправленные от сервиса «VAS_METEO», придут сюда -->
  <FROM masktype="VALUE" value="VAS_METEO"/>
</ADDRESSES>
```

▪ **<PAYMENT>** – элемент, задающий параметры тарификации для провайдера. Для каждого провайдера может быть определен интерфейс тарификации и специфические параметры, а также перечень событий, возникающих при обработке сообщения, которые инициируют тарификацию.

Привязка провайдера к системе тарификации описывается элементом **<PAYMENT>** с использованием следующих атрибутов:

- **ref** – строка, ссылка на идентификатор (атрибут **id**) одноименного элемента конфигурации системы тарификации в основных настройках **m.MMSC** (см. [раздел 6.7](#));
- **base** – число с плавающей точкой, определяющее базовое значение стоимости отправки сообщения данному провайдеру;
- **discount** – строка, ссылка на идентификатор коррекции стоимости **<DISCOUNT>**. Если значение не задано (пустая строка), никакой коэффициент не используется;
- **origID** – строка, произвольное значение, используется в качестве идентификатора провайдера-отправителя сообщения при взаимодействиях с тарификационными системами;
- **termID** – строка, произвольное значение, используется в качестве идентификатора провайдера-получателя при взаимодействии с системами тарификации;

- **direction** – строка, определяет направление сообщения обрабатываемого данным модулем. Учитывается при выборе интерфейса тарификации. Возможные варианты значения: МОМТ, МОАТ, АОМТ, АОАТ.

Список событий, вызывающий тарификацию, указывается в виде перечня вложенного элемента **<EVENTS>**, содержащего набор элементов **<EVENT>**, каждый из которых задает единичное событие. Элемент **<EVENTS>** не имеет атрибутов, а элемент **<EVENT>**, имеет один обязательный строковый атрибут **type**, задающий тип события. Допустимые типы событий приведены в таблице 6.4.

Таблица 6.4 - Типы событий тарификации

Название	Описание
SUBMITTED	Сообщение принято на обработку
RECIEVED	Сообщение получено от абонента. Для каждой копии формируется отдельная запись. Т.е. при получении одного сообщения с 3-мя адресатами, в CDR-файл будут внесены три строки, отличающиеся только адресами получателей
UPLOADED	Сообщение загружено на терминал абонента
REJECTED	Сообщение отвергнуто абонентом или фильтрами m.MMSC
CONFIRMED	Терминал абонента подтвердил получение
READ	Сообщение прочитано получателем
LEGACY	Сообщение передано на Legacy-терминал для альтернативной доставки
EXPIRED	Срок доставки сообщения истек
FORWARDED	Сообщение переадресовано без загрузки на МТ получателя
MMSTORED	Сообщение сохранено на сервере (MMBox)
MMDELETED	Сообщение удалено с сервера (MMBox)
MMUPLOADED	Сообщение загружено для редактирования (MMBox)
ERROR	Отказ в обработке, сообщение не может быть доставлено или произошла ошибка доставки
SENT	Передано по назначению (на терминал получателя отправлена нотификация о сообщении)
DELIVERED	Терминал получателя прислал подтверждение корректности MMS-сообщения, а отправителю (при установленном запросе) отправлен отчет о доставке сообщения
CHECK_AMOUNT	Проверка аккаунта на допустимость выполнения действия (для совместимости)
APPLY_PAYMENT	Служебное событие
REVERSE_PAYMENT	Сообщение отклонено получателем, запрос возврата тарификации
UNKNOWN	Неизвестно системное событие

Пример задания ссылки на интерфейс тарификации

```
<PAYMENT ref="DEFAULT" base="20.0" discount="DISC" origID="MO"
  termID="MT">
  <EVENTS>
    <EVENT type="SUBMITTED"/>
    <EVENT type="DELIVERED"/>
  </EVENTS>
</PAYMENT>
```

2.8.3. Пример задания конфигурации провайдера

Пример конфигурации провайдера, содержащей ссылки на списки доступа, конфигурацию клиента, набор правил, адресов и ссылку на интерфейс тарификации:

```
<?xml version="1.0" encoding="UTF-8"?>
<PROVIDER id="mobitech" name="mobitech" active="true"
logfile="./logs/mobitech.activity.log" logsize="1000" loglevel="100">
  <ACCESS>
    <ACCESS_LIST file="./etc/accesslist.xml"/>
    <ACCESS_LIST file="./etc/accesslist2.xml"/>
  </ACCESS>
  <CLIENT ref="first" url="http://localhost:50703/mobitech/eaif"
    maxconnections="10" contimeout="60"/>
  <RULE ref="recCount" action="reject">
    <PARAM type="int" value="2"/>
  </RULE>
  <RULE ref="tariff" action="reject">
    <PARAM type="string" value="between"/>
    <PARAM type="int" value="1"/>
    <PARAM type="int" value="10"/>
  </RULE>
  <RULE ref="reports" action="change"/>
  <IPSOURCE>
    <IADDRESS masktype="VALUE" value="192.168.0.1"/>
    <IADDRESS masktype="REGEXP" value="127.0.0.[0-9]"/>
  </IPSOURCE>
  <ADDRESSES>
    <TO masktype="REGEXP" type="PLMN" value="[+]?[0-9]{,6}$"/>
    <FROM masktype="REGEXP" type="PLMN" value="[+]?[0-9]{,6}$"/>
    <FROM masktype="VALUE" type="EMAIL" value="nh@nokia.com"/>
    <FROM masktype="VALUE" type="IPv4" value="127.0.0.1"/>
  </ADDRESSES>
  <PAYMENT ref="DEFAULT" base="20.0" discount="" origID="MO"
    termID="MT">
    <EVENTS>
      <EVENT type="SUBMITTED"/>
      <EVENT type="DELIVERED"/>
    </EVENTS>
  </PAYMENT>
</PROVIDER>
```

1. ЖУРНАЛЫ И СТАТИСТИКА

1.1. Журналы

Все события, возникающие в процессе работы программного обеспечения платформы «Сервисный центр обработки мультимедийных сообщений **m.MMSC**», фиксируются в едином журнале, который по умолчанию называется **mmsc.stdout**, и расположен в каталоге `/var/log/mmsc/`.

В случае если требуется изменить имя файла журнала, или путь к нему, следует отредактировать служебный скрипт запуска сервера.

В этот файл перенаправляется весь вывод со стандартных консолей вывода и ошибок, вследствие чего гарантируется запись в журнал, в том числе, системных сообщений и информации о трассировке стека вызовов при возникновении исключений.

Настройка уровня детализации журнала и формата вывода диагностических сообщений осуществляется редактированием файла **log4j.properties**, расположенного в подкаталоге `etc/`. Примерный вид файла:

```
log4j.rootLogger=error, ScreenLog

log4j.appender.ScreenLog=org.apache.log4j.ConsoleAppender
log4j.appender.ScreenLog.layout=org.apache.log4j.PatternLayout
log4j.appender.ScreenLog.layout.ConversionPattern=%d %-5p [%t] %m%n

log4j.appender.FileLog=org.apache.log4j.RollingFileAppender
log4j.appender.FileLog.file=/var/log/mmsc/mmsc.log
log4j.appender.FileLog.MaxFileSize=10000KB
log4j.appender.FileLog.layout=org.apache.log4j.PatternLayout
log4j.appender.FileLog.layout.ConversionPattern=%d %-5p [%t] %m%n

log4j.logger.com.mobitech=DEBUG
log4j.logger.com.mobitech.mmsc.db.MySQLConnector=ERROR
log4j.logger.com.openwave=ERROR
log4j.logger.httpclient.wire=ERROR
log4j.logger.org.apache.commons=ERROR
```

Элементы файла, которые можно изменять, выделены жирным шрифтом. Строка шаблона определяет формат вывода записи о событии в журнал. Подробнее с ключами форматирования можно ознакомиться на сайте разработчика (<http://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/PatternLayout.html>).

Настройки класса **log4j.appender.FileLog** не используются.

Уровень детализации вывода событий для классов пространства имен **log4j.logger.com.mobitech** может быть задан из следующего множества:

- **TRACE** — *Уровень трассировки*. Наиболее подробный уровень, выводятся сообщения обо всех ошибках и предупреждениях, информационные, отладочные и трассировочные сообщения.
- **DEBUG** — *Уровень отладки*. Выводятся сообщения обо всех ошибках и предупреждениях, информационные и отладочные сообщения.
- **INFO** — *Информационный уровень*. Выводятся сообщения обо всех ошибках и предупреждениях, а также информационные сообщения.

- **WARN** – *Уровень предупреждений*. Выводятся сообщения только об ошибках и предупреждениях.

- **ERROR** – *Уровень ошибок*. Выводятся сообщения только об ошибках.

- **FATAL** – *Уровень фатальных ошибок*. Выводятся сообщения только об ошибках, вызвавших крах программы.

Для настроек журнала, приведенных выше, журнал имеет следующий вид:

1. **Запись о запуске сервера**. При запуске сервер помещает в журнал запись следующего вида:

```
m.MMSC. Multimedia Messages Service Center.  
Version 1.1.  
  
(C) Mobile Informative Technologies, Ltd. (Mobitech) 2003-2008  
e-mail: info@mobitechtechnologies.com  
www: http://www.mobitechtechnologies.com
```

2. **Запись о событии**. При возникновении события в журнал помещается строка записи о событии, имеющая следующий формат:

```
YYYY-MM-DD HH:mm:ss,sss TYPE [module] message
```

YYYY-MM-DD HH:mm:ss,sss – Дата и время формирования записи о событии в журнал. При этом **YYYY** – год, **MM** – месяц (01-12), **DD** – число (01 – 31), **HH** – часы (00 – 23), **mm** и **SS** – минуты и секунды (00 – 59), а **sss** – миллисекунды.

TYPE – Тип сообщения. Может быть **TRACE**, **DEBUG**, **INFO**, **WARN**, **ERROR** или **FATAL**, в зависимости от типа события, породившего сообщение.

module – Имя модуля или компонента сервера, в котором произошло записанное в журнал событие;

message – Текст диагностического сообщения о событии.

Пример записи в журнале событий сервера:

```
m.MMSC. Multimedia Messages Service Center.  
Version 1.1.  
  
(C) Mobile Informative Technologies, Ltd. (Mobitech) 2003-2008  
e-mail: info@mobitechtechnologies.com  
www: http://www.mobitechtechnologies.com  
  
2009-01-23 04:03:08,888 DEBUG [main] Database connection properties set:      ↓  
mmmc:csmm@jdbc:mysql://mmmc3-1.admin:3306/MMSC  
2009-01-23 04:03:08,889 WARN  [main] Unsupported database description.      ↓  
Node 'NOTIFICATORS'  
2009-01-23 04:03:08,989 INFO  [main] File                                  ↓  
'/var/log/mmc/cdrs_MT/mmsi1_20090123_0016' closed. Copy Stored in          ↓  
'/home/bill/mmsi1_20090123_0016'  
2009-01-23 04:03:09,037 DEBUG [main] SEQ value '17' returned as NEXT value.
```

1.2.Статистика

В случае если в основном файле конфигурации **m.MMSC** заданы настройки ведения технологической статистики, в указанный файл с заданной периодичностью выводится статистическая информация о работе **m.MMSC**.

Статистическая информация выводится в виде записей, каждая запись содержит информацию о работе **m.MMSC** за один период накопления статистической информации.

Статистические записи имеют следующий формат:

```
DT; interval group_name1[ parameter, ... ]; ...; group_nameN[ parameter, ... ]
```

Где:

DT – дата и время формирования статистической записи,

interval – интервал между текущей и предыдущей записью в мс,

group_name – имя группы статистических параметров,

parameter = **name:sum(RPS)** – значение статистического параметра. Для параметра выводится его название (**name**), накопленное значение за период (**sum**), а также максимальное значение RPS параметра за период.

Набор групп параметров зависит от конфигурации **m.MMSC**. Первой группой обязательно следует итоговая группа статистики, не имеющая названия, содержащая суммарную статистику по всем группам. Кроме нее, в статистику могут входить группы параметров работы каждого сервера, заданного в конфигурации серверной стороны **m.MMSC** при этом название группы совпадает со значением атрибута **id**, заданного для сервера (см. описание элемента **<SERVER>**).

Дополнительно в состав статистической записи могут входить также группы параметров, описывающих статистику *Legacy*-продюсера (“**LG.RR**” для продюсера типа «**REROUTER**» и “**LG.WEB**” для продюсера типа «**WEB**») в случае его включения в конфигурации.

Статистические параметры, которые могут встречаться в статистических записях, указаны в таблице 8.1 (параметры перечислены в алфавитном порядке).

Таблица 8.1 - Статистические параметры работы MMSC

Название параметра	Описание	В каких группах встречается
CONF_REQ	Число полученных подтверждений (<i>confirm</i>)	Все, кроме LG.*
DR_REQ	Число полученных отчетов о доставке (<i>delivery report</i>)	Все, кроме LG.*
IN_MSG	Число входящих сообщений	Все, кроме LG.*
IN_REQ	Число входящих запросов	Все, кроме LG.*
LOAD_REQ	Число запросов на загрузку MMS	Все, кроме LG.*

Название параметра	Описание	В каких группах встречается
MSG	Общее число обработанных сообщений (Для LG.* – соответствующим <i>Legacy</i> -продюсером)	Все
PPG_SEND	Число сообщений, направленных через PPG	Только итоговая
PPG_SEND_DR	Число отчетов о доставке, направленных через PPG	Только итоговая
PPG_SEND_RR	Число отчетов о прочтении (<i>read report</i>), направленных через PPG	Только итоговая
PPG_SEND_SM	Число уведомлений о наличии нового MMS, направленных через PPG	Только итоговая
RR_REQ	Число принятых отчетов о прочтении	Все, кроме LG.*
SEND	Общее число отправленных сообщений (Для LG.* – соответствующим <i>Legacy</i> -продюсером)	Все
SEND_DR	Число отправленных отчетов о доставке	Все, кроме LG.*
SEND_REQ	Число отправленных запросов	Все, кроме LG.*
SEND_RR	Число отправленных отчетов о прочтении	Все, кроме LG.*
SEND_SM	Число отправленных уведомлений о наличии нового MMS	Все, кроме LG.*

Пример записи в журнале событий сервера:

Wed Mar 10 13:38:31 MSK 2010; 60000 [DR_REQ:0(0); IN_MSG:0(0); IN_REQ:1(1);	↓
LOAD_REQ:0(0); MSG:0(0); PPG:0(0); PPG DR:0(0); PPG NF:0(0); PPG RR:0(0);	↓
PPG_SEND:1(1); PPG_SEND_DR:0(0); PPG_SEND_RR:1(1); PPG_SEND_SM:0(0); RR_REQ:1(1);	↓
SEND:0(0); SEND DR:0(0); SEND LG:0(0); SEND SM:0(0); SEND_REQ:0(0);] MO_WWW[	↓
DR_REQ:0(0); IN_MSG:0(0); IN_REQ:1(1); LOAD_REQ:0(0); MSG:0(0); RR_REQ:1(1);	↓
SEND_REQ:0(0);]	
Wed Mar 10 13:39:31 MSK 2010; 60000 [DR_REQ:0(0); IN_MSG:0(0); IN_REQ:0(0);	↓
LOAD_REQ:0(0); MSG:0(0); PPG:0(0); PPG DR:0(0); PPG NF:0(0); PPG RR:0(0);	↓
PPG_SEND:0(0); PPG_SEND_DR:0(0); PPG_SEND_RR:0(0); PPG_SEND_SM:0(0); RR_REQ:0(0);	↓
SEND:0(0); SEND DR:0(0); SEND LG:0(0); SEND SM:0(0); SEND_REQ:0(0);] MO_WWW[	↓
DR_REQ:0(0); IN_MSG:0(0); IN_REQ:0(0); LOAD_REQ:0(0); MSG:0(0); RR_REQ:0(0);	↓
SEND_REQ:0(0);]	

В приведенном примере в одной записи выводятся статистические параметры, собранные в две группы – группу общей (итоговой) статистики MMSC (безымянная) и группу статистики сервера, имеющего в настройках **id**, равный **MO_WWW**.