



**МИНИСТЕРСТВО СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
(МИНКОМСВЯЗЬ РОССИИ)**

# **ПРИКАЗ**

№ \_\_\_\_\_

Москва

**Об утверждении Правил  
применения технических и программных средств информационных систем,  
содержащих базы данных абонентов оператора связи и предоставленных им  
услугах связи, обеспечивающих выполнение установленных действий при  
проведении оперативно-розыскных мероприятий**

В целях реализации требований статей 41 и 64 Федерального закона от 7 июля 2003 г. № 126-ФЗ «О связи» (Собрание законодательства Российской Федерации, 2003, № 28, ст. 2895; № 52, ст. 5038; 2004, № 35, ст. 3607; № 45, ст. 4377; 2005, № 19, ст. 1752; 2006, № 6, ст. 636; № 10, ст. 1069; № 31, ст. 3431, ст. 3452; 2007, № 1, ст. 8; № 7, ст. 835; № 31, ст. 4002; 2008, № 18, ст. 1941; 2009, № 29, ст. 3625; 2010, № 7, ст. 705; № 15, ст. 1737; № 27, ст. 3408; № 31, ст. 4190; 2011, № 7, ст. 901; № 9, ст. 1205; № 25, ст. 3535; № 27, ст. 3873, ст. 3880; № 29, ст. 4284, ст. 4291; № 30, ст. 4590; № 45, ст. 6333; № 49, ст. 7061; № 50, ст. 7351, ст. 7366; 2012, № 31, ст. 4322, ст. 4328; № 53, ст. 7578; 2013, № 19, ст. 2326; № 27, ст. 3450; № 30, ст. 4062; № 43, ст. 5451; № 44, ст. 5643; № 48, ст. 6162; № 49, ст. 6339, ст. 6347; № 52, ст. 6961; 2014, № 6, ст. 560; № 14, ст. 1552; № 19, ст. 2302; № 26, ст. 3366, ст. 3377; № 30, ст. 4229, ст. 4273; 2015, № 29, ст. 4342, ст. 4383, ст. 4389; 2016, № 10, ст. 1316, ст. 1318; № 15, ст. 2066; № 18, ст. 2498; № 26, ст. 3873; № 27, ст. 4213, ст. 4221; № 28, ст. 4558; 2017, № 17, ст. 2457) , пункта 2.2 статьи 13 Федерального закона от 6 июля 2016 г. № 374-ФЗ «О внесении изменений в Федеральный закон «О противодействии терроризму» и отдельные законодательные акты Российской Федерации в части установления дополнительных мер противодействия терроризму и обеспечения общественной безопасности» (Собрание законодательства Российской Федерации, 2016, № 28, ст. 4558), и пункта 4 Правил взаимодействия операторов связи с уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность, утвержденных постановлением Правительства Российской Федерации от 27 августа 2005 г. № 538 (Собрание

законодательства Российской Федерации, 2005, № 36, ст. 3704; 2007, № 48, ст. 6010; 2008, № 42, ст. 4832; 2013, № 15, ст. 1804),

**ПРИКАЗЫВАЮ:**

1. Утвердить прилагаемые Правила применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий.

2. Операторам связи до 1 июля 2018 г. привести в соответствие с Правилами применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий, используемые, а также вводимые в эксплуатацию средства связи (оборудование связи), на которые распространяются указанные требования.

3. Направить настоящий приказ на государственную регистрацию в Министерство юстиции Российской Федерации.

Министр

Н.А. Никифоров

УТВЕРЖДЕНЫ  
приказом Министерства связи и  
массовых коммуникаций Российской  
Федерации  
от \_\_\_\_\_ 2017 г. № \_\_\_\_\_

**ПРАВИЛА**  
**применения технических и программных средств**  
**информационных систем, содержащих базы данных абонентов оператора**  
**связи и предоставленных им услугах связи, обеспечивающих выполнение**  
**установленных действий при проведении**  
**оперативно-розыскных мероприятий**

**I. Общие положения**

1. Правила применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий (далее – Правила), разработаны в соответствии со статьями 41 и 64 Федерального закона от 7 июля 2003 г. № 126-ФЗ «О связи» (Собрание законодательства Российской Федерации, 2003, № 28, ст. 2895; № 52, ст. 5038; 2004, № 35, ст. 3607; № 45, ст. 4377; 2005, № 19, ст. 1752; 2006, № 6, ст. 636; № 10, ст. 1069; № 31, ст. 3431, ст. 3452; 2007, № 1, ст. 8; № 7, ст. 835; № 31, ст. 4002; 2008, № 18, ст. 1941; 2009, № 29, ст. 3625; 2010, № 7, ст. 705; № 15, ст. 1737; № 27, ст. 3408; № 31, ст. 4190; 2011, № 7, ст. 901; № 9, ст. 1205; № 25, ст. 3535; № 27, ст. 3873, ст. 3880; № 29, ст. 4284, ст. 4291; № 30, ст. 4590; № 45, ст. 6333; № 49, ст. 7061; № 50, ст. 7351, ст. 7366; 2012, № 31, ст. 4322, ст. 4328; № 53, ст. 7578; 2013, № 19, ст. 2326; № 27, ст. 3450; № 30, ст. 4062; № 43, ст. 5451; № 44, ст. 5643; № 48, ст. 6162; № 49, ст. 6339, ст. 6347; № 52, ст. 6961; 2014, № 6, ст. 560; № 14, ст. 1552; № 19, ст. 2302; № 26, ст. 3366, ст. 3377; № 30, ст. 4229, ст. 4273; 2015, № 29, ст. 4342, ст. 4383, ст. 4389; 2016, № 10, ст. 1316, ст. 1318; № 15, ст. 2066; № 18, ст. 2498; № 26, ст. 3873; № 27, ст. 4213, ст. 4221; № 28, ст. 4558; 2017, № 17, ст. 2457), пункта 2.2 статьи 13 Федерального закона от 6 июля 2016 г. № 374-ФЗ «О внесении изменений в Федеральный закон «О противодействии терроризму» и отдельные законодательные акты Российской Федерации в части установления дополнительных мер противодействия терроризму и обеспечения общественной безопасности» (Собрание законодательства Российской Федерации, 2016, № 28, ст. 4558), и пункта 4 Правил взаимодействия операторов связи с уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность, утвержденных постановлением Правительства Российской Федерации от 27 августа 2005 г. № 538 (Собрание законодательства Российской Федерации, 2005, № 36, ст. 3704; 2007, № 48, ст. 6010; 2008, № 42, ст. 4832; 2013, № 15, ст. 1804) в целях реализации

уполномоченными государственными органами, осуществляющими оперативно-розыскную деятельность, возложенных на них задач.

2. Правила устанавливают обязательные требования к техническим и программным средствам информационных систем, содержащим базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий (далее – ИС БД ОРМ), устанавливаемых на сетях операторов связи.

3. Технические и программные средства информационных систем, содержащие базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий, идентифицируются как оборудование средств связи, включая программное обеспечение, обеспечивающее выполнение установленных действий при проведении оперативно-розыскных мероприятий, и в соответствии с пунктом 29 Перечня средств связи, подлежащих обязательной сертификации, утвержденного постановлением Правительства Российской Федерации от 25 июня 2009 г. № 532 (Собрание законодательства Российской Федерации, 2009, № 26, ст. 3206), подлежат обязательной сертификации в порядке, установленном Правилами организации и проведения работ по обязательному подтверждению соответствия средств связи, утвержденными постановлением Правительства Российской Федерации от 13 апреля 2005 г. № 214 (Собрание законодательства Российской Федерации, 2005, № 16, ст. 1463; 2008, № 42, ст. 4832; 2012, № 6, ст. 687).

## **II. Требования к ИС БД ОРМ**

4. ИС БД ОРМ осуществляет сбор, накопление, хранение и предоставление уполномоченным государственным органам, осуществляющим в соответствии с Федеральным законом от 12 августа 1995 г. № 144-ФЗ «Об оперативно-розыскной деятельности» оперативно-розыскную деятельность (далее – уполномоченные органы), информации об абонентах и оказанных им услугах связи, в том числе о фактах приема, передачи, доставки и (или) обработки голосовой информации, текстовых сообщений, изображений, звуков видео или иных сообщений пользователей услугами связи, а также иной информации, необходимой для выполнения возложенных на уполномоченные органы задач в порядке и случаях, установленных федеральными законами.

Примечание. Под иной информацией следует понимать данные о сети оператора связи необходимые уполномоченному органу для идентификации соединений и сеансов связи.

5. ИС БД ОРМ по запросу от пульта управления уполномоченного государственного органа (далее – ПУ) обеспечивает передачу на ПУ данных о предоставленных абонентам/пользователям следующих услугах:

- 1) услуги местной телефонной связи, за исключением услуг местной телефонной связи с использованием таксофонов и средств коллективного доступа;
- 2) услуги междугородной и международной телефонной связи;

- 3) услуги телефонной связи в выделенной сети связи;
- 4) услуги внутризонавой телефонной связи;
- 5) услуги местной телефонной связи с использованием таксофонов;
- 6) услуги местной телефонной связи с использованием средств коллективного доступа;
- 7) услуги подвижной радиосвязи в сети связи общего пользования;
- 8) услуги подвижной радиосвязи в выделенной сети связи;
- 9) услуги подвижной радиотелефонной связи в сети связи общего пользования;
- 10) услуги подвижной спутниковой радиосвязи;
- 11) услуги связи по предоставлению каналов связи;
- 12) услуги связи в сети передачи данных, за исключением передачи голосовой информации;
- 13) услуги связи по передаче голосовой информации в сети передачи данных;
- 14) телематические услуги доступа к сети Интернет.

6. ИС БД ОРМ представляет собой аппаратно-программный комплекс (далее - АПК), обеспечивающий процессы сбора, накопления, хранения и предоставления по запросу на ПУ следующей информации:

- 1) об абонентах - пользователях услуг связи данной сети связи, с которым заключен договор об оказании таких услуг при выделении для этих целей абонентского номера или уникального кода идентификации;
- 2) о пользователях услугами связи – лицах, заказывающих и (или) использующих услуги связи, и использованных ими телефонных номеров и/или кодов идентификации;
- 3) о предоставляемых абонентам/пользователям данной сети связи услугах связи, о любых действиях абонентов/пользователей при получении услуг связи, обо всех состоявшихся и несостоявшихся соединениях, в том числе попытках соединений, а также сообщениях (текстовых, голосовых и мультимедийных) как доставленных адресату, так и не доставленных и сохраненных на средствах связи данной сети для последующей доставки;
- 4) содержимого текстовых сообщений пользователей, голосовой информации (в части лицензий на услуги связи по предоставлению каналов связи, услуги связи в сети передачи данных, за исключением передачи голосовой информации, телематические услуги связи), изображений, звуков, иных сообщений пользователей услугами связи;
- 5) иной информации, определенной данным документом как необходимой уполномоченному органу для проведения ОРМ.

7. В случае, если оператор связи предоставляет пользователям дополнительный информационный сервис с использованием сети передачи, ИС БД ОРМ обеспечивает сбор, накопление и хранение информации для сервиса в соответствии с п.11 настоящих Правил.

8. ИС БД ОРМ размещаются на территории Российской Федерации.

### III. Общие функциональные требования

9. ИС БД ОРМ обеспечивает:

9.1. Сбор из различных источников сети/сетей связи и первичную обработку информации, состав которой определяется данным документом, для последующего её накопления, хранения и предоставления по запросу.

9.2. Накопление, хранение до 6 месяцев содержимого текстовых сообщений пользователей, голосовой информации (в части лицензий п.5 пп. 11, 12, 14), изображений, звуков, иных сообщений пользователей услугами связи с момента окончания их приема, передачи, доставки и (или) обработки.

9.3. Контроль времени поступления из различных источников сети/сетей связи собираемых данных и информирование ПУ о превышении временных параметров, предусмотренных для соответствующих типов данных.

9.4. Накопление, хранение не менее 3-х лет и обработку информации, об абонентах/пользователях услуг связи данной сети, о выделенных им телефонных номерах и кодах идентификации, об оказываемых им услугах связи и иной информации, определенной данным документом как необходимой уполномоченному органу для проведения ОРМ. При этом данные об услугах связи, оказываемых абонентам/пользователям услуг связи данной сети, поступающие из источников информации накапливаются, не зависимо от того присутствует или нет в ИС БД ОРМ информация о выделении оператором связи телефонных номеров и/или кодов идентификации, присутствующих в поступающей информации.

9.5. Защиту от несанкционированного доступа (далее – НСД) к данным, связанным с проведением ОРМ, с помощью программных и технических средств, как с рабочих мест обслуживающего персонала оператора связи, так и с использованием удаленного доступа:

- 1) производителя ИС БД ОРМ;
- 2) неавторизованных пользователей;
- 3) технического персонала оператора связи;
- 4) третьих лиц.

Примечание. К данным, связанным с проведением ОРМ, относится информация, поступающая в ИС БД ОРМ с ПУ, и информация, подготовленная к передаче из ИС БД ОРМ на ПУ.

9.6. Информирование ПУ о попытках:

- 1) доступа в ИС БД ОРМ к данным, связанным с проведением ОРМ, с использованием команд или сервисных программ;
- 2) обращения к оперативной памяти ИС БД ОРМ, содержащей данные, связанные с проведением ОРМ;
- 3) резервного копирования данных, связанных с проведением ОРМ;
- 4) доступа к ИС БД ОРМ через порты, не предусмотренные для доступа к ИС БД ОРМ;
- 5) вскрытия корпуса технических средств приема запросов от ПУ.

9.7. Контроль работоспособности и загруженности АПК ИС БД ОРМ.

9.8. Доступ с ПУ и уполномоченного технического персонала для обслуживания АПК ИС БД ОРМ в соответствии с установленными правами доступа.

9.9. Круглосуточный удаленный доступ со стороны ПУ в соответствии с Приложением № 9 к настоящим Правилам.

9.10. Реализацию протокола взаимодействия оборудования ИС БД ОРМ и оборудования ПУ, представленного в Приложении № 9 к настоящим Правилам.

9.11. Прием от ПУ различных видов запросов об абонентах/ пользователях услуг связи и предоставленных им услугах связи, предусмотренных данным документом.

9.12. Передачу на ПУ из ИС БД ОРМ отобранных данных, соответствующих поступившим с ПУ запросам.

9.13. Взаимодействие с техническими средствами ОРМ в соответствии с протоколом Приложения № 3 «Правил применения оборудования систем коммутации, включая программное обеспечение, обеспечивающего выполнение установленных действий при проведении оперативно-розыскных мероприятий в сетях подвижной радиосвязи, подвижной радиотелефонной связи, подвижной спутниковой радиосвязи, междугородной и международной телефонной связи, телефонной связи в выделенной сети связи, внутризонавой телефонной связи, местной телефонной связи, передачи данных с целью передачи голосовой информации, включая систему хранения» для получения статистической информации, текстовых сообщений, голосовой информации (звуков), видеовызовов по запросу и передачи отобранных результатов в соответствии с Приложением № 9 к настоящим Правилам.

9.14. Ведение системных журналов, содержащих информацию о работе ИС БД ОРМ в части, не содержащей данные, связанные с проведением ОРМ, и включающих в себя:

1) информацию о поступлении данных из различных источников сети/сетей связи;

2) информацию о сессиях с ПУ, установленных и не установленных;

3) информацию о запросах на получение данных, за исключением данных самого запроса;

4) информацию об ответах на запросы получения данных;

5) информацию об отчетах;

6) информацию о текущей конфигурации АПК, системного и прикладного ПО;

7) информацию об изменениях в конфигурации АПК, системного и прикладного ПО;

8) информацию о сбоях в АПК, системном и прикладном программном обеспечении (далее – ПО);

9) информацию об обращении и доступе обслуживающего технического персонала оператора связи к ИС БД ОРМ. Описание системы контроля работоспособности и загруженности АПК ИС БД ОРМ предоставляется производителями ИС БД ОРМ.

9.15. Доступ уполномоченного технического персонала для выполнения регламентных и ремонтных работ к системным журналам ИС БД ОРМ и ПО АПК, в соответствии с правами, установленными парольной системой доступа с обязательной регистрацией команд и сообщений, используемых техническим персоналом при обращении к ИС БД ОРМ.

9.16. Сохранность и доступность для дальнейшего использования ранее накопленных данных при модернизации аппаратных и программных средств ИС БД ОРМ.

9.17. Возможность ввода в эксплуатацию сетей и средств связи с любым набором услуг, согласно Лицензий на предоставление услуг связи.

10. Для уполномоченного технического персонала ИС БД ОРМ предоставляет:

- 1) доступ к журналу ошибочных блоков переданных отчетов и возможность редактирования ошибочных записей соответствующих отчетов;
- 2) доступ к аппаратным и программным компонентам ИС БД ОРМ для проведения регламентных и ремонтных работ.

11. ИС БД ОРМ собирает и накапливает информацию:

- 1) о фактах состоявшихся соединениях абонентов/пользователей услуг сети связи;
- 2) о фактах несостоявшихся соединениях, включая попытки установления соединения, абонентов/пользователей услуг сети связи.

Примечание 1. Кратковременное соединение, не подлежащее тарификации с точки зрения системы учета сети связи, считается состоявшимся с точки зрения проведения ОРМ.

Примечание 2. Информация об услугах связи, предоставленных в рамках действующих лицензий на оказание услуг связи, заносится в ИС БД ОРМ в соответствии с Приложением № 1 к Правилам.

12. ИС БД ОРМ накапливает и хранит информацию о следующих абонентах/пользователях, получающих услуги связи в сети связи:

- 1) абонентах/пользователях услуг связи, постоянно закрепленных в пределах сети связи;
- 2) пользователях услуг связи, временно закрепленных в пределах данной сети связи (например, роумеры и др.);
- 3) пользователях услуг связи, обращающихся к информационному ресурсу, принадлежащему или присоединенному к данной сети связи.

13. ИС БД ОРМ выполняет сбор и накопление информации о следующих видах соединений, совершенных абонентами/пользователями посредством услуг сети передачи данных (в части лицензий п.5 пп. 11, 12, 14):

- 1) подключении/отключении абонента к сети передачи данных (AAA);
- 2) HTTP-обращения к информационному ресурсу сети связи (посещение Интернет-страниц сайтов, порталов и т.д.);
- 3) передаче почтового e-mail сообщения (в т.ч. сообщения веб-почты);
- 4) передаче электронных сообщений между пользователями (служебных сообщений, мгновенных сообщений, коротких сообщений, мультимедийных сообщений);

- 5) голосовой связи посредством сети передачи данных;
- 6) передаче файловых данных;
- 7) терминального доступа к оборудованию для удаленного управления;
- 8) прочих видов соединений абонентов в сети передачи данных;
- 9) информацию об изменении сетевых адресов пользователей, если такая замена (трансляция) сетевых адресов в процессе оказания услуг связи осуществляется на оборудовании связи сети передачи данных оператора

Примечание 1. ИС БД ОРМ для корпоративных абонентов, подключаемых к сети связи с использованием оборудования, осуществляющего сокрытие внутренней адресации корпоративной сети, предусматривает возможность регистрации как кодов идентификации выделенных данному абоненту сетью связи, так и адресацию, используемой внутри корпоративной сети в случае ее доступности.

Примечание 2. Информация п.1-7 хранится не менее 3-х лет, информация п. 8-9 хранится сроком 6 мес.

14. ИС БД ОРМ собирает, накапливает и хранит информацию о следующих видах телефонных соединений и сеансах связи пользователей услуг телефонной связи:

- 1) телефонных соединениях;
- 2) входящих/исходящих текстовых коротких сообщениях, как доставленных, так и не доставленных абоненту;
- 3) служебных запросах;
- 4) прочих видах телефонных соединений, формируемых абонентами в телефонной сети связи.

Примечание 1. При взаимодействии ИС БД ОРМ с ПУ (запросы/отчеты) телефонные номера абонентов Российской Федерации имеют международный формат согласно Е.164.

Примечание 2. ИС БД ОРМ для абонентского оборудования с функцией коммутации предусматривает возможность регистрации как номеров телефонов, выделенных сетью связи, так и внутренних номеров телефонов, используемых в абонентском оборудовании в случае их доступности.

15. ИС БД ОРМ собирает, накапливает и хранит информацию об изменении местоположения абонентов в соответствии с Приложением № 9 к Правилам (Locations.asn), доступную в сети связи как при предоставлении услуги связи, так и в пассивном режиме (ожидании вызова) (при переключении его обслуживания, выполняемого разными устройствами сети между разными географическими зонами обслуживания, при включении-выключении абонентского устройства, при рассылке запросов от средств связи).

16. ИС БД ОРМ при сборе информации о соединениях абонентов в сети передачи данных:

- 1) в случае если данные передавались, формирует одну общую запись о соединении в сети передачи с суммированием количества принятой, переданной информации в течение пяти минут по одной и той же паре IP-адресов и TCP/UDP-портов: для соединений терминального доступа к оборудованию, прочих видов соединений абонентов в сети передачи данных;

2) для HTTP-соединений абонентов в сети передачи данных с информационными ресурсами фиксируются сведения согласно Приложению № 1 к Правилам. В качестве URL HTTP-ресурса заполняется адрес страницы, запрошенной пользователем.

17. ИС БД ОРМ выполняет сбор содержимого соединений пользователей услугами связи:

1) с интерфейсов точек консолидации согласно Приложению № 11 к Правилам;

2) пассивно, исключая передачу информации в сеть связи.

18. ИС БД ОРМ выполняет накопление и хранение информации текстовых сообщений пользователей, голосовой информации, изображений, звуков, иных сообщений пользователей услугами связи в форматах в соответствии с Приложением № 12 к Правилам.

19. ИС БД ОРМ выполняет накопление:

1) для телефонных (в т.ч. подвижных) сетей связи:

- информации всех текстовых сообщений электросвязи;
- иных, в т.ч. факсимильных сообщений;
- передаваемой/принимаемой от абонентов справочной информации, услугах ДВО;
- прочей служебной информации передаваемой в сеть связи в ходе соединения, в т.ч. при регистрации и прекращении регистрации абонента в сети связи;

2) для сетей передачи данных:

- содержимого голосовой информации пользователей;
- информации всех текстовых сообщений пользователей передаваемых без дополнительного кодирования;
- содержимого прочих видов соединений в соответствии с устанавливаемыми с ПУ фильтрами содержимого соединений сети передачи данных.

20. ИС БД ОРМ для сетей передачи данных выполняет сбор, накопление и хранение содержимого прочих видов соединений только при установленных с ПУ фильтров содержимого соединений.

21. ИС БД ОРМ не должна оказывать влияние на работоспособность средств связи, собственные информационные системы оператора.

22. ИС БД ОРМ, осуществляющая сбор, накопление, хранение и обработку информации, определенной Правилами, нескольких филиалов оператора связи, если структура организации сети связи оператора связи обслуживает одновременно несколько входящих в его структуру территориально-распределенных филиалов по субъектам РФ, обеспечивает предоставление информации по запросу ПУ соответствующих субъектов Российской Федерации.

22.1. ИС БД ОРМ осуществляющая сбор, накопление, хранение и обработку информации, определенной Правилами, других сетей связи, обеспечивает предоставление информации по запросу ПУ тех регионов, где предоставляются услуги.

22.2. ИС БД ОРМ осуществляющая сбор, накопление, хранение и обработку информации распределенных сетей оператора связи и /или нескольких сетей связи, обеспечивает выполнение поисковых задач для тех сетей и филиалов, которые заданы с ПУ в запросе. При явном указании списка филиалов и/или других юридических лиц в параметрах поисковой задачи, запрос выполняется в соответствии с временными ограничениями, приведенными в разделе IV к Правилам.

23. Требования к составу накапливаемой ИС БД ОРМ информации о соединениях абонентов устанавливаются согласно Приложению № 1 к Правилам.

24. Требования к интерфейсу взаимодействия ПУ и ИС БД ОРМ устанавливаются согласно Приложению № 2 к Правилам.

25. Требования к функционированию канала передачи данных (кпд1) устанавливаются согласно Приложению № 3 к Правилам.

26. Требования к функционированию канала передачи данных (кпд2) устанавливаются согласно Приложению № 4 к Правилам.

27. Требования к функционированию канала передачи данных (кпд3) устанавливаются согласно Приложению № 5 к Правилам.

28. Требования к функционированию канала передачи данных (кпд4) устанавливаются согласно Приложению № 6 к Правилам.

29. Требования к функционированию канала передачи данных (кпд5) устанавливаются согласно Приложению № 7 к Правилам.

30. Требования к форматам сообщений ИС БД ОРМ устанавливаются согласно Приложению № 8 к Правилам.

31. Требования к параметрам ASN.1-спецификации протокола взаимодействия пункта управления ОРМ и ИС БД ОРМ устанавливаются согласно Приложению № 9 к Правилам.

32. Список используемых сокращений и терминов приведен в Приложении № 14 к Правилам (справочно).

#### **IV. Требования, предъявляемые к временным характеристикам обработки, поиска информации и выполнения задач ИС БД ОРМ**

##### **Требования, предъявляемые к временным характеристикам ИС БД ОРМ**

33. Время предварительной обработки информации с момента ее поступления в ИС БД ОРМ до момента, когда она становится доступной для выполнения запросов с ПУ не превышает следующих значений:

1) 5 с - для данных об абонентах сети связи и пользователей услуг связи, получивших телефонные номера и/или коды идентификации для доступа к услугам сети связи;

2) 5 с - для данных о кодах идентификации для доступа к услугам сети связи, предоставленных пользователям услуг связи путем распространения карточек доступа;

3) 30 с - для данных о кодах идентификации для доступа к услугам сети связи, предоставленных пользователям услуг связи через SMS сообщения;

4) 30 с - для данных о временных кодах идентификации, выделяемых пользователям услуг связи в ходе предоставления доступа к сети связи или услугам сети связи;

5) 30 с - для данных об изменениях в сети связи;

6) 5 минут с момента наступления события - для данных о предоставляемых услугах телефонной связи, о смене местоположения абонентов;

7) 10 минут с момента наступления события - для данных о соединениях, совершенных абонентами посредством сети передачи данных.

34. Время выполнения задачи в ИС БД ОРМ не превышает следующих значений:

1) 1 с - для данных о принадлежности идентификаторов абонентов сети оператора связи;

2) 1 с - для данных об использовании карты экспресс оплаты, телефонной карты;

3) 3 с - для данных об идентификаторах абонентов зарегистрированных на физическое или юридическое лицо;

4) 1 с - для данных о пополнении баланса личного счета абонента.

35. Время выполнения задач поиска информации о связях абонентов, накопленных в ИС БД ОРМ не превышает значений, приведенных в таблице 1.

35.1. ИС БД ОРМ обеспечивает выполнение не менее 100 одновременно выполняющихся поисковых задач. При выполнении от 1-ой до 100 одновременно запущенных задач обеспечиваются временные характеристики, определенные в таблице № 1. При выполнении ИС БД ОРМ более 100 одновременных поисковых задач, обеспечение временных характеристик в соответствии с таблицей № 1 не гарантируется.

35.2. Требования к временным характеристикам поиска информации в ИС БД ОРМ разделяются по следующим классам:

35.2.1. Класс 1 (К1) включает критерии:

1) номер телефона абонента сети оператора;

2) идентификатор подвижной станции абонента (IMEI, ESN/MIN);

3) идентификатор подвижного абонента (IMSI);

4) номер телефонной (таксофонной) карты.

35.2.2. Класс 2 (К2) включает критерии:

1) идентификатор абонента сети передачи данных, в т.ч. другого оператора связи;

2) идентификатор пользователя в сети передачи данных, в т.ч. другого оператора связи;

3) идентификатор прикладного уровня при использовании услуг сети передачи данных.

35.2.3. Класс 3 (К3) включает критерии:

1) номер базовой станции;

2) номер пучка;

3) номер коммутатора;

4) идентификатор узла связи;

5) идентификатор устройства сети передачи данных;

35.2.4. Класс 4 (К4) включает критерии:

1) идентификатор содержимого соединения.

Таблица № 1. Время выполнения задач поиска информации о связях абонентов, обслуженных сетью ИС БД ОРМ

№	Класс параметра запроса	Временной интервал				
		до суток	до 1 месяца	до 6 месяцев	До 1 года	до 3 лет
1	К1	< 3 с	< 5 с	< 15 с	< 25 с	< 50 с
2	К2	< 1 мин	< 10 мин	< 15 мин	< 1 часа	< 1,5 часов
3	К3	< 7 мин				
4	К4	1 с				

Примечание. Для отсутствующих в таблице идентификаторов, время выполнения задачи в ИС БД ОРМ может быть больше приведенного.

36. ИС БД ОРМ поддерживает возможность выполнения комбинированных запросов. Комбинированными запросами являются поисковые критерии согласно пункту 30 Правил, объединенные логическими операциями.

36.1. ИС БД ОРМ поддерживает следующие логические операции для объединения критериев: «И», «ИЛИ», операции группировки критериев «(», «)» и логическое «НЕ».

36.2. Использование критериев с использованием операции логического «НЕ» допускается только совместно с другими критериями, не являющимися логическим отрицанием, и объединяется с ними с помощью логических операций объединения критериев. Результирующий комбинированный запрос должен иметь хотя бы одну логическую операцию «И».

36.3. К времени поиска информации в базе данных ИС БД ОРМ при выполнении комбинированных поисковых запросов предъявляются следующие требования:

1) для операции «И»: время выполнения поискового запроса по двум и более критериям должно быть меньше суммарного времени выполнения поисковых запросов по каждому из критериев;

2) для операции «ИЛИ»: время выполнения поискового запроса по двум и более критериям должно быть не больше суммарного времени выполнения поисковых запросов по каждому из критериев;

3) для операции «НЕ»: время выполнения поискового запроса, в котором один из критериев задан с операцией «НЕ», не должно превышать времени выполнения аналогичного запроса, в котором используется соответствующий критерий без операции «НЕ».

4) в случае, если одним из поисковых критериев является составной критерий с использованием операций группировки, для определения времени выполнения исходного запроса время выполнения запроса с составным критерием должно оцениваться согласно требованиям пп.1-4.

Приложение № 1  
к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

**Требования к составу накапливаемой ИС БД ОРМ информации о соединениях и местоположении абонентов, справочной информации**

1. Идентификатор абонента в сетях персонального радиовызова, фиксированной сети телефонной связи, мобильной сети связи состоит из:
  - 1) ТФОП-идентификатора, включающего:
    - а) телефонный номер в международном формате;
    - б) внутренний номер, если есть;
  - 2) GSM-идентификатора, включающего:
    - а) телефонный номер в международном формате;
    - б) идентификатор мобильного абонента;
    - в) идентификатор мобильной станции;
    - г) идентификатор SIM-карты абонента (ICC);
  - 3) CDMA-идентификатора, включающего:
    - а) телефонный номер в международном формате;
    - б) идентификатор мобильного абонента;
    - в) идентификатор мобильной станции;
    - г) идентификатор мобильного абонента (CDMA);
    - д) идентификатор SIM-карты абонента (ICC);
  - 4) идентификатора сети передачи данных, включающего:
    - а) идентификатор пользовательского оборудования;
    - б) имя пользователя – login;
    - в) IP адрес;
    - г) адрес электронной почты;
    - д) PIN;
    - е) телефонный номер абонента сети передачи голосовой информации посредством сети передачи данных;
    - ж) пользовательский домен;
  - 5) идентификатора сети персонального радиовызова;
    - б) идентификатора в сети передачи голосовой информации посредством сети передачи данных, включающего:
      - а) IP-адрес абонента;
      - б) общедоступное имя инициатора связи;

в) телефонный номер абонента сети передачи голосовой информации посредством сети передачи данных.

2. ИС БД ОРМ накапливает следующую информацию о соединениях абонентов в сети персонального радиовызова:

- 1) дату и время соединения;
- 2) тип соединения;
- 3) идентификаторы абонента;
- 4) объем переданных данных;
- 5) причину завершения соединения;
- 6) идентификатор оператора связи или филиала.

3. ИС БД ОРМ накапливает следующую информацию о соединениях абонентов при предоставлении услуг фиксированной телефонной связи:

- 1) дату и время начала соединения;
- 2) длительность соединения, с;
- 3) тип соединения;
- 4) тип вызывающего абонента;
- 5) коммутатор, обслуживший соединение;
- 6) тип вызываемого абонента;
- 7) входящий пучок;
- 8) исходящий пучок;
- 9) код пограничного коммутатора;
- 10) причину завершения соединения;
- 11) идентификатор оператора связи или филиала;
- 12) услуги связи при соединении;
- 13) номер телефонной карты;
- 14) идентификаторы вызывающего абонента;
- 15) набранный номер вызываемого абонента;
- 16) идентификаторы вызываемого абонента;
- 17) телефонный номер при переадресации;
- 18) текстовое содержимое сообщения;
- 19) идентификатор содержимого соединения.

4. ИС БД ОРМ накапливает следующую информацию о соединениях абонентов при предоставлении услуг подвижной телефонной связи:

- 1) дату и время начала соединения;
- 2) длительность соединения, с;
- 3) тип соединения;
- 4) услуги связи при соединении;
- 5) тип вызывающего абонента;
- 6) коммутатор, обслуживший соединение;
- 7) тип вызываемого абонента;
- 8) входящий пучок;
- 9) исходящий пучок;
- 10) код пограничного коммутатора;
- 11) код роумингового партнера;
- 12) причину завершения соединения;

- 13) идентификатор оператора связи или филиала;
- 14) идентификаторы вызывающего абонента;
- 15) местоположение вызывающего абонента на начало вызова;
- 16) местоположение вызывающего абонента на конец вызова;
- 17) идентификаторы вызываемого абонента;
- 18) местоположение вызываемого абонента на начало вызова;
- 19) местоположение вызываемого абонента на конец вызова.
- 20) телефонный номер при переадресации;
- 21) текстовое содержимое сообщения;
- 22) идентификатор содержимого соединения.

5. ИС БД ОРМ накапливает следующую информацию о соединениях абонентов в сети передачи данных.

5.1. Информацию о подключении/отключении абонента к сети передачи данных (AAA), включающую:

- 1) дату и время подключения/отключения к сети передачи данных;
- 2) тип соединения (подключение к сети передачи данных, отключение от сети передачи данных, смена местоположения);
- 3) идентификатор сессии;
- 4) выделенный динамический IP-адрес;
- 5) имя пользователя (логин);
- 6) код протокола в соответствии с RFC 1700, либо номер порта для TCP/UDP;
- 7) вызывающий номер;
- 8) вызываемый номер;
- 9) IP-адрес/порт NAS-сервера;
- 10) объем принятых данных, байт;
- 11) объем переданных данных, байт;
- 12) идентификатор оператора связи или филиала;
- 13) идентификатор пользовательского оборудования;
- 14) наименование GPRS точки доступа (APN);
- 15) IP-адрес GPRS SGSN;
- 16) IP-адрес GPRS GGSN;
- 17) код зоны обслуживания (SAC) GRPS;
- 18) базовую станцию, через которую установлено соединение (передача данных посредством подвижной сети связи);
- 19) базовую станцию, через которую завершено соединение (передача данных посредством подвижной сети связи);
- 20) номер телефонной карты;
- 21) IMSI мобильного абонента;
- 22) идентификатор мобильной станции абонента;
- 23) номер модемного пула;

5.2. Общую информацию для всех записей об использовании абонентами услуг связи и ресурсов сети передачи данных (за исключением соединений подключения/отключения к сети связи, трансляции сетевых адресов):

- 1) идентификатор оператора связи или филиала;
- 2) дату и время начала соединения;
- 3) дату и время завершения соединения;
- 4) информацию о клиенте (IP/порт);
- 5) информацию о сервере (IP/порт);
- 6) информацию о трансляции (IP/порт);
- 7) код протокола в соответствии с RFC1700 либо номер порта для TCP/UDP;

8) идентификатор точки подключения к сети передачи данных, с которой получена запись о соединении;

Примечание: допускается накопление информации о трансляции в составе общей информации об использовании абонентами услуг и ресурсов сети передачи данных.

5.3. Информацию о HTTP-соединениях с информационными ресурсами сети передачи данных (посещение Интернет-страниц), включающую:

- 1) URL (адрес, наименование информационного ресурса);
- 2) объем принятых и переданных данных в соединении (включает как соединения управления, так и передачи данных), байт;
- 3) причину завершения соединения;
- 4) идентификатор абонента (AAA-имя пользователя/логин);
- 5) идентификатор содержимого соединения.

5.4. Информацию о соединениях передачи почтовых e-mail сообщений, включающую:

- 1) отправитель почтового сообщения;
- 2) список получателей почтового сообщения;
- 3) список получателей копии почтового сообщения;
- 4) адрес, куда следует отправлять ответ на письмо;
- 5) тему почтового сообщения;
- 6) размер почтового сообщения, включая прикрепленные файлы, байт;
- 7) наличие прикрепленных файлов в письме (да/нет);
- 8) список текстовых наименований почтовых серверов, через которые отправлено письмо (в т.ч. сервера веб-почты);
- 9) причину завершения соединения;
- 10) протокол, при помощи которого отправлено сообщение;
- 11) тип пользовательской операции;
- 12) идентификатор абонента (AAA-имя пользователя/логин);
- 13) текстовое содержимое сообщения;
- 14) идентификатор содержимого соединения.

5.5. Информацию о соединениях передачи электронных сообщений между пользователями (служебных сообщениях, мгновенных сообщениях, коротких сообщениях, мультимедийных сообщениях, отправленные посредством сети передачи данных), включающую:

- 1) наименование учетной записи пользователя при подключении;
- 2) общедоступное имя отправителя;
- 3) пользовательский идентификатор отправителя (в т.ч. для веб-чата);

4) список получателей, включающий для каждого получателя сообщения: общедоступное имя получателя, пользовательский идентификатор получателя (в т.ч. для веб-чата);

5) размер данных сессии, байт;

6) причину завершения соединения;

7) протокол, при помощи которого отправлены сообщения;

8) тип пользовательской операции;

9) идентификатор абонента (AAA-имя пользователя/логин);

10) текстовое содержимое сообщения;

11) идентификатор содержимого соединения.

5.6. Информацию о соединениях голосовой связи посредством сети передачи данных, включающую:

1) идентификатор сессии/call-id;

2) идентификатор конференции;

3) длительность разговора, с;

4) общедоступное имя инициатора связи;

5) способ подключения;

6) номер вызывающего абонента;

7) номер вызываемого абонента;

8) объем переданных данных (включает как соединения управления, так и передачи данных), байт;

9) объем принятых данных (включает как соединения управления, так и передачи данных), байт;

10) наличие переданной в соединении факсовой информации (Т.38);

11) причину завершения соединения;

12) входящий пучок (идентификатор направления/оконечного оборудования вызывающей стороны);

13) исходящий пучок (идентификатор направления/оконечного оборудования вызываемой стороны);

14) идентификаторы медиашлюзов, обслуживших соединение;

15) протокол передачи;

16) услуги связи при соединении;

17) идентификатор абонента (AAA-имя пользователя/логин);

18) идентификатор содержимого соединения.

5.7. Информацию о соединениях передачи файловых данных, включающую:

1) название сервера;

2) имя пользователя, наименование учетной записи;

3) режим работы файлового сервера;

4) объем переданных данных (включает как соединения управления, так и передачи данных), байт;

5) объем принятых данных (включает как соединения управления, так и передачи данных), байт;

6) причину завершения соединения;

7) идентификатор абонента (AAA-имя пользователя/логин);

8) идентификатор содержимого соединения.

5.8. Информацию о соединениях терминального доступа к оборудованию для удаленного управления, включающую:

1) объем переданных данных (включает как соединения управления, так и передачи данных), байт;

2) объем принятых данных (включает как соединения управления, так и передачи данных), байт;

3) протокол удаленного доступа к оборудованию;

4) идентификатор абонента (AAA-имя пользователя/логин);

5) идентификатор содержимого соединения;

5.9. Информацию о соединениях передачи прочих данных принимаемых, получаемых абонентом при помощи закрытых протоколов обмена, включающую:

1) объем переданных данных (включает как соединения управления, так и передачи данных), байт;

2) объем принятых данных (включает как соединения управления, так и передачи данных), байт;

3) протокол передачи данных;

4) идентификатор абонента (AAA-имя пользователя/логин);

5) идентификатор содержимого соединения;

5.10. Информацию о трансляции сетевых адресов пользователей, накапливаемую отдельными ИС БД ОРМ по согласованию с ПУ, включающую:

1) дату и время трансляции;

2) тип записи (начало/окончание трансляции);

3) внутренний сетевой адрес и порт;

4) внешний сетевой адрес и порт;

5) сетевой адрес назначения и порт;

6) тип трансляции сетевых адресов.

6. ИС БД ОРМ накапливает следующую информацию об изменении местоположения абонентов:

1) дату и время определения местоположения;

2) идентификатор абонента;

3) информацию о местоположении для услуг телефонной связи в сетях подвижной радиосвязи: код зоны, номер сектора базовой станции, компенсацию задержки прохождения сигнала от мобильной станции;

4) информацию о местоположении для услуг передачи данных в сети подвижной связи: код зоны/контроллера доступа, номер сектора базовой станции, либо MAC-адрес сетевого оборудования базовой станции и идентификатор сектора; информацию о широте/долготе местоположения абонента с указанием типа проекции координат (в случае технической возможности накопления).

7. ИС БД ОРМ накапливает информацию о пользователях услугами связи в соответствии с Приложением № 9 к Правилам (ReportsAbonents.asn).

8. ИС БД ОРМ накапливает информацию о платежах, совершенных абонентами в соответствии с Приложением № 9 к Правилам (ReportsPayments.asn).

9. ИС БД ОРМ накапливает справочную информацию в соответствии с Приложением № 9 (Dictionaries.asn):

- 1) пучки соединительных линий (идентификатор направления/оконечного оборудования вызываемой/вызывающей стороны);
- 2) базовые станции;
- 3) роуминговые партнеры;
- 4) коммутаторы;
- 5) IP шлюзы, в том числе: SGSN, GGSN, SMSC, VOIP-шлюзы, AAA-сервера, GMSC, PSTN-шлюзы;
- 6) типы вызовов;
- 7) список услуг связи;
- 8) способы оплаты (пополнения баланса);
- 9) причины завершения соединения;
- 10) справочник с планами IP-адресации;
- 11) план телефонной номерной емкости;
- 12) типы документов, удостоверяющих личность;
- 13) операторы связи, обслуживаемые ИС БД ОРМ;
- 14) идентификаторы точек подключения к сети передачи данных, с которых получены записи о соединениях;
- 15) специальные номера оператора связи, в т.ч.: короткие номера, номера технических служб, групповые номера, номера MSRN-пулов, собственные телематические сервера оператора (почтовые, файлообменные, развлекательные и прочие) а также иное оборудование сети передачи данных, предоставляющее доступ абонентов к собственным информационным ресурсам оператора;
- 16) карта связей пучков соединительных линий;
- 17) план нумерации идентификаторов мобильных телефонных абонентов.

Справочная информация накапливается в объеме достаточном для декодирования идентификаторов услуг, местоположения, а также иной информации об абонентах, подключенных услугах, соединениях, трафике накапливаемой ИС БД ОРМ.

---

## Приложение № 2

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

### Требования, предъявляемые к интерфейсу взаимодействия между ПУ и ИС БД ОРМ

1. ИС БД ОРМ подключаются к ПУ в точках подключения выделенными каналами связи. Интерфейс в точке подключения соответствует спецификации Ethernet 10/100/1000 Base-T, либо 100 Base-FX, 1000 Base-LN по согласованию с уполномоченными органами.

2. В качестве протокола передачи данных используется протокол TCP/IP.

2.1. Сетевой интерфейс первого уровня с протоколом TCP/IP используется для первичного подключения аппаратуры передачи данных – модемов, маршрутизаторов и т.п.

2.2. Для каждой ИС БД ОРМ внутри сетевого интерфейса первого уровня рекомендуется создавать свою виртуальную сеть VPN (Virtual Private Network) для туннелирования всего рабочего TCP/IP трафика между ИС БД ОРМ и ПУ, которая соответствует спецификации L2TP, IPSec VPN.

3. Пропускная способность каналов передачи данных между ИС БД ОРМ и ПУ в точках подключения соответствует данным, приведенным в Таблице № 1.

Таблица № 1. Пропускная способность каналов передачи данных между ИС БД ОРМ и ПУ

Емкость абонентской базы (тыс. абонентов), не более	1	10	100	1000	10000
Скорость передачи данных (кбит/с), не менее	2048	4096	10000	10000	100000

Для доставки на ПУ от ИС БД ОРМ информации содержимого соединений организовывается канал передачи данных:

- для сетей подвижной радиосвязи в сети связи общего пользования, подвижной радиосвязи в выделенной сети связи, подвижной радиотелефонной связи в сети связи общего пользования, подвижной радиотелефонной связи (при использовании бизнес-модели виртуальных сетей подвижной радиотелефонной связи),

подвижной спутниковой радиосвязи, сетей передачи данных с целью передачи голосовой информации в соответствии с Таблицей № 2:

Таблица № 2. Пропускная способность каналов передачи данных между ИС БД ОРМ ОРМ и ПУ для услуг подвижной телефонной связи

Емкость абонентской базы (тыс. абонентов), не более	10	100	1000	10000	Более 10000
Скорость передачи данных (Мбит/с), не менее	4	10	100	300	500

- для сетей подвижной спутниковой радиосвязи, междугородной и международной телефонной связи, телефонной связи в выделенной сети связи, внутризональной телефонной связи, местной телефонной связи, голосовой связи посредством сети передачи данных в соответствии с Таблицей № 3:

Таблица № 3. Пропускная способность каналов передачи данных между ИС БД ОРМ и ПУ для услуг фиксированной телефонной связи

Емкость абонентской базы (тыс. абонентов), не более	10	100	100 и более
Количество пучков каналов	1	10	100 и более
Скорость передачи данных (Мбит/с), не менее	4	10	100

- для услуг связи по предоставлению каналов связи, услуг связи в сети передачи данных, за исключением передачи голосовой информации, телематических услуг связи в соответствии с Таблицей № 4:

Таблица № 4. Пропускная способность каналов доставки содержимого соединений передачи данных ИС БД ОРМ и ПУ

Скорость потока информации сетевого трафика обрабатываемого ИС БД ОРМ, Мбит/с, не менее	Суммарная скорость передачи данных на выходе ИС БД ОРМ, предназначенном для связи с ПУ, Мбит/с
100	Не менее 5% от скорости потока информации обрабатываемого ИС БД ОРМ
400	
900	
4000	>100
9000	>100
20000	>1000
100000	>1000

4. Взаимодействие ИС БД ОРМ с оборудованием ПУ на транспортном уровне осуществляется по схеме «клиент – сервер»:

- 1) в качестве «сервера» выступает ИС БД ОРМ;
- 2) в качестве «клиента» выступает оборудование ПУ.

4.1. Для взаимодействия ИС БД ОРМ с ПУ организованы пять каналов передачи данных (далее – кпд):

- 1) кпд1 – канал управления;
- 2) кпд2 – канал данных;
- 3) кпд3 – канал мониторинга;
- 4) кпд4 – канал неформатированных данных;
- 5) кпд5 – канал доставки содержимого соединений.

4.2. Канал управления (кпд1) служит для передачи:

- 1) от ПУ в ИС БД ОРМ запросов (команд);
- 2) от ИС БД ОРМ на ПУ ответов и «сигналов».

4.3. Канал данных (кпд2) служит для передачи:

- 1) от ИС БД ОРМ на ПУ блоков данных отчетов, генерируемых ИС БД ОРМ в качестве ответов на запросы от ПУ, «сигнала» heartbeat;
- 2) от ПУ в ИС БД ОРМ подтверждений о принятии блоков данных отчетов.

4.4. Канал мониторинга (кпд3) служит для передачи:

- 1) от ПУ в ИС БД ОРМ запросов о текущей конфигурации оборудования, системного и прикладного ПО ИС БД ОРМ и запросов на модификацию конфигурации;
- 2) от ИС БД ОРМ на ПУ ответов на запросы ПУ, «сигналов».

4.5. Канал неформатированных данных (кпд4) служит для передачи:

- 1) от ПУ в ИС БД ОРМ команд на виды передаваемых неформатированных данных;
- 2) от ИС БД ОРМ на ПУ неформатированных данных, «сигналов».

4.6. Канал доставки содержимого соединений (кпд5) служит для передачи:

- 1) от ИС БД ОРМ на ПУ блоков содержимого соединений запрошенных ПУ, сигналов;
- 2) от ПУ в ИС БД ОРМ подтверждений о принятии блоков содержимого соединений.

4.7. Каналы передачи данных (кпд1, кпд2, кпд3, кпд4, кпд5) представляют собой ТСР-соединения, создаваемые для подключения на заранее определенные порты оборудования ИС БД ОРМ. Номера портов передаются на ПУ на внешних носителях. ИС БД ОРМ выполняет прослушивание данных портов для создания ТСР-соединений с ПУ.

5. ПУ выполняет попытки установления подключения к ИС БД ОРМ в соответствии с задаваемым интервалом по предоставленным ИС БД ОРМ ТСР-портам.

5.1. Установление ПУ соединений к ИС БД ОРМ по каналам кпд1, кпд2 осуществляется в следующей последовательности:

- 1) ПУ устанавливает ТСР-соединение к ИС БД ОРМ по порту канала кпд1;
- 2) выполняется процедура взаимной SSL/TLS аутентификации в соответствии с пунктом 15 Приложения № 2 к Правилам;

3) ПУ устанавливает TCP-соединение к ИС БД ОРМ по порту канала кпд2;

4) выполняется процедура взаимной SSL/TLS аутентификации в соответствии с пунктом 15 Приложения № 1 к Правилам;

5) после успешной аутентификации ПУ выполняет создание сессии к ИС БД ОРМ;

6) ПУ ожидает данных и сигналов по кпд2 только после того, как была создана сессия по каналу кпд1. При приеме данных и сигналов по кпд2 при отсутствии установленной сессии по кпд1 ПУ разрывает соединение по кпд2.

5.2. Установление ПУ соединений к ИС БД ОРМ по каналам кпд3, кпд4 и кпд5 осуществляется независимо друг от друга и не зависит от наличия установленных соединений по каналам кпд1 и кпд2:

1) ПУ устанавливает TCP-соединение к ИС БД ОРМ по TCP-порту канала кпд3/кпд4/кпд5;

2) выполняется процедура взаимной SSL/TLS аутентификации в соответствии с пунктом 15 Приложения № 2 к Правилам;

3) после успешной аутентификации ПУ выполняет создание сессии к ИС БД ОРМ;

4) после успешной аутентификации ПУ посылает на ИС БД ОРМ команды в соответствии с Приложением № 8 к Правилам.

6. ПУ разрывает соединения к ИС БД ОРМ, если в течение трех периодов приема сигнала HeartBeat в каналах не было сетевой активности. При отсутствии подтверждения посланного сигнала HeartBeat в течении времени «максимальный размер задержки подтверждения запроса или сигнала» ИС БД ОРМ разрывает соединения с ПУ по соответствующему каналу.

7. ИС БД ОРМ обеспечивает одновременное подключение нескольких ПУ. Предоставление ПУ доступа к информации ИС БД ОРМ по территориям оказания услуг данным узлом связи. Максимальное количество одновременно подключенных ПУ для одной ИС БД ОРМ – 100. ИС БД ОРМ обслуживает подключенные ПУ независимо друг от друга.

8. Каждому идентифицированному на ИС БД ОРМ ПУ назначается приоритет выполнения задач. В случае поступления ИС БД ОРМ задач от различных ПУ, вероятность постановки на выполнение задачи конкретного пункта управления ОРМ зависит от назначенного приоритета данного ПУ и приоритетов других ПУ. Распределение вероятности запуска задач от различных ПУ задается приоритетом каждого конкретного ПУ. Конфигурация по умолчанию обеспечивает равномерное распределение вероятности запуска задач от каждого ПУ. ИС БД ОРМ обеспечивает возможность конфигурирования приоритетов ПУ. ИС БД ОРМ обеспечивает одновременное выполнение задач по каждому идентифицированному ПУ:

1) по соединениям абонентов, по совершенным платежам, по пополнению справочников, по принадлежности абонентов – не менее 100 в целом;

2) по запросам содержимого соединений - не менее 10.

9. Единицей обмена в кпд1, кпд2, кпд3, кпд4 и кпд5 является «Сообщение» (Message). Форматы «Сообщений» представлены в Приложении №

8 к Правилам на языке абстрактного описания синтаксиса (ASN.1) согласно ГОСТ Р ИСО/МЭК 8824-1-2001. Способ кодирования сериализованных «Сообщений» соответствует отличительным (DER) по ГОСТ Р ИСО/МЭК 8825-1-2003.

10. Интерфейс взаимодействия между ПУ и ИС БД ОРМ по каналам кпд1 и кпд2 предусматривает наличие следующих видов «Сообщений»:

- 1) «запросы» – передаются от ПУ в ИС БД ОРМ по кпд1;
- 2) «ответы» – передаются из ИС БД ОРМ на ПУ по кпд1;
- 3) «сигналы» – передаются из ИС БД ОРМ на ПУ по кпд1 и кпд2 (для кпд2 только «сигнал» Heartbeat);
- 4) «отчеты» - формируются ИС БД ОРМ в качестве ответов на запросы от ПУ, передаются на ПУ по кпд2;
- 5) «подтверждения» о принятии «отчетов» – передаются из ПУ в ИС БД ОРМ по кпд2;
- 6) «подтверждения» о принятии «сигналов» - передаются от ПУ в ИС БД ОРМ по кпд1 и кпд2 (для кпд2 только для «сигнала» Heartbeat).

11. Интерфейс взаимодействия между ПУ и ИС БД ОРМ по каналу кпд3 предусматривает наличие следующих видов «Сообщений»:

- 1) запросы;
- 2) ответы;
- 3) сигнал Heartbeat;
- 4) подтверждения о принятии сигнала Heartbeat и ответов.

12. Интерфейс взаимодействия между ПУ и ИС БД ОРМ по каналу кпд4 предусматривает наличие следующих видов «Сообщений»:

- 1) «запросы» – передаются от ПУ в ИС БД ОРМ;
- 2) «ответы» – передаются из ИС БД ОРМ на ПУ;
- 3) «сигналы» – передаются из ИС БД ОРМ на ПУ;
- 4) «отчеты» - формируются ИС БД ОРМ в качестве ответов на запросы от ПУ;
- 5) «подтверждения» о принятии «отчетов» – передаются от ПУ в ИС БД ОРМ;
- 6) «подтверждения» о принятии «сигналов» - передаются от ПУ в ИС БД ОРМ.

13. Интерфейс взаимодействия между ПУ и ИС БД ОРМ по каналу кпд5 предусматривает наличие следующих видов «Сообщений»:

- 1) «запросы» – передаются от ПУ в ИС БД ОРМ, используются только запросы создания сессии;
- 2) «ответы» – передаются из ИС БД ОРМ на ПУ;
- 3) «сигналы» – передаются из ИС БД ОРМ на ПУ;
- 4) «отчеты» - формируются ИС БД ОРМ в качестве ответов на запросы от ПУ;
- 5) «подтверждения» о принятии «отчетов» – передаются от ПУ в ИС БД ОРМ;
- 6) «подтверждения» о принятии «сигналов» - передаются от ПУ в ИС БД ОРМ.

14. ИС БД ОРМ выполняет любые действия, связанные с выдачей информации о пользователях услуг связи и предоставленных им услугах связи, управлением и мониторингом технических средств и программным обеспечением ИС БД ОРМ только по «запросам» ПУ.

14.1. ИС БД ОРМ обеспечивает прием и обработку следующих «запросов», передаваемых с ПУ по кпд1:

- 1) «Запрос на открытие сессии» (ConnectRequest);
- 2) «Запрос на закрытие сессии» (DisconnectRequest);
- 3) «Запрос готовности данных» (DataReadyRequest);
- 4) «Запрос загрузки данных» (DataLoadRequest);
- 5) «Запрос удаления данных» (DataDropRequest);
- 6) «Запрос прерывания загрузки данных» (DataInterruptRequest);
- 7) «Запрос на создание задачи по обработке информации» (CreateTaskRequest);
- 8) «Запрос на создание задачи по обработке неформализованных данных» (NonFormalizedTaskRequest);
- 9) «Запрос на создание правила фильтрации содержимого соединений сети передачи данных» (CreateFilterRequest);
- 10) «Запрос на удаление правила фильтрации содержимого соединений сети передачи» (DropFilterRequest).

14.2. В зависимости от типа завершенной поисковой задачи передача блоков данных осуществляется по каналу кпд2 или кпд5:

14.2.1. В случае, если запрошенная задача является задачей получения содержимого соединений, используется канал кпд5;

14.2.2. Во всех остальных случаях используется канал кпд2.

14.3. ИС БД ОРМ обеспечивает одновременную передачу блоков данных отчетов для нескольких завершенных поисковых задач по каналам кпд2/кпд5. ИС БД ОРМ обеспечивает возможность многократной передачи отчетов выполненных задач на ПУ.

14.4. По запросу ПУ «Запрос загрузки данных» канала кпд1 ИС БД ОРМ обеспечивает передачу блоков отчетов по каналам кпд2/кпд5 по запрошенной задаче без внесения дополнительных временных задержек между операциями по получению записей результата задачи и преобразования их в блоки.

14.5. На каждый «запрос» по кпд1 ИС БД ОРМ на ПУ посылает «ответ» о принятии к обработке этого запроса. ИС БД ОРМ обеспечивает посылку по кпд1 на ПУ следующих «ответов»:

- 1) «Ответ на запрос открытия сессии» (ConnectResponse);
- 2) «Ответ на запрос закрытия сессии» (DisconnectResponse);
- 3) «Ответ на запрос готовности данных» (DataReadyResponse);
- 4) «Ответ на запрос загрузки данных» (DataLoadResponse);
- 5) «Ответ на запрос удаления данных» (DataDropResponse);
- 6) «Ответ на запрос прерывания загрузки данных» (DataInterruptResponse);
- 7) «Ответ на запрос создания задачи» (TaskResponse);
- 8) «Ответ на запрос создания задачи по обработке неформализованных данных» (NonFormalizedTaskResponse);

9) «Ответ на запрос создания правила фильтрации содержимого соединений сети передачи данных» (CreateFilterResponse);

10) «Ответ на запрос удаления создание правила фильтрации содержимого соединений сети передачи данных» (DropFilterResponse).

14.6. По «Запросу на создание задачи по обработке информации» ИС БД ОРМ обеспечивает подготовку и выдачу информации из ИС БД ОРМ, для следующих групп задач:

1) «Задачи пополнения справочников (нормативно-справочная информация)» (DictionaryTask);

2) «Задачи поисков по принадлежности абонентов» (AbonentsTask);

3) «Задачи поисков по соединениям абонентов» (ConnectionsTask);

4) «Задачи получения данных местоположения абонентов» (LocationTask);

5) «Задачи поисков по совершенным платежам» (PaymentsTask);

6) «Задачи предоставления сведений о наличии данных» (PresenseTask);

7) «Задача получения содержимого соединений» (DataContentTask).

14.6.1. В группу «Задачи пополнения справочников (нормативно-справочная информация)» входят запросы на получение информации справочников:

1) пучки соединительных линий (идентификатор направления/оконечного оборудования вызываемой/вызывающей стороны);

2) базовые станции;

3) роуминговые партнеры;

4) коммутаторы;

5) IP шлюзы;

6) типы вызовов;

7) список услуг связи;

8) способы оплаты (пополнения баланса);

9) причины завершения соединения;

10) справочник с планами IP-адресации;

11) план телефонной номерной емкости;

12) типы документов, удостоверяющих личность;

13) операторы связи, обслуживаемые ИС БД ОРМ;

14) идентификаторы точек подключения к сети передачи данных, с которых получены записи о соединениях;

15) специальные номера оператора связи;

16) карта связей пучков соединительных линий;

17) план нумерации идентификаторов мобильных телефонных абонентов.

14.6.2. В группу «Задачи поисков по принадлежности абонентов» входят:

1) «Задача на поиск информации о принадлежности идентификаторов абонентов сети оператора связи» (ValidateAbonentsTask);

2) «Задача на поиск информации об идентификаторах абонентов сети оператора связи зарегистрированных на физическое или юридическое лицо» (ValidateIdentifiersTask);

3) «Задача на поиск информации о доступных абоненту видам услуг связи» (ValidateServicesTask).

14.6.2.1. По задачам поиска принадлежности идентификаторов абонента и об идентификаторах абонентов, зарегистрированных на физическое или юридическое лицо, ИС БД ОРМ вместе с информацией об абоненте и зарегистрированных идентификаторах предоставляет сведения о текущих подключенных абонентом услугах связи.

14.6.2.2. По «Задаче на поиск информации о доступных абоненту видах услуг связи» ИС БД ОРМ формирует отчет, содержащий список с историей подключенных абонентом услуг связи за весь период накопления данных. Поисковым критерием задачи является идентификатор абонента, не содержащий символов маскирования.

14.6.3. В группу «Задачи поисков по соединениям абонентов» входит «Задача на поиск по соединениям абонентов» (ValidateCallsTask), в том числе:

- а) «Задача на поиск соединений абонентов» (ValidateConnectionsTask);
- б) «Задача на поиск соединений между абонентами сети передачи данных» (ValidateDataTask);
- в) «Задача на поиск информации о появлении абонента в сети связи (выхода на связь)» (ValidateEntranceTask).

14.6.3.1. В случае если в качестве параметров «задачи на поиск соединений абонентов» указаны только диапазон времени и (опционально) филиал оператора связи, ИС БД ОРМ формирует результат выполнения поисковой задачи, содержащий все соединения всех абонентов за указанный диапазон времени.

14.6.4. В группу «Задачи поисков по совершенным платежам» входят:

- 1) «Задача на поиск пополнения баланса через банковский перевод» (bankTransactionTask);
- 2) «Задача на поиск пополнения баланса через карты экспресс-оплаты» (expressCardTask);
- 3) «Задача на поиск пополнения баланса через терминалы моментальных платежей» (publicTerminalTask);
- 4) «Задача на поиск пополнения баланса через центры обслуживания клиентов (ЦОК)» (serviceCenterTask);
- 5) «Задача на поиск пополнения баланса посредством снятия денег со счета другого абонента» (crossAccountTask);
- 6) «Задача на поиск пополнения баланса через телефонные карты» (telephoneCardTask);
- 7) «Задача на поиск перевода средств со счета абонента для их снятия в отделении банка» (bankDivisionTransferTask);
- 8) «Задача на поиск перевода средств со счета абонента на банковскую карту» (bankCardTransferTask);
- 9) «Задача на поиск перевода средств со счета абонента на счет в банке» (bankAccountTransferTask);
- 10) «Общая задача на поиск пополнения баланса личного счета абонента» (balanceFillupTask).

14.6.4.1. По «Задаче получения данных местоположения абонентов» ИС БД ОРМ за запрошенный период времени предоставляет накопленную в базе данных

информацию об изменении местоположения абонентского устройства (историю) по запрошенному идентификатору.

14.6.5. В группу «Задачи предоставления сведений о наличии данных» (PresenseTask) входят:

- 1) запрос наличия информации по абонентам и их идентификаторам;
- 2) запрос наличия информации по соединениям;
- 3) запрос наличия имеющейся информации по платежам;
- 4) запрос наличия справочников;
- 5) запрос наличия информации по местоположениям абонентов.

14.6.6. По «Задаче получения содержимого соединений» ИС БД ОРМ передает по каналу кпд5 соответствующее содержимое соединения по запрошенному идентификатору.

14.7. По «Запросу на создание задачи по обработке неформализованных данных» ИС БД ОРМ обеспечивает подготовку и выдачу информации по следующим видам запросов:

- 1) «запрос получения списка типов сущностей» неформализованных данных ИС БД ОРМ (GetEntities);
- 2) «запрос получения списка атрибутов сущности» (GetEntityAtttributes);
- 3) «задача поиска неформализованных данных» (ValidateNonFormalizedTask);
- 4) «задача предоставления сведений о наличии неформализованных данных» (NonFormalizedPresenseTask).

14.8. ИС БД ОРМ по «задаче поиска неформализованных данных» предоставляет доступ ПУ к системным журналам ИС БД ОРМ.

14.9. ИС БД ОРМ обеспечивает выполнение поисковых задач по строковым критериям, содержащими символы маскирования, включающие:

- 1) «\*» – обозначает любую комбинацию символов;
- 2) «?» – обозначает любой один возможный символ.

14.9.1. ИС БД ОРМ обеспечивает выполнение поисковых задач по критериям, содержащим последовательность цифр с символом маскирования пробел (« »), означающим любую одну цифру.

Результат выполнения поисковой задачи с критерием, содержащим символы маскирования, содержит все записи, соответствующие заданной маске.

14.10. В случае возникновения в ИС БД ОРМ исключительных ситуаций на ПУ передаются следующие «Сообщения» типа «Сигналы», содержащие информацию об уровне важности исключительной ситуации, ее влиянии на сохранность данных и выполнение задач:

- 1) «Перезапуск ПО» (RestartDB);
- 2) «Попытка несанкционированного доступа» (UnauthorizedAccess);
- 3) «Критическая ошибка ПО, потеря данных, дальнейшая работа невозможна» (CriticalError);
- 4) «Серьезная ошибка ПО, потеря данных, но дальнейшая работа возможна» (MajorError);
- 5) «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» (MinorError);

б) «Тестовый пакет» (Heartbeat). Единственный, из «сигналов», передающийся по кпд1, кпд2, кпд3, кпд4 в отсутствие иной сетевой активности.

14.10.1. В ответ на «сигналы», поступившие от ИС БД ОРМ, ПУ передает «подтверждения сигнала» об их приеме.

14.11. При получении «Запроса на создание правила фильтрации содержимого соединений» ИС БД ОРМ добавляет заданное правило к списку правил, используемых для фильтрации соединений в сети передачи данных.

14.12. При получении «Запроса на удаление правила фильтрации содержимого потока» ИС БД ОРМ исключает заданное правило из списка правил, используемых для фильтрации соединений в сети передачи данных.

14.13. ИС БД ОРМ обеспечивает запись содержимого соединений сети передачи данных в соответствии с заданными правилами фильтрации:

1) Для каждого соединения проводится проверка соответствия одному из заданных правил фильтрации;

2) При обнаружении соответствия соединению присваивается уникальный идентификатор в рамках данной ИС БД ОРМ;

3) Содержимое соединения записывается в ИС БД ОРМ;

4) Идентификатор используется в «Задаче получения содержимого потоков»;

5) Статистическая информация о соединении по его завершению записывается в ИС БД ОРМ вместе с идентификатором соединения.

14.14. ИС БД ОРМ по «Запросу удаления данных» (DataDropRequest) осуществляет:

1) прерывание задачи, находящейся на выполнении;

2) удаление данных отчета по завершившейся задаче.

15. При установлении соединения ПУ и ИС БД ОРМ взаимно аутентифицируются. Аутентификация выполняется установлением SSL/TLS-соединения поверх установленного TCP-соединения между ПУ и ИС БД ОРМ. Для взаимной аутентификации ПУ и ИС БД ОРМ предварительно создаются X.509-сертификаты, которые сообщаются ПУ и ИС БД ОРМ. В случае невозможности аутентифицировать одну из сторон TCP-соединение разрывается. Созданный для ПУ сертификат используется для аутентификации только одного данного ПУ на одной ИС БД ОРМ по всем каналам передачи данных — кпд1, кпд2, кпд3, кпд4, кпд5. ПУ и ИС БД ОРМ используют TLS версии 1.2. Требования к сертификатам (длины ключей, прочие параметры) согласовываются для каждой пары ИС БД ОРМ и ПУ отдельно.

16. Открытие сессии осуществляется выполнением процедуры аутентификации согласно пункту 15 Приложения № 2 к Правилам перед началом выполнения всех запросов. Открытие сессии осуществляется посылкой по каналу управления от ПУ к ИС БД ОРМ «Запроса на открытие сессии» (ConnectRequest).

17. «Запрос на открытие сессии» устанавливает следующие параметры сессии:

- 1) «максимальное время отсутствия активности сессии» (session-timeout) – интервал времени, по истечении которого сессия принудительно прерывается от ИС БД ОРМ;
- 2) «максимальный размер блока данных отчетов» (max-data-length) в строках записей ИС БД ОРМ;
- 3) «размер окна для канала передачи данных» (data-packet-window-size);
- 4) «максимальная длительность задержки начала передачи блоков отчетов» (data-load-timeout);
- 5) «максимальный размер задержки подтверждения о получении данных» (data-packet-response-timeout);
- 6) «максимальный размер задержки подтверждения запроса или сигнала» (request-response-timeout).

Любое сообщение в соответствии с ASN.1-протоколом взаимодействия ИС БД ОРМ и ПУ, согласно Приложению № 9 к Правилам считается сетевой активностью.

17.1. ИС БД ОРМ при получении сообщения «Запрос на открытие сессии», анализирует параметр «размер окна для канала передачи данных», определяет максимально возможный размер окна, не превышающий полученного от ПУ. ИС БД ОРМ определяет минимальные значения таймаутов из параметров сессии, выбирая их не меньше, чем переданные в сообщении «Запрос на открытие сессии» (ConnectRequest). Рассчитанные значения размеров окна и таймаутов ИС БД ОРМ передает ПУ в сообщении «Ответ на открытие сессии» (ConnectResponse). Полученные ПУ значения в сообщении «Ответ на открытие сессии» являются параметрами сессии между ПУ и ИС БД ОРМ.

18. Закрытие сессии осуществляется посылкой по каналу управления, каналу мониторинга, каналу неформатированных данных или каналу доставки содержимого соединений от ПУ к ИС БД ОРМ «Запроса на закрытие сессии» или по истечению допустимого времени отсутствия активности ИС БД ОРМ, с посылкой сообщения-сигнала «Прерывание текущей сессии по таймауту». При этом ИС БД ОРМ и ПУ осуществляют разрыв текущих TCP соединений канала управления и канала данных, канала мониторинга, канала неформатированных данных или канала доставки содержимого соединений.

19. ПУ посылает ИС БД ОРМ «запросы» асинхронно, независимо от получения от ИС БД ОРМ «ответа» о приеме предыдущего «запроса».

20. «Запрос на создание задачи по обработке информации» приводит к созданию в ИС БД ОРМ задачи по обработке данных в базе данных ИС БД ОРМ, которой присваивается номер (идентификатор) задачи, передаваемый в «Ответ на запрос создания задачи» (TaskResponse). Идентификаторы задач генерируются ИС БД ОРМ независимо от сессий и являются уникальными в данной ИС БД ОРМ. ИС БД ОРМ присваивает идентификаторы задачам и выполняет обработку задач независимо для различных пунктов управления ОРМ.

21. ПУ получает информацию о ходе выполнения и завершения обработки задач в ИС БД ОРМ, посылая запрос «Запрос готовности данных». После завершения выполнения задачи, данные, сформированные в результате выполнения задачи, становятся доступными для загрузки в ПУ или для удаления.

22. ИС БД ОРМ при получении «Запрос загрузки данных» по кпд1 формирует сообщения типа - «отчет», состоящие из блоков данных обработанной задачи.

23. При передаче блоков данных по задаче получения содержимого соединений используется канал кпд5.

24. При передаче блоков данных по всем остальным задачам используется канал кпд2.

25. При получении "Запроса загрузки данных" для задачи получения содержимого соединений при отсутствии установленной сессии по каналу кпд5, ИС БД ОРМ посылает ответ на "Запрос загрузки данных" с установленным флагом ошибки. При этом в случае установления сессии по каналу кпд5 передача ранее запрошенных отчетов не производится.

26. Количество строк в каждом блоке при передаче по каналу кпд2 не превышает параметр «максимальный размер блока данных отчетов», заданный при открытии сессии.

27. Количество байт в каждом блоке при передаче по каналу кпд5 устанавливается при конфигурации ИС БД ОРМ.

28. В каждом последовательном блоке данных из серии указываются идентификатор задачи, сгенерировавшей данный отчет, общее количество блоков в отчете, порядковый номер каждого блока.

29. Данные задачи, полученные в одной сессии, могут быть считаны и/или удалены в другой сессии ПУ, инициировавшим данную задачу. Данные по завершенной задаче доступны между сессиями по тому же идентификатору задачи. При получении задачи «запрос загрузки данных» (DataLoadRequest) от ПУ, не являющегося инициатором данной задачи, ИС БД ОРМ посылает ответ на «запрос загрузки данных», в котором указывается отсутствие результата исполнения задачи (data-exists), а в поле «краткое описание ошибки» (error-description) записывается расшифровка отказа в доступе к данным задачи. Далее ИС БД ОРМ посылает на этот ПУ «сигнал» попытки несанкционированного доступа (unauthorized-access) и ожидает его подтверждения.

30. ИС БД ОРМ производит уничтожение данных, сформированных в результате выполнения задачи и самой выполненной задачи после поступления с ПУ запроса на удаление данных.

31. В случае возникновения в ИС БД ОРМ или каналах передачи данных исключительных ситуаций на ПУ передаются «Сообщения» типа «Сигнал».

---



1.1. ИС БД ОРМ по TCP-порту кпд1 ожидает входящих соединений. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

1.2. Если SSL/TLS-соединения по каналам управления и данных установлены, а сессия не открыта, ИС БД ОРМ реагирует только на сообщение «Запрос на открытие сессии». При попытке посылок каких-либо других сообщений со стороны ПУ ИС БД ОРМ разрывает TCP соединения по каналу управления и каналу данных и переводит канальный уровень подключения в исходное состояние.

1.3. При получении сообщения «Запрос на открытие сессии» ИС БД ОРМ создает список поддерживаемых ИС БД ОРМ типов запросов, отчетов и сигналов (в том числе и предыдущих версий) и отправляет его.

1.4. После отсылки списка поддерживаемых типов, ИС БД ОРМ ожидает от ПУ списка поддерживаемых им запросов, отчетов и сигналов. Список поддерживаемых ПУ типов является подмножеством списка типов в ИС БД ОРМ.

1.5. При получении от ПУ списка поддерживаемых ПУ запросов ИС БД ОРМ посылает сообщение «Ответ на согласование списка поддерживаемых типов» и создает сессию.

1.6. После создания сессии кпд1 ИС БД ОРМ переводится в режим ожидания команд от ПУ. При поступлении команды со стороны ПУ производится ее выполнение и формируется результат. Результат в виде сообщения «ответа» отправляется на ПУ.

1.7. В случае серьезного сбоя в ИС БД ОРМ, вызванного причинами, не предусмотренными режимом нормального функционирования системы, по каналу управления передается «сигнал» - «Критическая ошибка ПО, потеря данных, дальнейшая работа невозможна», с соответствующим описанием проблемы. Все задачи, которые были в процессе выполнения, когда произошел сбой, а также данные выполненных задач, поврежденные в результате произошедшего сбоя, имеют «признак результата выполнения задачи» (TaskResult) равный значению «ошибка» (error), с соответствующим описанием проблемы. В случае, если для восстановления работоспособности ИС БД ОРМ требуется ее перезагрузка, то по каналу управления выдается прерывание «Перезапуск ПО». В этом случае ИС БД ОРМ и ПУ закрывают все открытые на текущий момент сессии.

1.8. В случае наличия признаков сбоя или ошибки выполнения конкретной задачи ИС БД ОРМ в режиме нормального функционирования, по каналу управления передает прерывание «Серьезная ошибка ПО, потеря данных, но дальнейшая работа возможна», с соответствующим описанием проблемы. Результат выполнения данной задачи имеет «признак результата выполнения задачи» (TaskResult) равный значению «ошибка» (error), с соответствующим описанием проблемы.

1.9. На каждый «сигнал», переданный ИС БД ОРМ, ПУ отвечает сообщением «подтверждение сигнала» по кпд1. Отсутствие подтверждения в течение времени RequestResponseTimeout, которое задается при открытии сессии, свидетельствует о прерывании соединения и вызывает действия, согласно пункту 18 Приложения № 2 к Правилам.

1.10. При отсутствии команд ПУ в течение «максимального времени неактивности» (session-timeout, задается в ConnectRequest), ИС БД ОРМ посылает на ПУ «сигнал» heartbeat и ожидает его подтверждения аналогично описанному в пункте 1.9 Приложения № 3 к Правилам.

2. Диаграмма состояний переходов ПУ по кпд1 приведена на Рисунке 2.

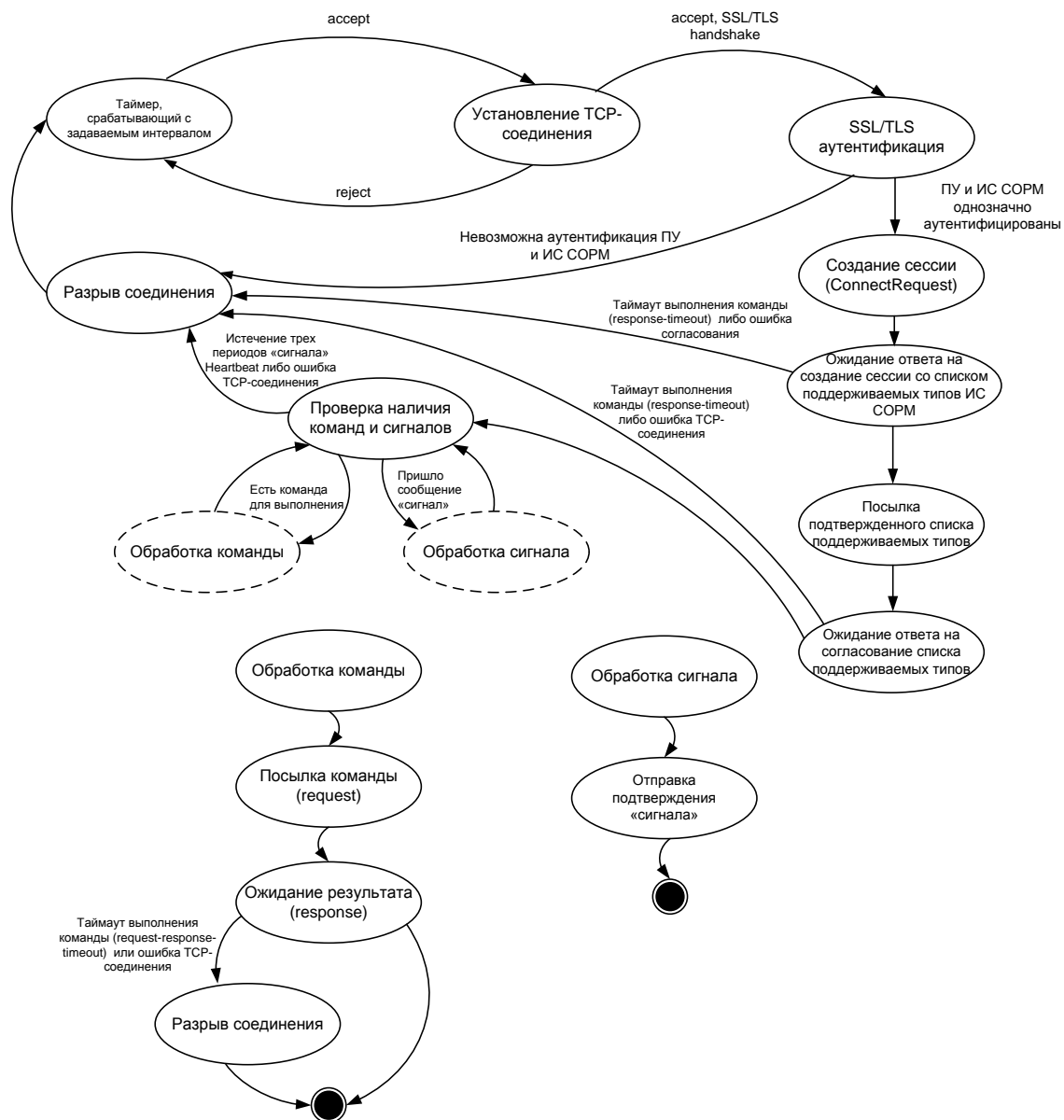


Рисунок 2. Диаграмма состояний перехода ПУ по кпд1

2.1. ПУ с задаваемым интервалом выполняет попытки установления TCP-соединения к ИС БД ОРМ по заданному порту кпд1. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

2.2. ПУ ожидает установления соединения по кпд2.

2.3. После установления TCP-соединения по кпд2 и прохождения по нему взаимной аутентификации ИС БД ОРМ и ПУ, ПУ по кпд1 посылает команду создания сессии (ConnectRequest).

2.4. ПУ ожидает сообщения «ответ» от ИС БД ОРМ на отправленную команду в течение времени «таймаут ответа на запрос» (request-response-timeout).

2.5. Если сообщение не получено в течение времени «таймаут ответа на запрос», ПУ разрывает соединения к ИС БД ОРМ по кпд1 и кпд2 и переводит их в изначальное состояние согласно пункту 2.1 Приложения № 3 к Правилам. Время ожидания сообщения «ответ» не зависит от приема сообщений «сигналов», поступающих в этот интервал времени.

2.6. При получении сообщения «Ответ на запрос создания сессии» пункт управления ОРМ создает список поддерживаемых ПУ типов запросов, отчетов и сигналов и отправляет его ИС БД ОРМ. Список поддерживаемых ПУ типов является подмножеством списка типов ИС БД ОРМ согласно пункту 2.4 Приложения № 3 к Правилам.

2.7. После отправки сообщения «Согласование поддерживаемых типов со стороны ПУ» ПУ ждет ответа на сообщение от ИС БД ОРМ. Поведение ПУ при ожидании ответа согласно пункту 2.5 Приложения № 3 к Правилам.

2.8. Если во время ожидания сообщения «ответ» ИС БД ОРМ посылает на ПУ сообщение «сигнал» (в т.ч. heartbeat), то ПУ посылает «подтверждение» о принятии «сигнала» (в т.ч. heartbeat) и продолжает ожидать сообщение «ответ» на отосланную команду.

2.9. После создания сессии ПУ посылает поступающие команды на ИС БД ОРМ и ожидает сообщений «ответов» на них согласно пунктам 2.4, 2.5, 2.8 Приложения № 3 к Правилам.

2.10. Если при ожидании поступления в ПУ команд ИС БД ОРМ не посылала «сигнал» heartbeat в течение трех интервалов «максимального времени неактивности» (session-timeout, задается в ConnectRequest) ПУ разрывает соединения к ИС БД ОРМ по кпд1 и кпд2 и переводит их в изначальное состояние согласно пункту 2.1 Приложения № 3 к Правилам.

2.11. При поступлении команды «запрос на закрытие сессии» (DisconnectRequest), ПУ отсылает ее на ИС БД ОРМ, ожидает сообщения «ответ» согласно пунктам 2.4, 2.5, 2.8 Приложения № 3 к Правилам после чего разрывает соединение по кпд2 и кпд1, и переводит их в изначальное состояние согласно пункту 2.1 Приложения № 3 к Правилам.

---

## Приложение № 4

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий

### **Требования, предъявляемые к функционированию канала кпд2**

1. ИС БД ОРМ обеспечивает подключение ПУ и обработку поступающих запросов по каналу кпд2 в соответствии с приведенными диаграммами состояний переходов.

1.1. Диаграмма состояний переходов ИС БД ОРМ по кпд2 приведена на рисунке 1.

1.1.1. ИС БД ОРМ по TCP-порту кпд2 ожидает входящих соединений. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

1.1.2. Если на ИС БД ОРМ был передан «запрос загрузки данных» (DataLoadRequest), ИС БД ОРМ посылает «ответ на запрос загрузки данных» (DataLoadResponse) по каналу кпд1 и начинает передачу данных блоков отчетов по кпд2 при их наличии. ПУ получает блоки отчетов по кпд2 до получения «ответа на запрос загрузки данных» (DataLoadResponse) по кпд1.

1.1.3. Если количество переданных без получения «подтверждения» о принятии серии блоков «отчетов» по всем задачам, по которым выполняется загрузка на ПУ данных, меньше «окна канала передачи данных» (параметр data-packet-window-size в запросе создания сессии ConnectRequest), то ИС БД ОРМ выполняет подготовку новых блоков отчетов по загружаемым задачам и посылает их на ПУ. Количество подготовленных и переданных без получения «подтверждения» блоков не превышает размера окна канала передачи данных.

1.1.4. Максимальная задержка подтверждения приема блока данных со стороны ПУ не превышает параметр «таймаут подтверждения приема блока данных отчета» (data-packet-response-timeout), указываемый при создании сессии. Если задержка подтверждения превысила заданное значение, то оставшиеся для передачи блоки данных не отправляются, и по каналу управления передается «сигнал» «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» с соответствующим описанием проблемы, при этом в поле «reference-message» сообщения «сигнал» указывается идентификатор сообщения блока отчета, по которому не поступило подтверждение приема.

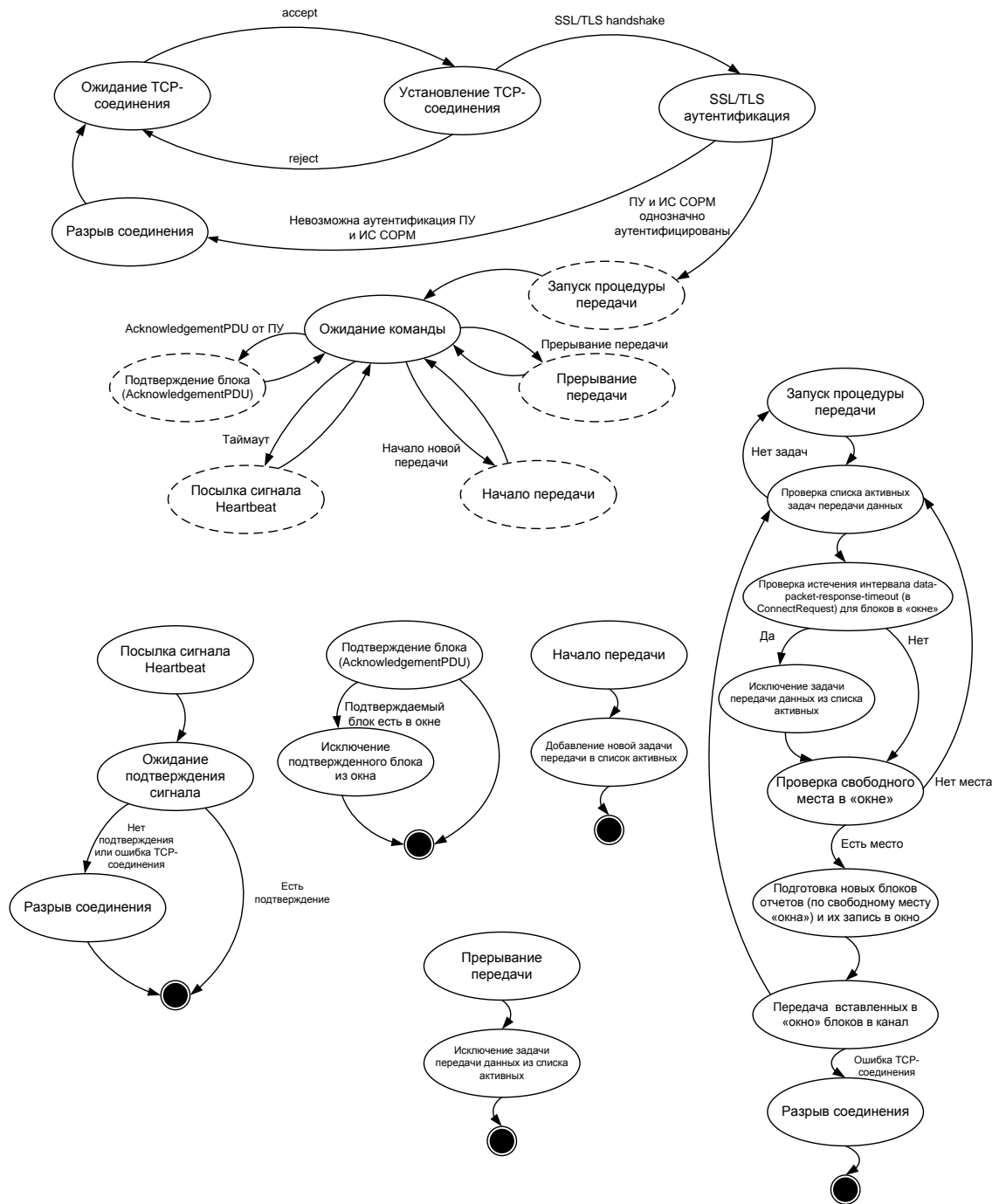


Рисунок 1. Диаграмма состояний перехода ИС БД ОРМ по кпд2

1.1.5. При получении «подтверждения» блока «отчета» ИС БД ОРМ записывает информацию об ошибочно принятом ПУ блоке и ошибочных записях в блоке в журнал, передача последующих блоков по задаче на ПУ не прерывается. ИС БД ОРМ предоставляет техническому персоналу оператора связи доступ к журналу с записями об ошибочно принятых на ПУ блоках отчетов и средства исправления ошибочных данных в отчетах. Подтвержденные блоки исключаются из окна канала передачи данных (в окне канала передачи данных остаются только неподтвержденные блоки).

1.1.6. В случае разрыва TCP/IP соединения кпд2, при существующем соединении кпд1, по кпд1 передаётся прерывание «Незначительная ошибка ПО,

данные не потеряны, дальнейшая работа возможна» с соответствующим описанием проблемы, работа в данном случае не прекращается, выполняется установление соединения по кпд2 согласно пункту 15 Приложения № 2 к Правилам.

1.1.7. Передача блоков данных прерывается в случае получения ИС БД ОРМ «запроса прерывания загрузки данных».

1.1.8. Если по кпд2 не производится передача блоков отчетов в течение «максимального времени неактивности» (session-timeout при создании сессии ConnectRequest), ИС БД ОРМ посылает на ПУ «сигнал» heartbeat и ожидает его подтверждения согласно пункту 1.9 Приложения № 3 к Правилам.

2. Диаграмма состояний переходов ПУ по кпд2 приведена на Рисунке 2.

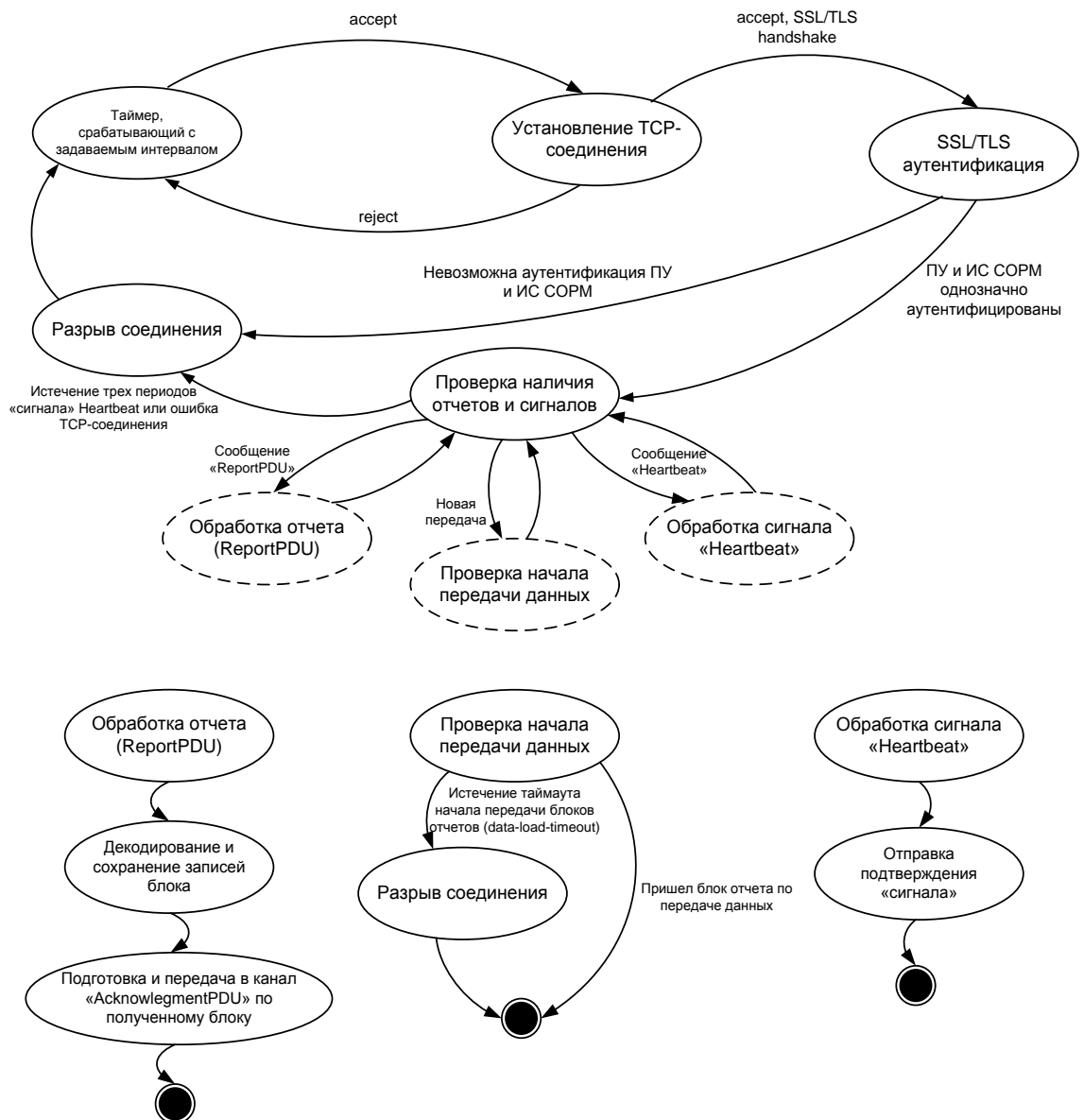


Рисунок 2. Диаграмма состояний перехода ПУ по кпд2

2.1. ПУ с задаваемым интервалом выполняет попытки установления TCP-соединения к ИС БД ОРМ по заданному порту кпд2. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

2.2. При поступлении «запроса загрузки данных конкретной задачи» (DataLoadRequest) ПУ ожидает начала передачи данных в течение времени «таймаут начала передачи блоков отчетов» (data-load-timeout в ConnectRequest). Если данные не поступают в течение вышеописанного периода, то ПУ разрывает соединения по каналам кпд1 и кпд2 и переводит соединения в изначальное состояние согласно пункту 2.1 Приложения № 3 к Правилам и пункту 2.1 Приложения № 4 к Правилам.

2.3. При поступлении блока отчета ПУ производит декодирование полученного блока и сохранение декодированных данных.

2.4. В ответ на переданный блок данных ПУ посылает сообщение «подтверждение» получения блока отчета. Количество последовательно переданных ИС БД ОРМ блоков данных без подтверждения со стороны ПУ определяется параметром «размер окна канала передачи данных», который согласовывается при создании сессии. При подтверждении блока отчета ПУ сигнализирует об ошибке декодирования блока. В этом случае в сообщении «подтверждение» приема для ошибочно декодированного блока ПУ, в случае возможности, указывает номер записи в блоке, начиная с которой декодирование не удалось.

2.5. Если при ожидании поступления в ПУ блоков отчетов ИС БД ОРМ не посылала «сигнал» heartbeat в течение трех интервалов «максимального времени неактивности» (session-timeout, задается в ConnectRequest) ПУ разрывает соединения к ИС БД ОРМ по кпд1 и кпд2 и переводит их в изначальное состояние согласно пункту 2.1 Приложения № 3 к Правилам и пункту 2.1 Приложения № 4 к Правилам.

---

## Приложение № 5

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий

### **Требования, предъявляемые к функционированию канала кпд3**

1. ИС БД ОРМ обеспечивают подключение ПУ и обработку поступающих запросов по каналу кпд3 (канал мониторинга).

1.1. ИС БД ОРМ обеспечивает прием и обработку следующих видов запросов от ПУ по каналу кпд3 (канал мониторинга):

1) запрос на получение структуры КТС ИС БД ОРМ и списка модулей прикладного ПО КТС ИС БД ОРМ («GetStructureRequest»);

2) запрос на получение конфигурации модуля ПО КТС ИС БД ОРМ («GetModuleConfigRequest»);

3) запрос на изменение конфигурации модуля ПО КТС ИС БД ОРМ («SetModuleConfigRequest»);

4) запрос на получение состояния модуля ПО КТС ИС БД ОРМ («CheckModuleRequest»).

1.2. На каждый запрос по кпд3 ИС БД ОРМ посылает ответ на, содержащий результат обработки соответствующего запроса - «ManagementResponse».

1.3. Диаграммы состояний перехода ИС БД ОРМ и ПУ по кпд3 соответствуют диаграммам для кпд1 (рисунок 1, рисунок 2).

1.3.1. ИС БД ОРМ по TCP-порту кпд3 ожидает входящих соединений. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

1.3.2. Если SSL/TLS-соединение по кпд3 установлено, а сессия не открыта, ИС БД ОРМ реагирует только на сообщение «Запрос на открытие сессии». Создание сессии аналогично, представленному в пунктах 1.3 - 1.5 Приложения № 3 к Правилам. При попытке посылок каких-либо других сообщений со стороны ПУ ИС БД ОРМ разрывает TCP соединение по кпд3 и переводит канальный уровень подключения в исходное состояние.

1.3.3. После создания сессии ИС БД ОРМ переводится в режим ожидания команд от ПУ. Обработка поступающих команд и посылка сигналов производится аналогично, представленному в пунктах 1.6-1.10 Приложения № 3 к Правилам.

1.3.4. ПУ с задаваемым интервалом выполняет попытки установления TCP-соединения к ИС БД ОРМ по заданному порту. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

1.3.5. ПУ по кпд3 посылает команду создания сессии (ConnectRequest).

1.3.6. Ожидание подтверждения создания сессии, согласование списка поддерживаемых типов, отправка команд, ожидание ответов и обработка полученных от ИС БД ОРМ сигналов по кпд3 производится ПУ согласно пунктам 2.4-2.10 Приложения № 3 к Правилам.

2. ИС БД ОРМ обеспечивает получение следующих видов информации о структуре и функционировании ИС БД ОРМ по запросу ПУ:

1) о структуре и составе КТС ИС БД ОРМ, составе и состоянии интерфейсов взаимодействия КТС ИС БД ОРМ;

2) об установленном на КТС ИС БД ОРМ общесистемном и программном обеспечении ИС БД ОРМ, перечне и состоянии программных модулей в составе программного обеспечения ИС БД ОРМ;

3) о точках подключения ИС БД ОРМ к сети оператора связи и интерфейсах ввода информации в ИС БД ОРМ.

2.1. В части структуры и состава КТС ИС БД ОРМ, состава и состоянии интерфейсов взаимодействия КТС ИС БД ОРМ, ИС БД ОРМ по запросу ПУ предоставляет следующую информацию:

1) перечень коммутационного и серверного оборудования, средств хранения данных с его идентификацией;

2) идентификацию интерфейсов подключения КТС ИС БД ОРМ друг к другу;

3) параметры для серверного оборудования (на момент формирования ПУ запроса):

а) общий и занятый объем оперативной памяти;

б) количество сетевых интерфейсов с их идентификацией, текущую нагрузку;

в) общее количество процессоров, текущую загрузку;

г) общий объем дискового пространства, объем свободного пространства;

4) параметры о технических средствах хранения данных:

а) перечень модулей, составляющих средства хранения данных с их идентификацией;

б) для каждого входящего в состав средств хранения данных модуля - общий объем дискового пространства, объем свободного дискового пространства и текущее состояние модуля (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя.

2.2. В части точек подключения ИС БД ОРМ к сети оператора связи, интерфейсов ввода информации в ИС БД ОРМ, ИС БД ОРМ по запросу ПУ предоставляет текущую информацию на момент формирования запроса, содержащую:

1) перечень точек подключения к сети связи и точек ввода информации в ИС БД ОРМ с их идентификацией;

2) для каждой точки подключения предоставляет информацию:

а) вид поступающих в ИС БД ОРМ сведений (о соединениях абонентах, о платежах, об абонентах и т.д.);

б) состояние точки (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

в) сведения об объеме поступающей информации в секунду, в том числе количество записей, объем (байт);

г) период, в течение которого на точку подключения/ввода информации в ИС БД ОРМ не поступала информация.

2.3. В части состава общесистемного и программного обеспечения ИС БД ОРМ, их текущего состояния, ИС БД ОРМ по запросу ПУ предоставляет следующую информацию:

1) перечень установленного общесистемного программного обеспечения с его идентификацией;

2) предоставление для общесистемного программного обеспечения информации:

а) идентификатора КТС ИС БД ОРМ, на котором установлено;

б) наименование общесистемного программного обеспечения;

в) текущее состояние (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

3) перечень установленного программного обеспечения ИС БД ОРМ с его идентификацией;

4) предоставление для программного обеспечения ИС БД ОРМ информации:

а) идентификатора КТС ИС БД ОРМ, на котором установлено;

б) назначение (определяется разработчиком ИС БД ОРМ);

в) текущее состояние (штатное функционирование, сбой, не функционирует), текстовую расшифровку сбоя;

г) список контролируемых параметров (определяется разработчиком ИС БД ОРМ).

2.4. ИС БД ОРМ предоставляет возможность изменения отдельных параметров функционирования КТС ИС БД ОРМ общесистемного и программного обеспечения по запросу ПУ посредством кпд3. Перечень доступных ПУ для изменения параметров определяется разработчиком ИС БД ОРМ.

---

## Приложение № 6

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий

### **Требования, предъявляемые к функционированию канала кпд4**

1. ИС БД ОРМ обеспечивает подключение ПУ и обработку поступающих запросов по каналу кпд4 (канал неформатированных данных).

2. Кпд4 является необязательным и реализуется ИС БД ОРМ по согласованию с ПУ. ИС БД ОРМ обеспечивает подключение ПУ и обработку поступающих запросов по каналу кпд4 (канал неформатированных данных).

2.1. ИС БД ОРМ обеспечивает прием и обработку следующих видов запросов от ПУ по каналу кпд4:

1) «запрос проверки наличия вида неформатированных данных в ИС БД ОРМ» (DataTypesRequest);

2) «запрос на начало передачи неформатированных данных» (DataStartRequest);

3) «запрос на остановку передачи неформатированных данных» (DataStopRequest).

2.2. На каждый запрос по кпд4 ИС БД ОРМ посылает ответ на ПУ, содержащий результат обработки соответствующего запроса.

2.3. ИС БД ОРМ накапливает информацию с неформатированными данными в буфере. Данные в буфере упорядочиваются согласно времени их поступления.

2.3.1. Для телефонных соединений абонентов (в соответствии с лицензиями на оказание услуг связи, перечисленных в пункте 5 Правил) длительность временного хранения неформатированных данных в буфере – до 6 месяцев.

2.4. ИС БД ОРМ записывает в буфер информацию о телефонных соединениях абонентов – файлы, формируемые коммутационным оборудованием, содержащие только записи о соединениях абонентов в не декодированном виде.

2.5. ИС БД ОРМ записывает информацию в буфер циклически, т.е. при переполнении буфера старая информация в нем перезаписывается.

2.6. Диаграмма состояний переходов ИС БД ОРМ по кпд4 приведена на Рисунке 1.

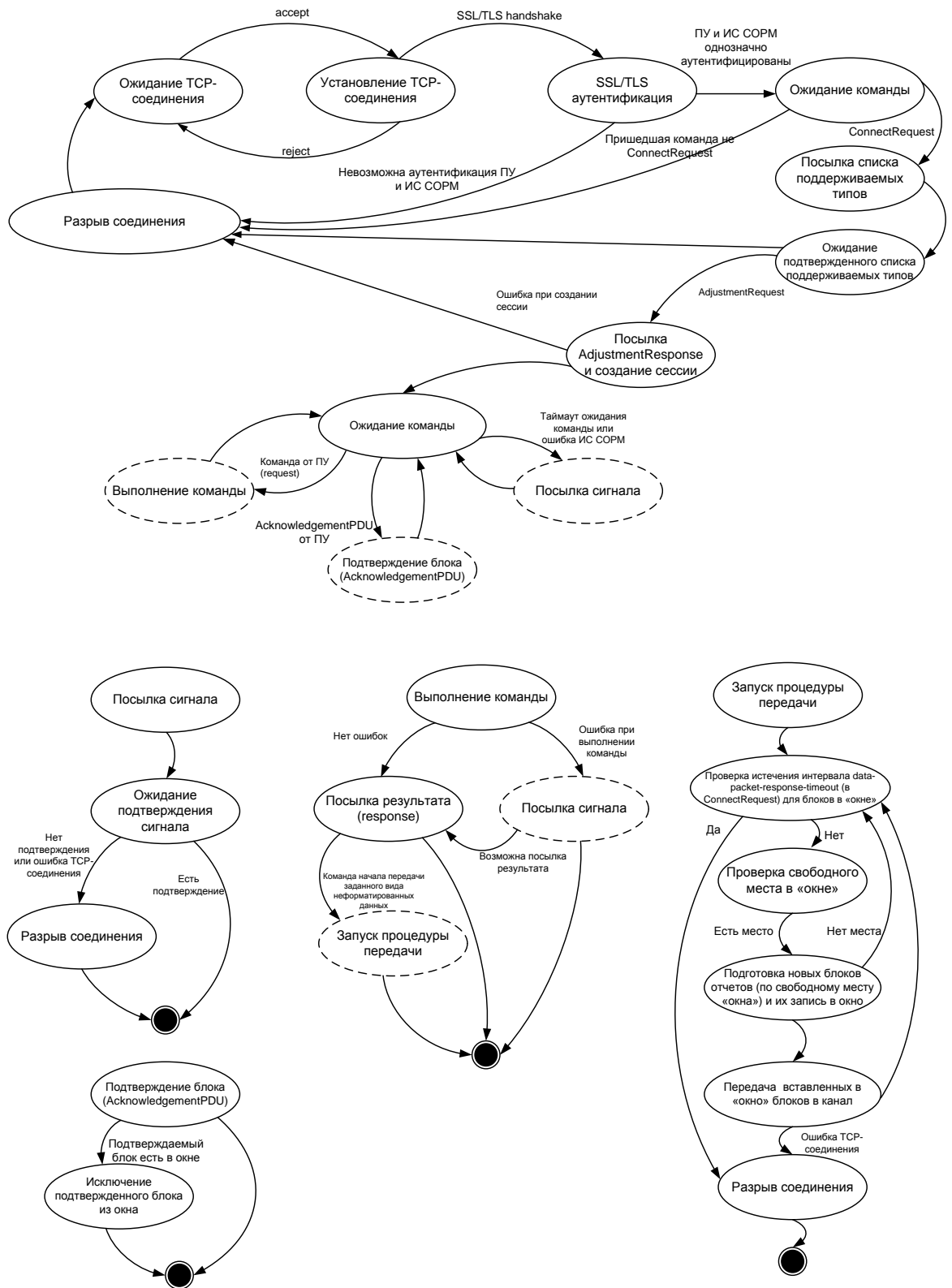


Рисунок 1. Диаграмма состояний перехода ИС БД ОРМ по кпд4

2.6.1. ИС БД ОРМ ожидает и устанавливает соединение согласно пунктам 1.3.1-1.3.2 Приложения № 5 к Правилам.

2.6.2. После создания сессии ИС БД ОРМ переводится в режим ожидания команд. Обработка команд и посылка «сигналов» осуществляется согласно пунктам 1.6-1.10 Приложения № 3 к Правилам, за исключением команд «запрос

на начало, остановку передачи неформатированных данных» (DataStartRequest/DataStopRequest).

2.6.3. При приеме команды «запрос на начало передачи неформатированных данных» ИС БД ОРМ, находясь в режиме ожидания команд, отправляет «ответ на запрос начала передачи неформатированных данных» и переводит канал кпд4 в режим передачи данных, при этом производится передача данных того типа, который указан в команде запроса.

2.6.4. При приеме команды «запрос на остановку передачи неформатированных данных» ИС БД ОРМ, находясь в режиме ожидания команд, посылает «ответ на запрос остановки передачи неформатированных данных» с отрицательным результатом.

2.6.5. В режиме передачи данных посылка блоков отчетов и их подтверждение производится согласно пунктам 1.1.3, 1.1.5 и 1.1.8 Приложения № 4 к Правилам.

2.6.6. Максимальная задержка подтверждения приема блока данных со стороны ПУ не превышает параметр «таймаут подтверждения приема блока данных отчета» (data-packet-response-timeout), указываемый при создании сессии. Если задержка подтверждения превысила заданное значение, то оставшиеся для передачи блоки данных не отправляются, ИС БД ОРМ разрывает соединение по кпд4 и переводит кпд4 в изначальное состояние.

2.6.7. Передача неформатированных данных соответствующего типа производится из буфера кпд4 согласно пункту 1.3 Приложения № 6 к Правилам.

2.6.8. При приеме команды «запрос на остановку передачи неформатированных данных» ИС БД ОРМ, находясь в режиме передачи данных, посылает «ответ на запрос остановки передачи неформатированных данных», завершает посылку блоков данных и переводится в режим ожидания команд.

2.6.9. При приеме команды «запрос на начало передачи неформатированных данных» ИС БД ОРМ, находясь в режиме передачи данных, посылает «ответ на запрос начала передачи неформатированных данных» с отрицательным результатом, при этом передача неформатированных данных не прекращается.

2.6.10. Исходные данные о соединениях в виде неформатированных данных записываются в буфер независимо от текущего режима работы канала кпд4 ИС БД ОРМ.

2.6.11. Если команда «запрос на остановку передачи неформатированных данных» поступила в момент передачи из буфера файловых данных, то передаваемый файл сохраняется в буфере и будет доступным для передачи после посылки команды «запрос на начало передачи неформатированных данных» с учетом ограничений по длительности хранения, согласно пункту 1.3 Приложения № 6 к Правилам.

2.7. Диаграмма состояний переходов ПУ по кпд4 представлена на Рисунке 2.

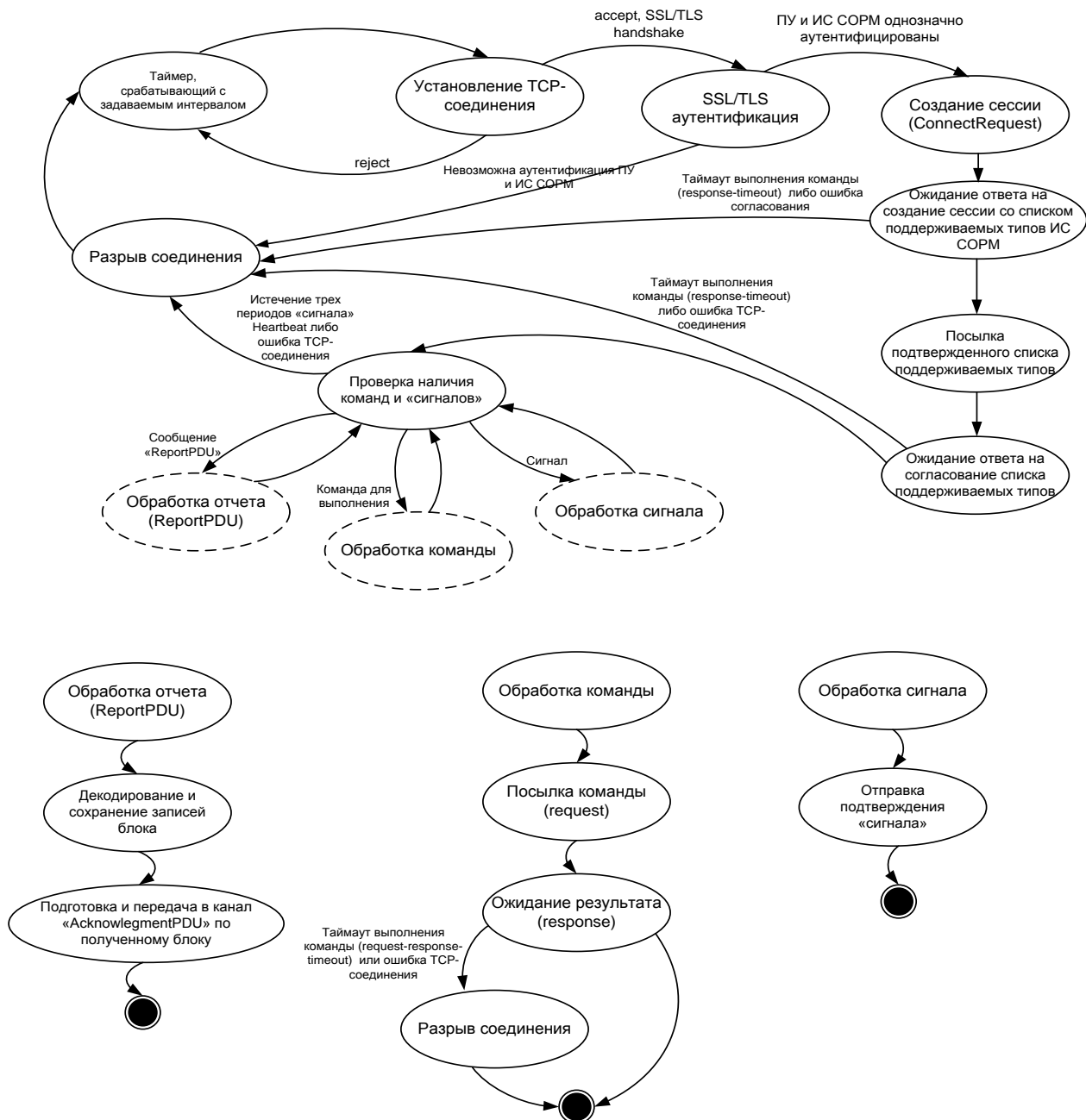


Рисунок 2. Диаграмма состояний перехода ПУ по кпд4

2.7.1. ПУ с задаваемым интервалом выполняет попытки установления TCP-соединения к ИС БД OPM по заданному порту. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

2.7.2. ПУ по кпд4 посылает команду создания сессии (ConnectRequest).

2.7.3. Ожидание подтверждения создания сессии, согласование списка поддерживаемых типов, отправка команд, ожидание ответов и обработка полученных от ИС БД OPM сигналов по кпд4 производится ПУ согласно пунктам 2.4-2.10 Приложения № 3 к Правилам, за исключением команд «запрос на начало, остановку передачи неформатированных данных» (DataStartRequest/DataStopRequest).

2.7.4. При посылке команды «запрос на начало/остановку передачи неформатированных данных», пункт управления OPM ожидает результата

(DataStartResponse) согласно пунктам 2.4-2.5 Приложения № 3 к Правилам, после чего переводит канал кпд4 в режим передачи данных.

2.7.5. В режиме передачи данных ПУ производит прием, декодирование и подтверждение приема данных согласно пунктам 2.3-2.5 Приложения № 4 к Правилам.

2.7.6. При приеме команды «запрос на остановку передачи неформатированных данных» (DataStopRequest) ПУ, находясь в режиме передачи данных, завершает прием блоков данных и переводится в режим передачи команд.

---

## Приложение № 7

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

### **Требования, предъявляемые к функционированию канала кпд5**

1. ИС БД ОРМ обеспечивает подключение ПУ и обработку поступающих запросов по каналу кпд5 в соответствии с приведенной диаграммой состояний переходов. Диаграмма состояний переходов ИС БД ОРМ по кпд5 приведена на рисунке 1.

1.1. ИС БД ОРМ по TCP-порту кпд5 ожидает входящих соединений. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

1.2. Если SSL/TLS-соединение по каналу кпд5 установлено, а сессия не открыта, ИС БД ОРМ реагирует только на сообщение "Запрос на открытие сессии". Создание сессии аналогично представленному в пунктах 1.3 - 1.5 Приложения № 3 к Правилам. При попытке посылок каких-либо других сообщений со стороны ПУ ИС БД ОРМ разрывает TCP соединение по кпд3 и переводит канальный уровень подключения в исходное состояние.

1.3. После создания сессии канал кпд5 ИС БД ОРМ переводится в режим ожидания. При поступлении сообщения "Запрос на открытие сессии" или "Согласование списка поддерживаемых типов" в режиме ожидания, ИС БД ОРМ закрывает текущую сессию и разрывает TCP-соединение.

1.4. Если на ИС БД ОРМ был передан «запрос загрузки данных» (DataLoadRequest) с указанием идентификатора задачи получения содержимого потоков, в случае отсутствия соединения по каналу кпд5 ИС БД ОРМ посылает «ответ на запрос загрузки данных» (DataLoadResponse) с флагом "признака существования результата", установленным в FALSE. При наличии открытой сессии по каналу кпд5, ИС БД ОРМ начинает передачу данных блоков отчетов. ПУ может получать блоки отчетов по кпд5 до получения «ответа на запрос загрузки данных» (DataLoadResponse) по кпд1.

1.5. Если количество переданных без получения «подтверждения» о принятии серии блоков «отчетов» по всем задачам, по которым выполняется загрузка на ПУ данных, меньше «окна канала передачи данных» (параметр data-packet-window-size в запросе создания сессии ConnectRequest), то ИС БД ОРМ выполняет подготовку новых блоков отчетов по загружаемым задачам и посылает их на ПУ. Количество подготовленных и переданных без получения «подтверждения» блоков не превышает размера окна канала передачи данных.

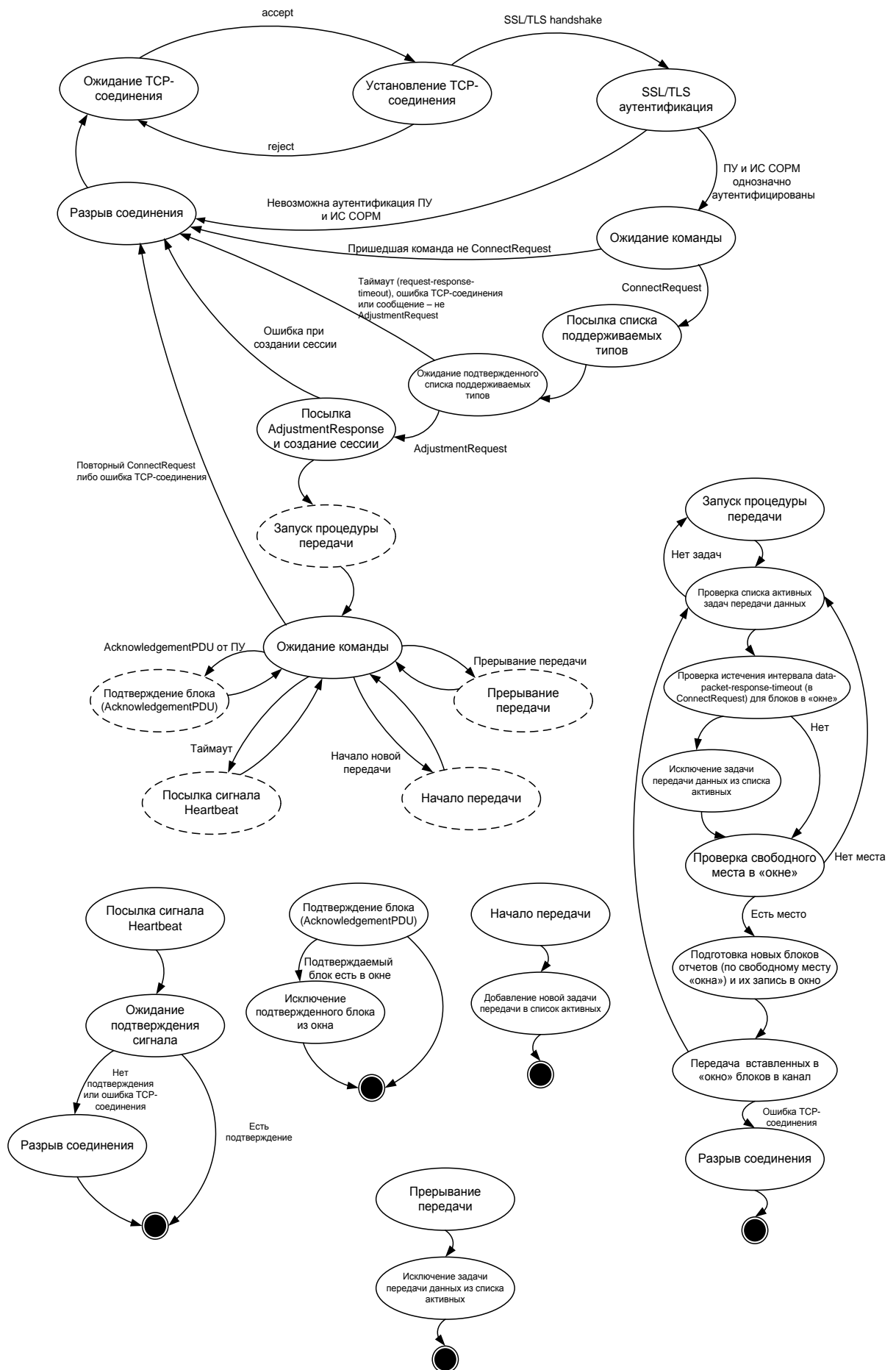


Рисунок 1. Диаграмма состояний перехода ИС БД ОРМ по кпд5

1.6. Максимальная задержка подтверждения приема блока данных со стороны ПУ не превышает параметр «таймаут подтверждения приема блока данных отчета» (data-packet-response-timeout), указываемый при создании сессии. Если задержка подтверждения превысила заданное значение, то оставшиеся для передачи блоки данных не отправляются и передается «сигнал» «Незначительная ошибка ПО, данные не потеряны, дальнейшая работа возможна» с соответствующим описанием проблемы, при этом в поле «reference-message» сообщения «сигнал» указывается идентификатор сообщения блока отчета, по которому не поступило подтверждение приема.

1.7. При получении негативного «подтверждения» блока «отчета» ИС БД ОРМ записывает информацию об ошибочно принятом ПУ блоке и ошибочных записях в блоке в журнал, передача последующих блоков по задаче на ПУ не прерывается. ИС БД ОРМ предоставляет техническому персоналу оператора связи доступ к журналу с записями об ошибочно принятых на ПУ блоках отчетов. Подтвержденные блоки исключаются из окна канала передачи данных (в окне канала передачи данных остаются только неподтвержденные блоки).

1.8. Передача блоков данных прерывается в случае получения ИС БД ОРМ «запроса прерывания загрузки данных».

1.9. Если по кпд5 не производится передача блоков отчетов в течение «максимального времени неактивности» (session-timeout при создании сессии ConnectRequest), ИС БД ОРМ посылает на ПУ «сигнал» heartbeat и ожидает его подтверждения согласно пункту 1.9 Приложения № 3 к Правилам.

2. Диаграмма состояний переходов ПУ по кпд5 приведена на Рисунке 2.

2.1. ПУ с задаваемым интервалом выполняет попытки установления TCP-соединения к ИС БД ОРМ по заданному порту кпд5. После установления соединения выполняется взаимная SSL/TLS-аутентификация.

2.2. После выполнения SSL/TLS-аутентификации ПУ посылает команду создания сессии (ConnectRequest).

2.3. ПУ ожидает сообщения «ответ» от ИС БД ОРМ на отправленную команду в течение времени «таймаут ответа на запрос» (request-response-timeout).

2.4. Если сообщение не получено в течение времени «таймаут ответа на запрос», ПУ разрывает соединение к ИС БД ОРМ и переводит его в изначальное состояние согласно пункту 2.1 настоящего Приложения к Правилам. Время ожидания сообщения «ответ» не зависит от приема сообщений «сигналов», поступающих в этот интервал времени.

2.5. При получении сообщения «Ответ на запрос создания сессии» пункт управления ОРМ создает список поддерживаемых ПУ типов запросов, отчетов и сигналов и отправляет его ИС БД ОРМ. Список поддерживаемых ПУ типов является подмножеством списка типов ИС БД ОРМ согласно пункту 2.3 настоящего Приложения к Правилам.

2.6. При поступлении «запроса загрузки данных конкретной задачи» (DataLoadRequest) ПУ ожидает начала передачи данных в течение времени «таймаут начала передачи блоков отчетов» (data-load-timeout в ConnectRequest). Если данные не поступают в течение вышеописанного периода, то ПУ разрывает

соединение по каналу кпд5 и переводит его в изначальное состояние согласно пункту 2.1 настоящего Приложения к Правилам.

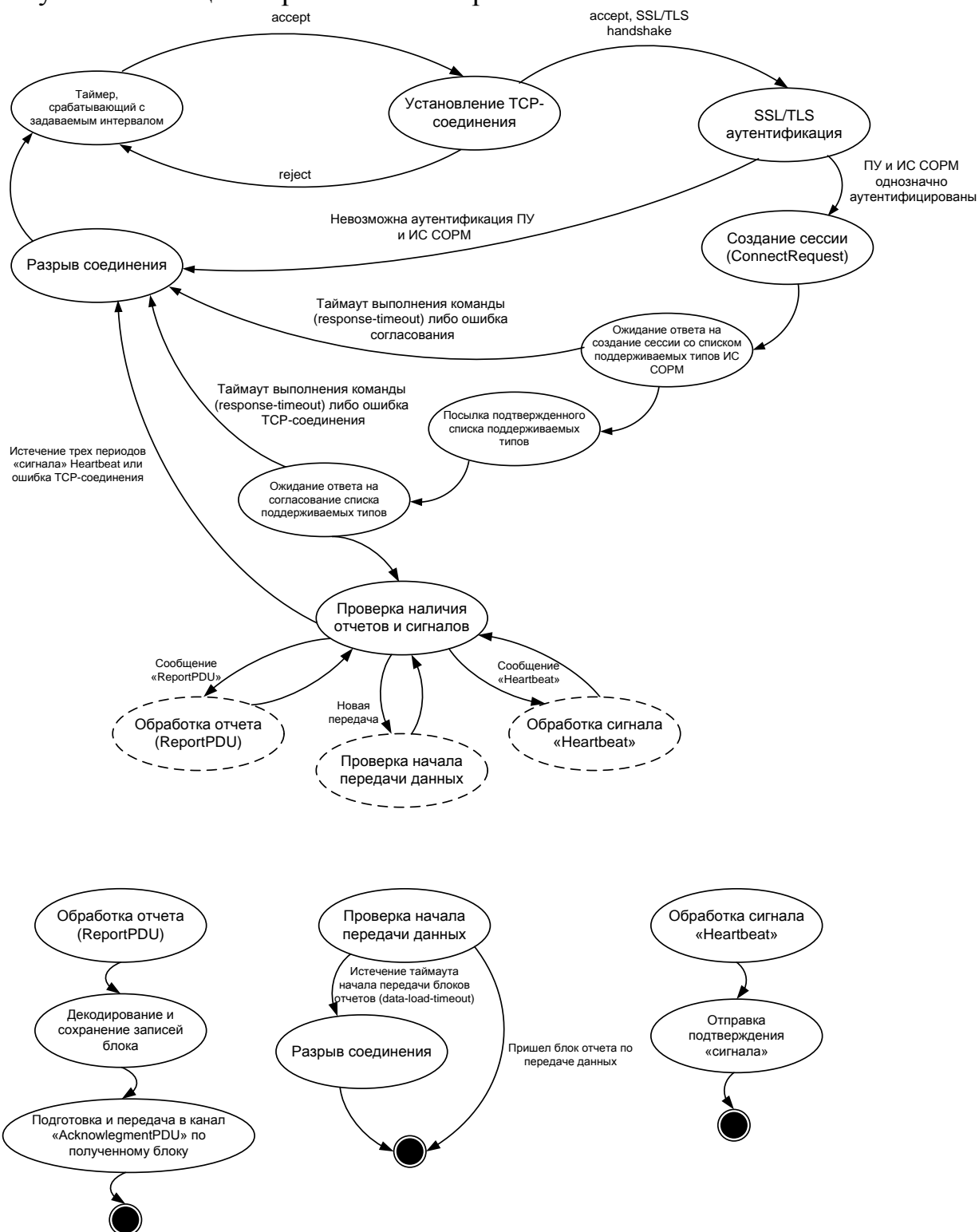


Рисунок 2. Диаграмма состояний перехода ПУ по кпд5

2.7. При поступлении блока отчета ПУ производит декодирование полученного блока и сохранение декодированных данных.

2.8. В ответ на переданный блок данных ПУ посылает сообщение «подтверждение» получения блока отчета. Количество последовательно переданных ИС БД ОРМ блоков данных без подтверждения со стороны ПУ

определяется параметром «размер окна канала передачи данных», который согласовывается при создании сессии. При подтверждении блока отчета ПУ сигнализирует об ошибке декодирования блока. В этом случае в сообщении «подтверждение» приема для ошибочно декодированного блока ПУ, в случае возможности, указывает номер записи в блоке, начиная с которой декодирование не удалось.

2.9. Если при ожидании поступления в ПУ блоков отчетов ИС БД ОРМ не послала «сигнал» heartbeat в течение трех интервалов «максимального времени неактивности» (session-timeout, задается в ConnectRequest) ПУ разрывает соединение по каналу кпд5 и переводит его в изначальное состояние согласно пункту 2.1 настоящего Приложения к Правилам.

---

## Приложение № 8

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий

### Требования к форматам сообщений ИС БД СОРМ

1. Структура разделения ASN.1 модулей протокола взаимодействия ПУ и ИС БД ОРМ приведена на Рисунке 4.
2. Модуль Classification.asn задает правила, в соответствии с которыми:
  - 1) выполняется расширение:
    - а) списка типов запросов к ИС БД ОРМ;
    - б) списка видов поисковых критериев к ИС БД ОРМ;
    - в) списка типов отчетов, формируемых ИС БД ОРМ ИС;
  - 2) выполняется ввод новых версий сообщений протокола.
3. Структура разделения ASN.1 «Сообщений» протокола приведена на Рисунке 1.

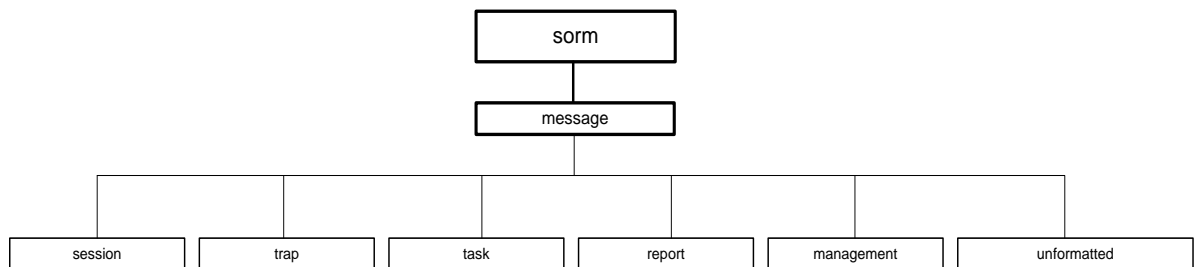


Рисунок 1. Структура видов сообщений протокола взаимодействия ПУ и ИС БД ОРМ

4. На Рисунке 2 представлена структура разделения поисковых критериев кпд1, на Рисунке 3 представлена структура разделения видов отчетов кпд2.
5. В соответствии с рисунками 2, 3 в протокольных ASN.1 модулях выполняется подстановка соответствующих версий форматов поисковых критериев, отчетов, справочников, «сообщений» интерфейса взаимодействия ПУ и ИС БД ОРМ.
6. ASN.1–модуль «Classification.asn» содержит кодированные в иерархическом виде идентификаторы:
  - 1) видов «Сообщений» верхнего уровня интерфейса взаимодействия ПУ и ИС БД ОРМ, составляющих кпд1, кпд2, кпд3, кпд4;
  - 2) видов поисковых критериев для формирования задач к ИС БД ОРМ;

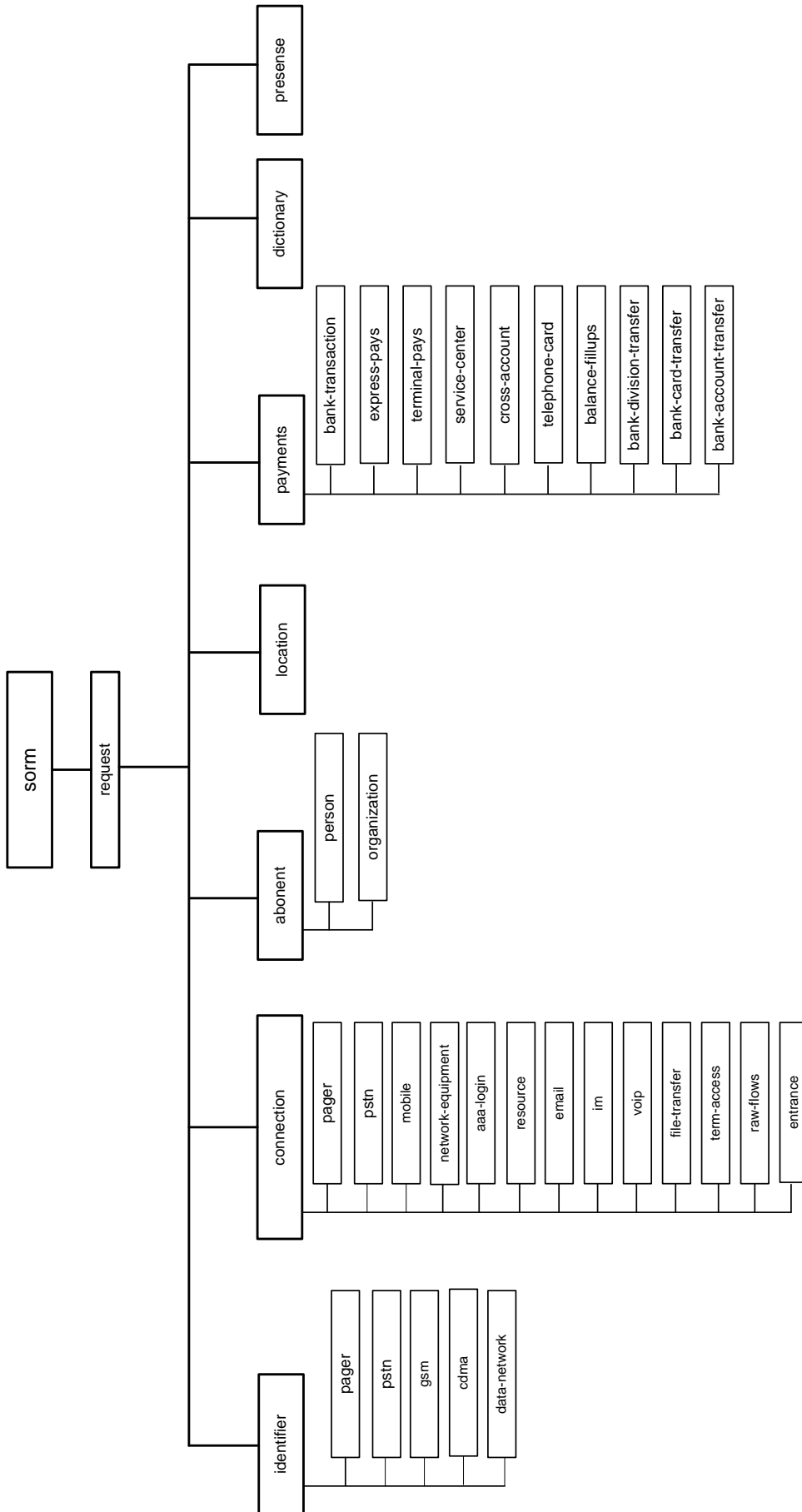


Рисунок 2. Структура разделения поисковых критериев кпд1

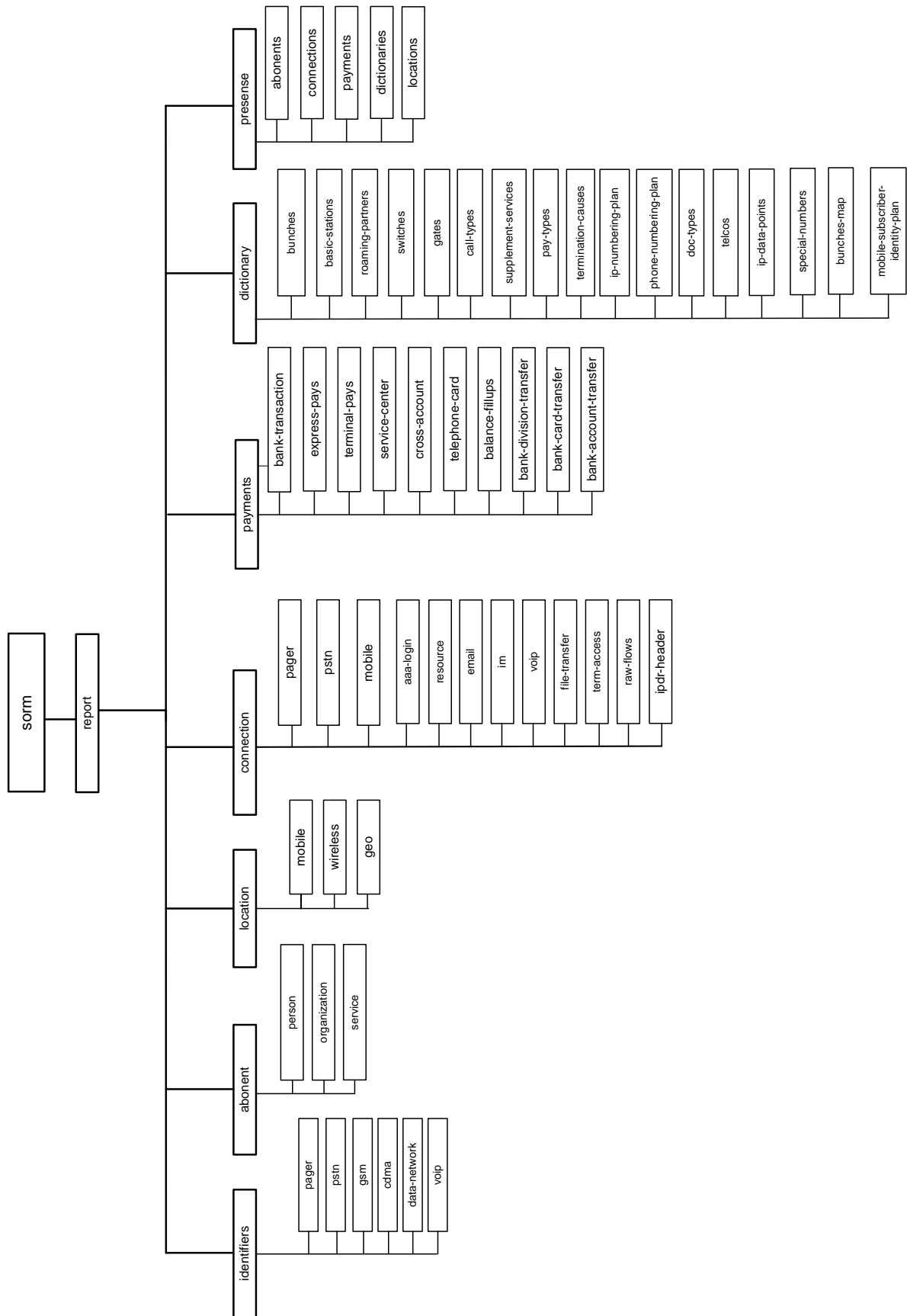


Рисунок 3. Структура разделения видов отчетов кпд2

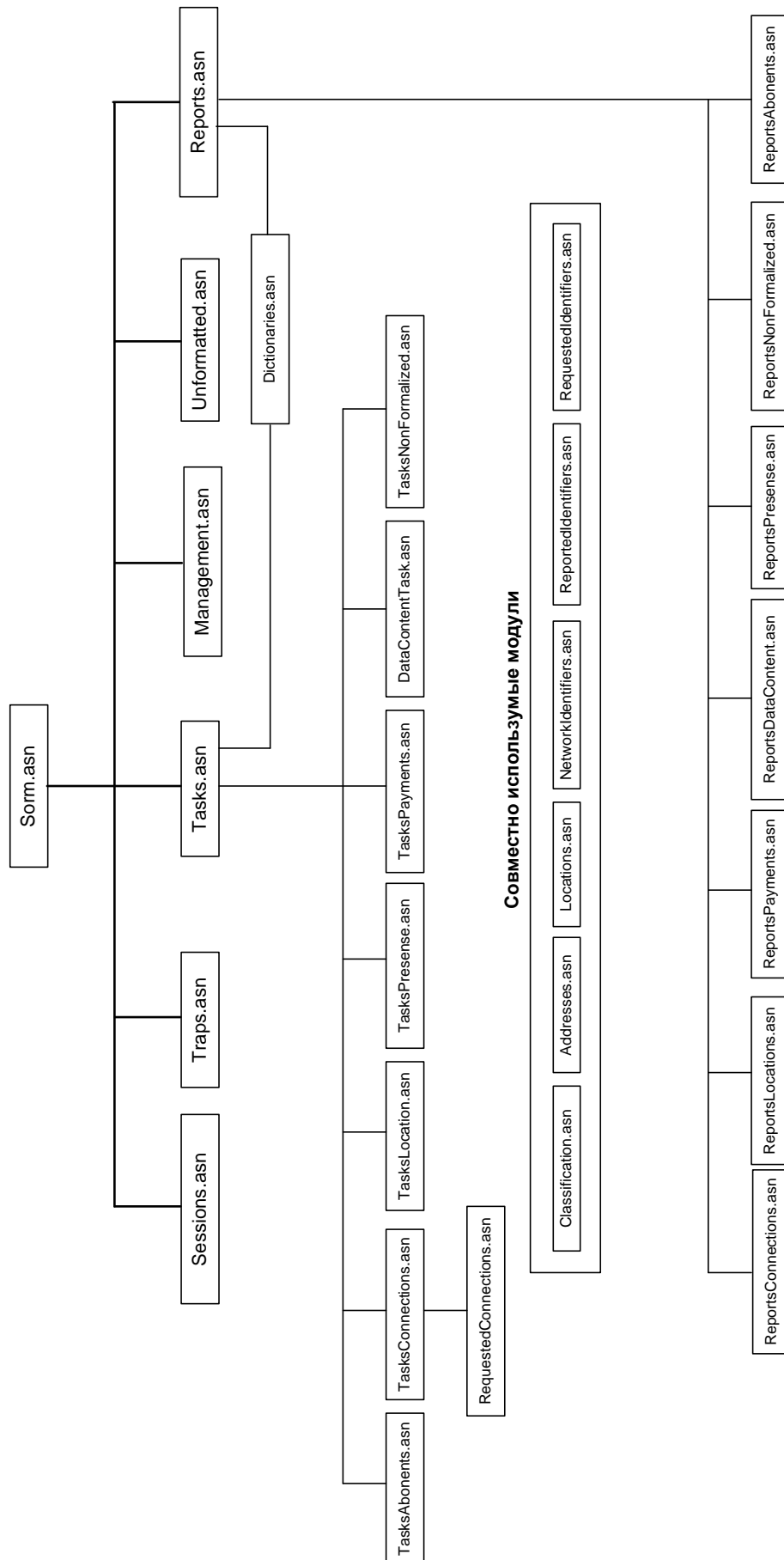


Рисунок 4. Структура ASN.1-модулей интерфейса взаимодействия ПУ и ИС БД ОРМ

3) видов форматов отчетов, формируемых ИС БД ОРМ.

7. Соответствующие идентификаторы используются в других ASN.1-модулях интерфейса взаимодействия ПУ и ИС БД ОРМ (рисунок 4 с модулями), при этом идентификатор определяет конкретную версию и расширения формата соответствующего элемента (поисковых критериев, отчетов, справочников «сообщений» – в соответствии с Рисунками 2 и 3).

8. Предоставленный ИС БД ОРМ при создании сессии перечень идентификаторов и согласованное из него ПУ подмножество в целом определяют конкретные возможности взаимодействия ПУ и ИС БД ОРМ в соответствии с выбранными идентификаторами.

9. Расширение интерфейса взаимодействия ПУ и ИС БД ОРМ обеспечивается введением новых идентификаторов, определяющих соответствующие расширенные элементы (поисковые критерии, отчеты, справочники, «сообщения»). Кодирование новых вводимых идентификаторов элементов осуществляется в соответствии со структурами на Рисунках 2 и 3 и стандартным кодированием ASN.1-модуля «Classification.asn».

---

Приложение № 9  
к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

**ASN.1-спецификация протокола взаимодействия  
пункта управления ОРМ и ИС БД ОРМ**

---

Classification.asn

Classification **DEFINITIONS IMPLICIT TAGS ::=**

**BEGIN**

**EXPORTS TAGGED,**

sorm-message-session,  
sorm-message-trap,  
sorm-message-task,  
sorm-message-report,  
sorm-message-management,  
sorm-message-unformatted,  
sorm-message-filter,

sorm-request-identifier-pager,  
sorm-request-identifier-pstn,  
sorm-request-identifier-gsm,  
sorm-request-identifier-cdma,  
sorm-request-identifier-data-network,

sorm-report-identifier-pager,  
sorm-report-identifier-pstn,  
sorm-report-identifier-gsm,  
sorm-report-identifier-cdma,  
sorm-report-identifier-data-network,  
sorm-report-identifier-voip,

sorm-request-payment-bank-transaction,  
sorm-request-payment-express-pays,  
sorm-request-payment-terminal-pays,  
sorm-request-payment-service-center,  
sorm-request-payment-cross-account,  
sorm-request-payment-telephone-card,  
sorm-request-payment-balance-fillups,  
sorm-request-payment-bank-division-transfer,  
sorm-request-payment-bank-card-transfer,  
sorm-request-payment-bank-account-transfer,

sorm-report-payment-bank-transaction,  
sorm-report-payment-express-pays,  
sorm-report-payment-terminal-pays,  
sorm-report-payment-service-center,  
sorm-report-payment-cross-account,  
sorm-report-payment-telephone-card,  
sorm-report-payment-balance-fillups,  
sorm-report-payment-bank-division-transfer,  
sorm-report-payment-bank-card-transfer,  
sorm-report-payment-bank-account-transfer,

sorm-request-connection-pager,  
sorm-request-connection-pstn,  
sorm-request-connection-mobile,  
sorm-request-connection-aaa-login,  
sorm-request-connection-resource,  
sorm-request-connection-email,  
sorm-request-connection-im,  
sorm-request-connection-voip,  
sorm-request-connection-file-transfer,  
sorm-request-connection-term-access,  
sorm-request-connection-raw-flows,  
sorm-request-connection-entrance,  
sorm-request-connection-address-translations,

sorm-report-connection-pager,  
sorm-report-connection-pstn,  
sorm-report-connection-mobile,  
sorm-report-connection-ipdr-header,  
sorm-report-connection-aaa-login,  
sorm-report-connection-resource,  
sorm-report-connection-email,  
sorm-report-connection-im,  
sorm-report-connection-voip,  
sorm-report-connection-file-transfer,  
sorm-report-connection-term-access,  
sorm-report-connection-raw-flows,  
sorm-report-connection-address-translations,

sorm-request-dictionaries,  
sorm-report-dictionary-bunches,  
sorm-report-dictionary-basic-stations,  
sorm-report-dictionary-roaming-partners,  
sorm-report-dictionary-switches,  
sorm-report-dictionary-gates,  
sorm-report-dictionary-call-types,  
sorm-report-dictionary-supplement-services,  
sorm-report-dictionary-pay-types,  
sorm-report-dictionary-termination-causes,  
sorm-report-dictionary-ip-numbering-plan,  
sorm-report-dictionary-phone-numbering-plan,  
sorm-report-dictionary-doc-types,  
sorm-report-dictionary-telcos,  
sorm-report-dictionary-ip-data-points,  
sorm-report-dictionary-special-numbers,  
sorm-report-dictionary-bunches-map,

sorm-request-presense,  
sorm-report-presense-abonents,  
sorm-report-presense-connections,  
sorm-report-presense-payments,  
sorm-report-presense-dictionaries,  
sorm-report-presense-locations,

sorm-request-abonent-person,  
sorm-request-abonent-organization,

sorm-report-abonent-abonent,  
sorm-report-abonent-service,  
sorm-report-abonent-person,  
sorm-report-abonent-organization,

sorm-report-data-content-raw;

```
TAGGED ::= CLASS {
    &id ObjectDescriptor UNIQUE,
    &Data
```

```

}
WITH SYNTAX {
  OID &id
  DATA &Data
}

```

--- Классификация

OID ::= ObjectDescriptor

-- Подструктура сообщений

sorm-message-session OID ::= "280"

sorm-message-trap OID ::= "281"

sorm-message-task OID ::= "282"

sorm-message-report OID ::= "283"

sorm-message-management OID ::= "284"

sorm-message-unformatted OID ::= "285"

sorm-message-filter OID ::= "286"

-- Идентификаторы

sorm-request-identifier-pager OID ::= "140"

sorm-request-identifier-pstn OID ::= "141"

sorm-request-identifier-gsm OID ::= "142"

sorm-request-identifier-cdma OID ::= "143"

sorm-request-identifier-data-network OID ::= "144"

sorm-report-identifier-pager OID ::= "1"

sorm-report-identifier-pstn OID ::= "2"

sorm-report-identifier-gsm OID ::= "3"

sorm-report-identifier-cdma OID ::= "4"

sorm-report-identifier-data-network OID ::= "5"

sorm-report-identifier-voip OID ::= "6"

-- Параметры соединений

sorm-request-connection-pager OID ::= "160"

sorm-request-connection-pstn OID ::= "161"

sorm-request-connection-mobile OID ::= "162"

sorm-request-connection-aaa-login OID ::= "164"

sorm-request-connection-resource OID ::= "165"

sorm-request-connection-email OID ::= "166"

sorm-request-connection-im OID ::= "167"

sorm-request-connection-voip OID ::= "168"

sorm-request-connection-file-transfer OID ::= "169"

sorm-request-connection-term-access OID ::= "170"

sorm-request-connection-raw-flows OID ::= "171"

sorm-request-connection-entrance OID ::= "172"

sorm-request-connection-address-translations OID ::= "173"

sorm-report-connection-pager OID ::= "20"

sorm-report-connection-pstn OID ::= "21"

sorm-report-connection-mobile OID ::= "22"

sorm-report-connection-ipdr-header OID ::= "23"

sorm-report-connection-aaa-login OID ::= "24"

sorm-report-connection-resource OID ::= "25"

sorm-report-connection-email OID ::= "26"

sorm-report-connection-im OID ::= "27"

sorm-report-connection-voip OID ::= "28"

sorm-report-connection-file-transfer OID ::= "29"

sorm-report-connection-term-access OID ::= "30"

sorm-report-connection-raw-flows OID ::= "31"

sorm-report-connection-address-translations OID ::= "32"

-- Абоненты

sorm-request-abonent-person OID ::= "180"

sorm-request-abonent-organization OID ::= "181"

sorm-report-abonent-abonent OID ::= "40"

sorm-report-abonent-service OID ::= "41"

sorm-report-abonent-person OID ::= "42"

sorm-report-abonent-organization OID ::= "43"

-- Местоположение

sorm-request-location OID ::= "200"

sorm-report-location-mobile OID ::= "60"

sorm-report-location-wireless OID ::= "61"

sorm-report-location-geo OID ::= "62"

-- Платежи

sorm-request-payment-bank-transaction OID ::= "220"

sorm-request-payment-express-pays OID ::= "221"

sorm-request-payment-terminal-pays OID ::= "222"

sorm-request-payment-service-center OID ::= "223"

sorm-request-payment-cross-account OID ::= "224"

sorm-request-payment-telephone-card OID ::= "225"

sorm-request-payment-balance-fillups OID ::= "226"

sorm-request-payment-bank-division-transfer OID ::= "227"

sorm-request-payment-bank-card-transfer OID ::= "228"

sorm-request-payment-bank-account-transfer OID ::= "229"

sorm-report-payment-bank-transaction OID ::= "80"

sorm-report-payment-express-pays OID ::= "81"

sorm-report-payment-terminal-pays OID ::= "82"

sorm-report-payment-service-center OID ::= "83"

sorm-report-payment-cross-account OID ::= "84"

sorm-report-payment-telephone-card OID ::= "85"

sorm-report-payment-balance-fillups OID ::= "86"

sorm-report-payment-bank-division-transfer OID ::= "87"

sorm-report-payment-bank-card-transfer OID ::= "88"

sorm-report-payment-bank-account-transfer OID ::= "89"

-- Справочники

sorm-request-dictionaries OID ::= "240"

sorm-report-dictionary-bunches OID ::= "100"

sorm-report-dictionary-basic-stations OID ::= "101"

sorm-report-dictionary-roaming-partners OID ::= "102"

sorm-report-dictionary-switches OID ::= "103"

sorm-report-dictionary-gates OID ::= "104"

sorm-report-dictionary-call-types OID ::= "105"

sorm-report-dictionary-supplement-services OID ::= "106"

sorm-report-dictionary-pay-types OID ::= "107"

sorm-report-dictionary-termination-causes OID ::= "108"

sorm-report-dictionary-ip-numbering-plan OID ::= "109"

sorm-report-dictionary-phone-numbering-plan OID ::= "110"

sorm-report-dictionary-doc-types OID ::= "111"

sorm-report-dictionary-telcos OID ::= "112"

sorm-report-dictionary-ip-data-points OID ::= "113"

sorm-report-dictionary-special-numbers OID ::= "114"

sorm-report-dictionary-bunches-map OID ::= "115"

sorm-report-dictionary-mobile-subscriber-identity-plan OID ::= "116"

-- Запрос о наличии данных

sorm-request-presense OID ::= "260"

sorm-report-presense-abonents OID ::= "120"

sorm-report-presense-connections OID ::= "121"

sorm-report-presense-payments OID ::= "122"

sorm-report-presense-dictionaries OID ::= "123"

sorm-report-presense-locations OID ::= "124"

-- Запрос о содержимом потоков

sorm-report-data-content-raw OID ::= "50"

END

---

Addresses.asn

Addresses **DEFINITIONS IMPLICIT TAGS** ::=

**BEGIN**

```
EXPORTS AddressType,
         ReportedAddresses,
         ReportedAddress,
         RequestedAddress;
```

```
AddressType ::= ENUMERATED {
  registered (0),           --- Адрес регистрации (обязателен для юридических и физических лиц)
  postal (1),              --- Почтовый адрес (дополнительный адрес для юридических лиц)
  invoice (2),            --- Адрес доставки счета (дополнительный адрес для юридических лиц)
  device-location (3),    --- Адрес установки устройства (телефонов) (дополнительный адрес для
абонентов                                     --- телефонов общего пользования)

  reserved (4)            --- Резерв
}
```

```
ReportedAddresses ::= SEQUENCE OF ReportedAddress
```

```
ReportedAddress ::= SEQUENCE {
  title      AddressType,   --- тип адреса
  address-info AddressInfoReport --- адрес
}
```

```
AddressInfoReport ::= CHOICE {
  struct-info[1] AddressStructInfoReport, --- структурированный адрес
  unstruct-info[2] UTF8String(SIZE (1 .. 1024)) --- неструктурированный адрес
}
```

```
AddressStructInfoReport ::= SEQUENCE {
  zip [0] UTF8String (SIZE (1 .. 32)) OPTIONAL, --- почтовый индекс, zip-код
  country [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- страна
  region [2] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- область
  zone [3] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- район, муниципальный округ
  city [4] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- город, поселок, деревня
  street [5] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- улица
  building [6] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- дом, строение
  build-sect [7] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- корпус
  apartment [8] UTF8String (SIZE (1 .. 128)) OPTIONAL --- квартира, офис
}
```

```
-- поля адресных данных
```

```
RequestedAddress ::= SEQUENCE {
  zip [0] UTF8String (SIZE (1 .. 32)) OPTIONAL, --- почтовый индекс, zip-код
  country [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- страна
  region [2] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- область
  zone [3] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- район, муниципальный округ
  city [4] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- город, поселок, деревня, населенный пункт
  street [5] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- улица
  building [6] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- дом, строение
  build-sect [7] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- корпус
  apartment [8] UTF8String (SIZE (1 .. 128)) OPTIONAL --- квартира, офис
}
```

```
END
```

## Dictionaries.asn

```
Dictionaries DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

**EXPORTS**

```
TelcoID,
TelcoList,
```

```
DictionaryTask,
DictionaryReport,
```

```
PhoneAbonentType;
```

**IMPORTS** DateAndTime**FROM** Sorm

```
TAGGED,
sorm-request-dictionaries,
sorm-report-dictionary-telcos,
sorm-report-dictionary-bunches,
sorm-report-dictionary-basic-stations,
sorm-report-dictionary-roaming-partners,
sorm-report-dictionary-switches,
sorm-report-dictionary-gates,
sorm-report-dictionary-call-types,
sorm-report-dictionary-supplement-services,
sorm-report-dictionary-pay-types,
sorm-report-dictionary-termination-causes,
sorm-report-dictionary-ip-numbering-plan,
sorm-report-dictionary-phone-numbering-plan,
sorm-report-dictionary-doc-types,
sorm-report-dictionary-ip-data-points,
sorm-report-dictionary-special-numbers,
sorm-report-dictionary-bunches-map,
sorm-report-dictionary-mobile-subscriber-identity-plan
```

**FROM** Classification

```
ReportedAddress
```

**FROM** Addresses

```
Bunch,
DataNetworkEquipment,
IPAddress,
IPPort,
IPMask,
NetworkPeerInfo,
NetworkType
```

**FROM** NetworkIdentifiers

```
GeoLocation FROM Locations;
```

```
-- Идентификатор оператора связи или филиала
```

```
TelcoID ::= INTEGER (0 .. 65535)
```

```
--- список идентификаторов операторов связи или филиалов
```

```
TelcoList ::= SEQUENCE OF TelcoID
```

```
-- Запрос
```

```
DictionaryTask ::= SEQUENCE {
  id TAGGED.&id ( {DictionaryTaskVariants} ),
  data TAGGED.&Data ( {DictionaryTaskVariants} { @id } )
}
```

```
DictionaryTaskVariants TAGGED ::= { dictionaryTask }
```

```
dictionaryTask TAGGED ::= {
  OID { sorm-request-dictionaries }
  DATA ObjectDescriptor
}
```

```
-- тип запрашиваемого справочника (идентификатор отчёта)
```

```

-- ObjectDescriptor принимает значение одно из:
-- sorm-report-dictionary-telcos
-- sorm-report-dictionary-bunches
-- sorm-report-dictionary-basic-stations
-- sorm-report-dictionary-roaming-partners
-- sorm-report-dictionary-switches
-- sorm-report-dictionary-gates
-- sorm-report-dictionary-call-types
-- sorm-report-dictionary-supplement-services
-- sorm-report-dictionary-pay-types
-- sorm-report-dictionary-termination-causes
-- sorm-report-dictionary-ip-numbering-plan
-- sorm-report-dictionary-phone-numbering-plan
-- sorm-report-dictionary-doc-types
-- sorm-report-dictionary-ip-data-points
-- sorm-report-dictionary-special-numbers
-- sorm-report-dictionary-bunches-map,
-- sorm-report-dictionary-mobile-subscriber-identity-plan

-- Отчёт
DictionaryReport ::= SEQUENCE {
  id      TAGGED.&id ({DictionaryRecordsVariants}),      --- идентификтор записи справочника
  data    TAGGED.&Data({DictionaryRecordsVariants}){@id} --- данные записи справочника
}

DictionaryRecordsVariants TAGGED ::= {
  telcosRecords          --- операторы связи, обслуживаемые ИС СОРМ
  | bunchesRecords       --- пучки соединительных линий
  | basicStationsSectorRecords --- базовые станции
  | roamingPartnersRecords --- роуминговые партнеры
  | switchesRecords      --- коммутаторы
  | gatesRecords         --- IP шлюзы
  | callTypesRecords     --- типы вызовов
  | supplementServicesRecords --- список ДВО
  | payTypesRecords      --- способы оплаты (пополнения баланса)
  | terminationCausesRecords --- причины завершения соединения
  | ipNumberingPlanRecords --- IP-план адресации
  | telephoneNumberingPlanRecords --- план телефонной номерной емкости
  | docTypesRecords      --- типы документов, удостоверяющих личность
  | ipDataPointsRecords  --- идентификаторы точек подключения к сети
  передачи данных, с которых получены записи о соединениях
  | specialNumbersRecords --- специальные номера оператора (SMS-центры, ТМС,
  сервисы)
  | bunchesMapRecords    --- карта связей пучков соединительных линий
  | mobileSubscriberIdentityPlanRecords --- план нумерации идентификаторов мобильных
  телефонных абонентов
}

--- операторы связи, обслуживаемые ИС СОРМ
telcosRecords TAGGED ::= {
  OID { sorm-report-dictionary-telcos }
  DATA SEQUENCE OF TelcosRecord }

TelcosRecord ::= SEQUENCE {
  telco-id   TelcoID,      --- номер филиала или оператора связи
  begin-time DateAndTime,  --- время начала действия
  end-time   DateAndTime OPTIONAL, --- время конца действия
  description UTF8String (SIZE (1 .. 256)), --- описание (наименование) оператора связи или
  филиала
  mcc [0]   NumericString (SIZE(3)) OPTIONAL, --- код страны
  mnc [1]   NumericString (SIZE(3)) OPTIONAL --- код оператора связи
}

--- пучки соединительных линий
bunchesRecords TAGGED ::= {
  OID { sorm-report-dictionary-bunches }
  DATA SEQUENCE OF BunchRecord
}

```

```

}

BunchRecord ::= SEQUENCE {
telco-id TelcoID,
bunch-id Bunch,
switch-id UTF8String (SIZE (1 .. 128)),
bunch-type ENUMERATED {
inbound (0),
outbound (1),
bidirectional (3)
},
begin-time DateAndTime,
end-time DateAndTime OPTIONAL,
description UTF8String (SIZE(1 .. 256))
}

--- базовые станции
basicStationsSectorRecords TAGGED ::= {
OID { sorm-report-dictionary-basic-stations }
DATA SEQUENCE OF BasicStationSectorRecord
}

BasicStationSectorRecord ::= SEQUENCE {
telco-id TelcoID,
begin-time DateAndTime,
end-time DateAndTime,
address UTF8String (SIZE (1 .. 256)),
расположения

sector-identifiers BasicStationIdentifiers,
antenna-configuration BasicStationAntenna,
station-type BasicStationType,

structured-address [0] ReportedAddress OPTIONAL,
location [1] GeoLocation OPTIONAL
}

--- идентификаторы сектора
BasicStationIdentifiers ::= CHOICE {
telephone [0] TelephoneIdentifiers,
wireless [1] SEQUENCE OF WirelessIdentifiers
}

--- идентификаторы сектора для телефонной сети
TelephoneIdentifiers ::= SEQUENCE {
lac INTEGER (0 .. 65535),
cell INTEGER (0 .. 4294967295),
(идентификатор базовой станции и сектор)
cell-sign UTF8String (SIZE (1 .. 18)) OPTIONAL
}

--- идентификаторы сектора для сети передачи данных
WirelessIdentifiers ::= SEQUENCE {
cell UTF8String (SIZE (1 .. 64)),
ip-list IPList OPTIONAL,
mac OCTET STRING (SIZE (6)) OPTIONAL
}

IPList ::= SEQUENCE OF NetworkPeerInfo

--- параметры антенной системы
BasicStationAntenna ::= CHOICE {
gsm-antenna [0] GsmAntenna,
cdma-antenna [1] SEQUENCE OF CdmaAntenna,
wireless-antenna [2] SEQUENCE OF WirelessAntenna
сектора
}

```

--- идентификатор оператора связи или филиала  
--- идентификатор пучка  
--- идентификатор коммутатора  
--- тип - входящий/исходящий

--- время начала назначения пучка  
--- время конца назначения пучка  
--- расшифровка пучка

--- идентификатор оператора связи или филиала  
--- время начала действия базовой станции  
--- время конца действия базовой станции  
--- произвольное текстовое описание адреса или места

--- идентификаторы сектора  
--- параметры антенной системы  
--- тип сети базовой станции

--- адрес места установки базовой станции  
--- географическое местоположение

--- идентификаторы сектора для телефонной сети  
--- идентификаторы сектора для сети передачи данных

--- код зоны  
--- идентификатор сектора базовой станции

--- телефонный идентификатор соты

--- идентификатор сектора  
--- перечень назначенных сектору IP-адресов/портов  
--- MAC-адрес сетевого оборудования сектора

--- параметры антенной системы GSM-сектора  
--- параметры антенной системы CDMA-сектора  
--- параметры антенной системы WiFi/WiMAX-

```

--- параметры антенной системы GSM-сектора
GsmAntenna ::= SEQUENCE {
  azimuth      INTEGER (-1 .. 359),
  -1, то нет направленности
  width        INTEGER (0 .. 359),
  horizon-angle  INTEGER (0 .. 359),
  power [0]    INTEGER (0 .. 25000) OPTIONAL,
  frequency [1] INTEGER (0 .. 10000000000) OPTIONAL,
  vertical-lift [2] INTEGER (0 .. 100) OPTIONAL,
  gain-factor [3] INTEGER (-100 .. 100) OPTIONAL,
  polarization [4] INTEGER (-45 .. 45) OPTIONAL
}
--- азимут относительно направления на север, в градусах, если
--- ширина растра в градусах
--- угол наклона сектора к горизонту
--- мощность в ваттах (сектор)
--- частота излучения (сектор)
--- высота подвеса сектора
--- коэффициент усиления антенны (Дб)
--- поляризация антенной системы

CdmaAntenna ::= BroadbandWirelessParameters
WirelessAntenna ::= BroadbandWirelessParameters

BroadbandWirelessParameters ::= SEQUENCE {
  azimuth      INTEGER (-1 .. 359),
  -1, то нет направленности
  width        INTEGER (0 .. 359),
  horizon-angle  INTEGER (0 .. 359),
  power [0]    INTEGER (0 .. 25000) OPTIONAL,
  frequency-start [1] INTEGER (0 .. 10000000000) OPTIONAL,
  frequency-stop [2] INTEGER (0 .. 10000000000) OPTIONAL,
  leaf-level [3] INTEGER (-45 .. 45) OPTIONAL,
  vertical-lift [4] INTEGER (0 .. 100) OPTIONAL,
  gain-factor [5] INTEGER (-100 .. 100) OPTIONAL,
  polarization [6] INTEGER (-45 .. 45) OPTIONAL
}
--- азимут относительно направления на север, в градусах, если
--- ширина растра в градусах
--- угол наклона сектора к горизонту
--- мощность в ваттах (сектор)
--- нижняя частота излучения диапазона (сектор)
--- верхняя частота излучения диапазона (сектор)
--- уровень боковых лепестков
--- высота подвеса сектора
--- коэффициент усиления антенны (Дб)
--- поляризация антенной системы

--- виды базовых станций
BasicStationType ::= ENUMERATED {
  gsm (0),
  cdma (1),
  umts (2),
  wifi (3),
  wimax (4)
}

roamingPartnersRecords TAGGED ::= {
  OID { sorm-report-dictionary-roaming-partners }
  DATA SEQUENCE OF RoamingPartnerRecord
}

RoamingPartnerRecord ::= SEQUENCE {
  telco-id TelcoID,
  roaming-id INTEGER (0 .. 4294967295),
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256))
}
--- идентификатор оператора связи или филиала
--- идентификатор роумингового партнёра
--- время начала действия роуминга
--- время конца действия роуминга
--- описание

switchesRecords TAGGED ::= {
  OID { sorm-report-dictionary-switches }
  DATA SEQUENCE OF SwitchesRecord
}

SwitchesRecord ::= SEQUENCE {
  telco-id TelcoID,
  switch-id UTF8String (SIZE (1 .. 128)),
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256)),
  network-type NetworkType,
  address ReportedAddress,
  switch-sign NumericString (SIZE (1 .. 18)) OPTIONAL,
  switch-type ENUMERATED {
    internal(0),
    --- внутренний
  }
  --- идентификатор оператора связи или филиала
  --- идентификатор коммутатора
  --- время начала действия коммутатора
  --- время конца действия коммутатора
  --- описание
  --- тип сети связи
  --- адрес места установки коммутатора
  --- телефонный идентификатор коммутатора

```

```

    border(1)          --- пограничный
  }
}

gatesRecords TAGGED ::= {
  OID { sorm-report-dictionary-gates }
  DATA SEQUENCE OF GatesRecord
}

GatesRecord ::= SEQUENCE {
  telco-id TelcoID,
  gate-id INTEGER (0 .. 4294967295),
  ip-list IPList,
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256)),
  address ReportedAddress,
  gate-type ENUMERATED {
    sgsn(0),
    ggsn(1),
    smsc(2),
    gmsc(3),
    hss(4),
    pstn(5),
    voip-gw(6),
    aaa(7)
  }
}
--- идентификатор оператора связи или филиала
--- идентификатор шлюза
--- IP адрес шлюза
--- время начала действия шлюза
--- время конца действия шлюза
--- описание
--- адрес места установки шлюза
--- тип IP-шлюза

callTypesRecords TAGGED ::= {
  OID { sorm-report-dictionary-call-types }
  DATA SEQUENCE OF CallsTypesRecord
}

CallsTypesRecord ::= SEQUENCE {
  telco-id TelcoID,
  call-type-id INTEGER (0 .. 4294967295),
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256))
}
--- идентификатор оператора связи или филиала
--- идентификатор типа вызова
--- время начала действия
--- время конца действия
--- описание

supplementServicesRecords TAGGED ::= {
  OID { sorm-report-dictionary-supplement-services }
  DATA SEQUENCE OF SupplementServicesRecord
}

SupplementServicesRecord ::= SEQUENCE {
  telco-id TelcoID,
  service-id INTEGER (0 .. 4294967295),
  mnemonic UTF8String (SIZE (1..10)),
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256))
}
--- идентификатор оператора связи или филиала
--- идентификатор сервиса
--- мнемоническое обозначение
--- время начала действия
--- время конца действия
--- описание

payTypesRecords TAGGED ::= {
  OID { sorm-report-dictionary-pay-types }
  DATA SEQUENCE OF PayTypesRecord
}

PayTypesRecord ::= SEQUENCE {
  telco-id TelcoID,
  pay-type-id INTEGER (0 .. 4294967295),
  begin-time DateAndTime,
  end-time DateAndTime OPTIONAL,
  description UTF8String (SIZE (1 .. 256))
}
--- идентификатор оператора связи или филиала
--- идентификатор типа оплаты
--- время начала действия
--- время конца действия
--- описание

```

```

}

terminationCausesRecords TAGGED ::= {
  OID { sorm-report-dictionary-termination-causes }
  DATA SEQUENCE OF TerminationCausesRecord
}

TerminationCausesRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  termination-cause-id INTEGER (0 .. 16384), --- код причины
  begin-time    DateAndTime,            --- время начала действия
  end-time      DateAndTime OPTIONAL,   --- время конца действия
  description   UTF8String (SIZE (1 .. 256)), --- описание
  network-type  NetworkType             --- тип сети связи
}

ipNumberingPlanRecords TAGGED ::= {
  OID { sorm-report-dictionary-ip-numbering-plan }
  DATA SEQUENCE OF IpNumberingPlanRecord }

IpNumberingPlanRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  description   UTF8String (SIZE (1 .. 256)), --- описание назначения диапазона
  network-address IPAddress,            --- подсеть
  network-mask  IPMask,                 --- маска подсети
  begin-time    DateAndTime,            --- время начала действия
  end-time      DateAndTime OPTIONAL,   --- время конца действия
}

telephoneNumberingPlanRecords TAGGED ::= {
  OID { sorm-report-dictionary-phone-numbering-plan }
  DATA SEQUENCE OF TelephoneNumberingPlanRecord
}

TelephoneNumberingPlanRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  iso-3166-alpha-2 UTF8String (SIZE (2)), --- 2-х символьная аббревиатура страны
  iso-3166-alpha-3 UTF8String (SIZE (3)), --- 3-х символьная аббревиатура страны
  country-code   UTF8String (SIZE (3)), --- международный код страны
  national-significant-number UTF8String (SIZE (14)), --- номерной телефонный префикс оператора связи,
  включая код зоны
  area-code-length INTEGER (0 .. 6), --- длина кода зоны в телефонном префиксе оператора
  связи
  min-subscr-nr-length INTEGER (1 .. 15), --- минимальная длина абонентского номера, символов
  (national-significant-number + min-subscr)
  max-subscr-nr-length INTEGER (1 .. 15), --- максимальная длина абонентского номера,
  символов (national-significant-number + max-subscr)
  utc-min         INTEGER (-12 .. 12), --- минимальный часовой пояс
  utc-max        INTEGER (-12 .. 12), --- максимальный часовой пояс
  destination    UTF8String (SIZE (2 .. 255)), --- страна
  operator-type-id NetworkType,         --- тип сети связи оператора

  capacity-from  NumericString (SIZE (1 .. 15)), --- нижняя граница диапазона выданных номеров (от)
  capacity-to    NumericString (SIZE (1 .. 15)), --- верхняя граница диапазона выданных номеров (до)
  capacity-size  INTEGER (1 .. 1000000), --- количество выделенных номеров в диапазоне
  (емкость)

  location       UTF8String (SIZE (0 .. 255)), --- текстовое описание местоположения оператора связи
  registrar      UTF8String (SIZE (0 .. 255)), --- наименование оператора связи
  range-activation DateAndTime, --- дата и время начала действия номерной емкости
  mobile-country-code NumericString (SIZE(3)), --- MCC
  mobile-network-code NumericString (SIZE(3)), --- MNC

  range-deactivation [0] DateAndTime OPTIONAL, --- дата и время завершения действия номерной
  емкости
  range-status [1] UTF8String (SIZE (2 .. 128)) OPTIONAL, --- текущее состояние номерной емкости
  description [2] UTF8String (SIZE (2 .. 255)) OPTIONAL, --- расшифровка оказываемых услуг связи по
  номерной емкости

```

```

operating-company-number [3] UTF8String (SIZE (0 .. 4)) OPTIONAL --- международный идентификатор оператора связи
}

docTypesRecords TAGGED ::= {
  OID { sorm-report-dictionary-doc-types }
  DATA SEQUENCE OF DocTypesRecord
}

DocTypesRecord ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор оператора связи или филиала
  doc-type-id INTEGER (0 .. 65535), --- идентификатор типа документа
  begin-time DateAndTime, --- время начала действия
  end-time DateAndTime OPTIONAL, --- время конца действия
  description UTF8String (SIZE (1 .. 256)) --- описание (наименование)
}

ipDataPointsRecords TAGGED ::= {
  OID { sorm-report-dictionary-ip-data-points }
  DATA SEQUENCE OF IpDataPointRecord
}

IpDataPointRecord ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор оператора связи или филиала
  point-id INTEGER (0 .. 1000), --- идентификатор точки подключения
  description UTF8String (SIZE (1 .. 256)), --- описание (наименование) точки подключения
  begin-time DateAndTime, --- время начала действия
  end-time DateAndTime OPTIONAL --- время конца действия
}

specialNumbersRecords TAGGED ::= {
  OID { sorm-report-dictionary-special-numbers }
  DATA SEQUENCE OF SpecialNumberRecord
}

SpecialNumberRecord ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор оператора связи или филиала
  directory-number UTF8String (SIZE (2 .. 32)), --- специальный номер
  description UTF8String (SIZE (1 .. 256)), --- описание (наименование, назначение) специального
номера
  begin-time DateAndTime, --- время начала действия
  end-time DateAndTime OPTIONAL, --- время конца действия
  network-address IPAddress OPTIONAL --- адрес в сети передачи данных
}

bunchesMapRecords TAGGED ::= {
  OID { sorm-report-dictionary-bunches-map }
  DATA SEQUENCE OF BunchesMapRecord
}

BunchesMapRecord ::= SEQUENCE {
  a-bunch BunchMapPoint, --- пара коммутатор/пучок в связи
  b-bunch BunchMapPoint, --- пара коммутатор/пучок в связи
  begin-time DateAndTime, --- время начала действия связи
  end-time DateAndTime OPTIONAL --- время конца действия связи
}

BunchMapPoint ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор оператора связи или филиала
  switch-id UTF8String (SIZE (1 .. 128)), --- идентификатор коммутатора
  bunch-id Bunch --- идентификатор пучка коммутатора
}

mobileSubscriberIdentityPlanRecords TAGGED ::= {
  OID { sorm-report-dictionary-mobile-subscriber-identity-plan }
  DATA SEQUENCE OF MobileSubscriberIdentityPlanRecord
}

```

```

}

MobileSubscriberIdentityPlanRecord ::= SEQUENCE {
telco-id      TelcoID,          --- идентификатор оператора связи или филиала
mcc           NumericString (SIZE(3)), --- код страны
mnc           NumericString (SIZE(3)), --- код оператора связи
area-code     NumericString (SIZE(3 .. 10)), --- код зоны/региона
capacity-from NumericString (SIZE (0 .. 7)), --- нижняя граница диапазона (от)
capacity-to   NumericString (SIZE (0 .. 7)), --- верхняя граница диапазона (до)
capacity-size INTEGER (1 .. 1000000), --- количество выделенных в диапазоне (емкость)
description   UTF8String (SIZE (2 .. 255)), --- описание, назначение
region        UTF8String (SIZE (1 .. 128)), --- область
city          UTF8String (SIZE (1 .. 128)), --- город, поселок, деревня
range-activation DateAndTime, --- дата и время начала действия номерной емкости
range-deactivation [0] DateAndTime OPTIONAL, --- дата и время завершения действия номерной емкости
range-status [1] UTF8String (SIZE (2 .. 128)) OPTIONAL --- текущее состояние номерной емкости
}

```

```

-- Тип абонента
PhoneAbonentType ::= ENUMERATED {
local (0),          --- абонент данного коммутатора
network (1),       --- абонент сети связи
roamer (2),        --- роумер
undefined (3)
}

```

END

---

## Locations.asn

Locations DEFINITIONS IMPLICIT TAGS ::= BEGIN

### EXPORTS

Location,  
GeoLocation;

```

Location ::= CHOICE {
mobile-location [0] MobileLocation, --- местоположение мобильного абонента
wireless-location [1] WirelessLocation, --- местоположение абонента мобильной сети передачи данных
geo-location [2] GeoLocation --- географическое местоположение
}

```

```

MobileLocation ::= SEQUENCE {
lac INTEGER (0 .. 65535), --- код зоны
cell INTEGER (0 .. 4294967295), --- идентификатор базовой станции
ta [0] INTEGER (0 .. 63) OPTIONAL --- Timing Advance (временная компенсация)
}

```

```

WirelessLocation ::= SEQUENCE {
cell UTF8String (SIZE (1 .. 64)), --- идентификатор сектора
mac OCTET STRING (SIZE (6)) --- MAC-адрес сетевого оборудования сектора
}

```

```

GeoLocation ::= SEQUENCE {
latitude-grade REAL, --- широта
longitude-grade REAL, --- долгота
projection-type ENUMERATED {
wgs84 (0),
utm (1),
sgs85 (2)
}
}

```

END

---

## Management.asn

Management DEFINITIONS IMPLICIT TAGS ::= BEGIN

**EXPORTS** managementMessage;

**IMPORTS TAGGED,**

sorm-message-management  
**FROM** Classification;

```
managementMessage TAGGED ::= {
  OID { sorm-message-management }
  DATA CHOICE {
    request [0] ManagementRequest,
    response [1] ManagementResponse
  }
}
```

--- тип сообщения "команда управления ИС СОРМ"

ManagementRequest ::= **CHOICE** {

get-structure [0] GetStructureRequest,  
СПО

get-module-config [1] GetModuleConfigRequest,

set-module-config [2] SetModuleConfigRequest,

check-module [3] CheckModuleRequest,

get-module-types [4] GetModuleTypesRequest

}

--- запрос на получение структуры ИС СОРМ - КТС и модулей

--- запрос на получение конфигурации КТС/модуля СПО

--- запрос на изменение конфигурации КТС/модуля СПО

--- запрос на получение состояния модуля

--- запрос на получение типов модулей КТС и СПО

--- запрос на получение структуры ИС СОРМ - КТС и модулей СПО

GetStructureRequest ::= **NULL**

--- запрос на получение конфигурации КТС/модуля СПО

GetModuleConfigRequest ::= **CHOICE** {

hw-modules-list [0] RequestedHardwareModules,

sw-modules-list [1] RequestedSoftwareModules

}

--- перечень идентификаторов узлов КТС ИС СОРМ

--- перечень идентификаторов модулей СПО ИС СОРМ

RequestedHardwareModules ::= **SEQUENCE OF** ModuleId

RequestedSoftwareModules ::= **SEQUENCE OF** ModuleId

--- перечень идентификаторов узлов КТС ИС СОРМ

--- перечень идентификаторов модулей СПО ИС СОРМ

--- запрос на изменение конфигурации КТС/модуля СПО

SetModuleConfigRequest ::= **SEQUENCE** {

module-id ModuleId,

module-config ConfiguredModule

}

--- идентификатор конфигурируемого модуля

--- устанавливаемая в модуль конфигурация

ConfiguredModule ::= **CHOICE** {

sw-module [0] SormSoftwareModule,

hw-module [1] SormHardwareModule

}

--- для узла КТС

--- для узла СПО

--- запрос на получение состояния модуля

CheckModuleRequest ::= RequestedModulesList

RequestedModulesList ::= **CHOICE** {

sw-modules [0] RequestedHardwareModules,

состояние

hw-modules [1] RequestedSoftwareModules

запрашивается состояние

}

--- идентификаторы узлов КТС, для которых запрашивается

--- идентификаторы модулей ИС СОРМ, для которых

--- запрос на получение типов модулей КТС и СПО

GetModuleTypesRequest ::= **NULL**

--- уникальный идентификатор КТС/модуля СПО ИС СОПМ  
 ModuleId ::= **OCTET STRING (SIZE (8))**

--- параметр модуля  
 ModuleParameter ::= **SEQUENCE** {  
   parameter-name **UTF8String (SIZE (1 .. 256))**,  
   read-only **BOOLEAN**,  
   parameter-value ParameterValue  
 }  
 --- наименование параметра  
 --- контролируемый или измеряемый параметр  
 --- значение параметра

--- варианты значений параметров  
 ParameterValue ::= **CHOICE** {  
   string [0] **UTF8String (SIZE (1 .. 256))**,  
   integer [1] **INTEGER (0 .. 999999999)**,  
   boolean [2] **BOOLEAN**  
 }

ModuleParameters ::= **SEQUENCE OF** ModuleParameter

SormSoftwareModule ::= **SEQUENCE** {  
   module-id ModuleId,  
   hardware-module-id ModuleId,  
   модуль СПО  
   block-name **INTEGER (0 .. 1024)**,  
   module-name **UTF8String (SIZE (1 .. 512))**,  
   module-type **INTEGER (1 .. 512)**,  
   module-parameters ModuleParameters,  
   sub-modules-list SubmodulesList **OPTIONAL**  
 }  
 --- уникальный идентификатор данного модуля  
 --- идентификатор КТС, на котором работает данный блок  
 --- номер блока СПО модуля  
 --- наименование модуля  
 --- идентификатор типа модуля  
 --- список параметров модуля  
 --- submodule

SubmodulesList ::= **SEQUENCE OF** SormSoftwareModule

SormHardwareModule ::= **SEQUENCE** {  
   module-id ModuleId,  
   block-name **INTEGER (0 .. 1024)**,  
   module-name **UTF8String (SIZE (1 .. 512))**,  
   module-parameters HwParameterGroups  
 }  
 --- уникальный идентификатор данного модуля  
 --- номер блока КТС  
 --- наименование модуля  
 --- значение группы параметров КТС

HwParameterGroup ::= **SEQUENCE** {  
   group-name **UTF8String (SIZE (1 .. 512))**,  
   module-parameters ModuleParameters  
 }  
 --- наименование группы параметров для КТС  
 --- перечень параметров для КТС

HwParameterGroups ::= **SEQUENCE OF** HwParameterGroup

SormSoftwareModules ::= **SEQUENCE OF** SormSoftwareModule

SormHardwareModules ::= **SEQUENCE OF** SormHardwareModule

--- тип сообщения "ответ на команду управления ИС СОПМ"

ManagementResponse ::= **CHOICE** {  
   get-structure [0] GetStructureResponse,  
   модулей СПО  
   get-module-config [1] GetModuleConfigResponse,  
   set-module-config [2] SetModuleConfigResponse,  
   check-module [3] CheckModuleResponse,  
   get-module-types [4] GetModuleTypesResponse  
 }  
 --- ответ на запрос получения структуры ИС СОПМ - КТС и  
 --- ответ на запрос получения конфигурации КТС/модуля СПО  
 --- ответ на запрос изменения конфигурации КТС/модуля СПО  
 --- ответ на запрос получения состояния модуля  
 --- ответ на запрос получения типов модулей КТС и СПО

--- ответ на запрос получения структуры ИС СОПМ - КТС и модулей СПО

GetStructureResponse ::= **SEQUENCE** {  
   sw-modules SormHardwareModules,  
 }  
 --- перечень всех узлов КТС

```

hw-modules SormSoftwareModules          --- перечень всех модулей СПО ИС СОРМ
}

--- ответ на запрос получения конфигурации КТС/модуля СПО
GetModuleConfigResponse ::= SEQUENCE {
  sw-modules SormHardwareModules,      --- конфигурации запрошенных узлов КТС
  hw-modules SormSoftwareModules      --- конфигурации запрошенных модулей СПО ИС СОРМ
}

--- отчет на запрос изменения конфигурации КТС/модуля СПО
SetModuleConfigResponse ::= ConfiguredModule --- установленная в модуль конфигурация

--- ответ на запрос получения состояния модуля
CheckModuleResponse ::= CHOICE {
  hw-modules [0] SormHardwareModules, --- текущее состояние запрошенных модулей СПО ИС СОРМ
  sw-modules [1] SormSoftwareModules --- текущее состояние запрошенных узлов КТС
}

--- ответ на запрос получения типов модулей КТС и СПО
GetModuleTypesResponse ::= SEQUENCE OF ModuleType

ModuleType ::= SEQUENCE {
  module-type INTEGER (1 .. 512),      --- идентификатор типа модуля
  type-description UTF8String (SIZE (1 .. 128)) --- расшифровка типа модуля
}

END

```

---

### NetworkIdentifiers.asn

```

NetworkIdentifiers DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS Bunch, DataNetworkEquipment, IPAddress, IPPort, IPMask, NetworkPeerInfo, PortRange, NetworkType,
DataVoipNumber, IMProtocol, VoipProtocol, HttpMethod;

```

```

--- идентификаторы пучка

```

```

Bunch ::= CHOICE {
  gsm [0] INTEGER (0 .. 4294967295), --- идентификатор пучка для GSM-Сети
  cdma-umts [1] DataNetworkEquipment --- идентификатор пучка для W/CDMA, UMTS-сети
}

```

```

-- идентификатор оборудования сети передачи данных

```

```

DataNetworkEquipment ::= CHOICE {
  mac [0] OCTET STRING (SIZE (6)), --- MAC-адрес оконечного сетевого оборудования
  atm [1] DataNetworkATM
}

```

```

--- ATM адрес (SDH/PDH сети)

```

```

DataNetworkATM ::= SEQUENCE {
  vpi OCTET STRING (SIZE (1)), --- номер виртуального пути сети ATM (VPI)
  vci OCTET STRING (SIZE (2)) OPTIONAL --- номер виртуального канала сети ATM (VCI)
}

```

```

--- тип сети связи (стандарт)

```

```

NetworkType ::= ENUMERATED {
  not-specified(0), --- неконкретизированный стандарт
  mob-gsm(1), --- сеть мобильной связи стандарта GSM
  mob-cdma(2), --- сеть мобильной связи стандарта CDMA
  fix-pstn(3), --- ТФoП-сеть
  data-ip(4), --- стационарные сети передачи данных
  data-srv(5), --- ТМС-службы
  data-ip-mob(6), --- мобильная сеть передачи данных
  data-ip-wifi (7), --- беспроводная сеть передачи данных стандарта WiFi
  data-ip-max (8), --- беспроводная сеть передачи данных стандарта WiMAX
  paging(9), --- персональный радиовывоз
}

```

```

    voip(10) --- сеть передачи голосовой информации посредством сети
передачи данных
}

--- IP-адрес
IPAddress ::= CHOICE {
    ipv4 [0] IPV4Address, --- IPv4-адрес
    ipv6 [1] IPV6Address --- IPv6-адрес
}

-- IPv4-адрес
IPV4Address ::= OCTET STRING (SIZE (4)) --- IP адрес (4 байта)

-- IPv6-адрес
IPV6Address ::= OCTET STRING (SIZE (16)) --- IP адрес (16 байт)

--- IP/UDP/TCP-порт
IPPort ::= OCTET STRING (SIZE (2)) -- порт

--- маска IP-подсети
IPMask ::= CHOICE {
    ipv4-mask [0] IPV4Mask, --- IPv4-маска
    ipv6-mask [1] IPV6Mask --- IPv6-маска
}

--- информация об участнике соединения передачи данных
NetworkPeerInfo ::= SEQUENCE {
    ip-address IPAddress, --- IP-адрес
    ip-port IPPort OPTIONAL --- IP-порт
}

PortRange ::= SEQUENCE {
    port-from [0] IPPort,
    port-to [1] IPPort
}

--- IPv4-маска
IPV4Mask ::= OCTET STRING (SIZE (4))

--- IPv6-маска
IPV6Mask ::= OCTET STRING (SIZE (16))

-- Номер телефона VoIP-абонента
DataVoipNumber ::= SEQUENCE {
    original-number UTF8String (SIZE (1 .. 128)), --- исходный номер абонента
    translated-number [0] UTF8String (SIZE (1 .. 32)) OPTIONAL, --- номер абонента после преобразования
    e164-number [1] UTF8String (SIZE (1 .. 15)) OPTIONAL --- номер абонента в E164-сети
}

VoipProtocol ::= ENUMERATED {
    sip(0),
    h323(1),
    iax(2),
    skype(100)
}
IMProtocol ::= ENUMERATED {
    icq(0), --- протокол, при помощи которого отправлены сообщения
    aol(1),
    msn(2),
    yahoo(3),
    web-mail(4), --- передача сообщения пользователем посредством почтового
веб-сервера
    skype(5),
    irc(6),
}

```

```

jabber(7),
mra(8),
tencent(9),
airway (10),
mms(98),
sms(99),
vk(100),
facebook(101),
myspace(102),
twitter(103),
odnoklassniki (104),
moymir(105),
zello(106)
}

--- http метод
HttpMethod ::= ENUMERATED {
    get(0),
    post(1),
    put(2),
    delete (3)
}

```

END

---

## ReportedIdentifiers.asn

ReportedIdentifiers **DEFINITIONS IMPLICIT TAGS ::=**  
**BEGIN**

**EXPORTS**

```

    ReportedIdentifier
;

```

**IMPORTS TAGGED,**

```

    sorm-report-identifier-pager,
    sorm-report-identifier-pstn,
    sorm-report-identifier-gsm,
    sorm-report-identifier-cdma,
    sorm-report-identifier-data-network,
    sorm-report-identifier-voip
FROM Classification

```

```

    IPAddress, DataNetworkEquipment
FROM NetworkIdentifiers;

```

```

ReportedIdentifier ::= SEQUENCE {
    id TAGGED.&id ({ReportedIdentifierVariants}),
    data TAGGED.&Data ({ReportedIdentifierVariants}@id)
}

```

--- варианты идентификаторов отчёта

```

ReportedIdentifierVariants TAGGED ::= {
    reportedPagerIdentifier
| reportedPstnIdentifier
| reportedGsmIdentifier
| reportedCdmaIdentifier
| reportedDataNetworkIdentifier
| reportedVoipIdentifier
}

```

```

--- идентификатор сети персонального радиовызова
--- идентификатор ТФОП
--- идентификатор GSM
--- идентификатор CDMA
--- идентификатор сети передачи данных

```

--- идентификатор сети персонального радиовызова

```

reportedPagerIdentifier TAGGED ::= {
    OID { sorm-report-identifier-pager }
    DATA ReportedPagerIdentifier
}

```

ReportedPagerIdentifier ::= **NumericString (SIZE (2 .. 18))**

```

--- идентификатор телефонной сети общего пользования
reportedPstnIdentifier TAGGED ::= {
  OID { sorm-report-identifier-pstn }
  DATA ReportedPstnIdentifier
}

ReportedPstnIdentifier ::= SEQUENCE {
  directory-number UTF8String (SIZE (2 .. 32)),          --- телефонный номер в международном формате
  internal-number NumericString (SIZE (1 .. 32)) OPTIONAL --- дополнительный внутренний номер, если есть
}

-- идентификатор абонента GSM
reportedGsmIdentifier TAGGED ::= {
  OID { sorm-report-identifier-gsm }
  DATA ReportedGsmIdentifier
}

ReportedGsmIdentifier ::= SEQUENCE {
  directory-number UTF8String (SIZE (2 .. 32)),          --- телефонный номер в международном формате
  imsi NumericString (SIZE (2 .. 18)),                 --- идентификатор мобильного абонента
  imei [0] NumericString (SIZE (2 .. 18)) OPTIONAL,    --- идентификатор мобильной станции
  icc [1] NumericString (SIZE (10 .. 20)) OPTIONAL,    --- идентификатор SIM-карты абонента
}

-- идентификатор абонента CDMA
reportedCdmaIdentifier TAGGED ::= {
  OID { sorm-report-identifier-cdma }
  DATA ReportedCdmaIdentifier
}

ReportedCdmaIdentifier ::= SEQUENCE {
  directory-number UTF8String (SIZE (2 .. 32)),          --- телефонный номер в международном формате
  imsi NumericString (SIZE (2 .. 18)),                 --- идентификатор мобильного абонента
  esn [0] NumericString (SIZE (2 .. 18)) OPTIONAL,    --- идентификатор мобильной станции
  min [1] NumericString (SIZE (2 .. 18)) OPTIONAL,    --- идентификатор мобильного абонента (CDMA)
  icc [2] NumericString (SIZE (10 .. 20)) OPTIONAL,    --- идентификатор SIM-карты абонента
}

reportedDataNetworkIdentifier TAGGED ::= {
  OID { sorm-report-identifier-data-network }
  DATA ReportedDataNetworkIdentifier
}

ReportedDataNetworkIdentifier ::= SEQUENCE {
  user-equipment [0] DataNetworkEquipment OPTIONAL,   --- идентификатор пользовательского оборудования
  login [1] UTF8String (SIZE (1 .. 128)) OPTIONAL,    --- имя пользователя - login
  ip-address [2] IPAddress OPTIONAL,                 --- IP адрес
  e-mail [3] UTF8String (SIZE (1 .. 128)) OPTIONAL,   --- адрес электронной почты
  pin [4] NumericString (SIZE (2 .. 20)) OPTIONAL,    --- PIN
  phone-number [5] UTF8String (SIZE (2 .. 32)) OPTIONAL, --- номер телефона
  user-domain [6] UTF8String (SIZE (2 .. 128)) OPTIONAL, --- пользовательский домен
  reserved [7] UTF8String (SIZE (1 .. 128)) OPTIONAL --- резерв
}

reportedVoipIdentifier TAGGED ::= {
  OID { sorm-report-identifier-voip }
  DATA ReportedVoipIdentifier
}

ReportedVoipIdentifier ::= SEQUENCE {
  ip-address IPAddress,                               --- IP-адрес абонента
  originator-name UTF8String (SIZE (1 .. 32)),       --- общедоступное имя инициатора связи
  calling-number UTF8String (SIZE (1 .. 32))         --- номер вызывающего абонента
}

```

END

## ReportsAbonents.asn

ReportsAbonents **DEFINITIONS IMPLICIT TAGS ::=**  
**BEGIN**

**EXPORTS** AbonentsReport;

**IMPORTS TAGGED,**

sorm-report-abonent-abonent,  
sorm-report-abonent-service,  
sorm-report-abonent-person,  
sorm-report-abonent-organization

**FROM** Classification

DateAndTime

**FROM** Sorm

TelcoID

**FROM** Dictionaries

ReportedIdentifier

**FROM** ReportedIdentifiers

NetworkType

**FROM** NetworkIdentifiers

AddressType,  
ReportedAddresses,  
ReportedAddress

**FROM** Addresses

Location

**FROM** Locations;

AbonentsReport ::= **SEQUENCE** {

id TAGGED.&id ({ AbonentsReportVariants }),  
data TAGGED.&Data({ AbonentsReportVariants } { @id })  
}

AbonentsReportVariants TAGGED ::= {

reportAbonent | --- информация об абоненте  
reportService --- информация об сервисах  
}

reportAbonent TAGGED ::= {

OID sorm-report-abonent-abonent  
DATA **SEQUENCE OF** AbonentsRecord  
}

AbonentsRecord ::= **SEQUENCE** {

telco-id TelcoID, --- идентификатор оператора связи или филиала  
idents ReportedIdentifier, --- идентификаторы абонента  
contract-date DateAndTime, --- дата и время заключения договора  
contract **UTF8String** (SIZE (1 .. 64)), --- номер договора  
actual-from DateAndTime, --- дата и время начала интервала времени, на котором  
актуальна информация  
actual-to DateAndTime, --- дата и время окончания интервала времени, на котором  
актуальна информация  
abonent AbonentInfo, --- информация об абоненте (клиенте оператора связи)  
status ActiveStatus, --- текущий статус абонента (подключен/отключен)  
attach [0] DateAndTime **OPTIONAL**, --- дата и время подключения основной услуги (если  
подключение производилось на интервале актуальности)  
detach [1] DateAndTime **OPTIONAL**, --- дата и время отключения основной услуги (если отключение  
производилось на интервале актуальности)  
last-location [2] Location **OPTIONAL**, --- последнее зафиксированное местоположение для  
мобильных абонентов  
services [3] ActiveServices **OPTIONAL**, --- активированные услуги  
line-data [4] LineData **OPTIONAL**, --- линейные данные (кросс, рамка, пара и т.д.)  
standard [5] Standard **OPTIONAL**, --- стандарт связи

addresses [6] ReportedAddresses <b>OPTIONAL</b>	---	адреса абонента
}		
ActiveServices ::= <b>SEQUENCE OF</b> AbonentService	---	набор подключенных ДБО
ActiveStatus ::= <b>ENUMERATED</b> {		
active (0),	---	активный
not-active (1)	---	не активный
}		
LineData ::= <b>SEQUENCE</b> {		
object [0] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)) <b>OPTIONAL</b> ,	---	описание объекта связи
cross [1] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)) <b>OPTIONAL</b> ,	---	описание кросса
block [2] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)) <b>OPTIONAL</b> ,	---	описание блока
pair [3] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)) <b>OPTIONAL</b> ,	---	описание пары
reserved [4] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)) <b>OPTIONAL</b>	---	резерв
}		
AbonentInfo ::= <b>SEQUENCE</b> {		
id TAGGED.&id ({ AbonentsInfoVariants },		
data TAGGED.&Data({ AbonentsInfoVariants }{ @id })		
}		
AbonentsInfoVariants TAGGED ::= {		
abonentPerson	---	информация об абоненте - физическом лице
abonentOrganization	---	информация об абоненте - юридическом лице
}		
---		Физическое лицо
abonentPerson TAGGED ::= {		
OID { sorm-report-abonent-person }		
DATA AbonentPerson		
}		
AbonentPerson ::= <b>SEQUENCE</b> {		
name-info PersonNameInfoReport,	---	ФИО
birth-date GeneralizedTime <b>OPTIONAL</b> ,	---	дата рождения
passport-info PassportInfoReport,	---	паспортные данные
bank [1] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 256)) <b>OPTIONAL</b> ,	---	банк абонента (используемый при расчетах с оператором
связи)		
bank-account [2] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 30)) <b>OPTIONAL</b>	---	счет абонента в банке (используемый при расчетах с
оператором связи)		
}		
PersonNameInfoReport ::= <b>CHOICE</b> {		
struct-name[0] PersonStructNameInfoReport,	---	структурированное ФИО
unstruct-name[1] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 1024))	---	неструктурированное ФИО
}		
PersonStructNameInfoReport ::= <b>SEQUENCE</b> {		
given-name <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)),	---	имя
initial <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128)),	---	отчество
family-name <b>UTF8String</b> ( <b>SIZE</b> (1 .. 128))	---	фамилия
}		
PassportInfoReport ::= <b>SEQUENCE</b> {		
ident-card-info IdentCardInfoReport,	---	описание удостоверения личности
doc-type-id <b>INTEGER</b> (0 .. 65535)	---	идентификатор типа документа, удостоверяющего личность
}		
IdentCardInfoReport ::= <b>CHOICE</b> {		
struct-info [0] IdentCardStructInfoReport,	---	структурированная информация
unstruct-info [1] <b>UTF8String</b> ( <b>SIZE</b> (1 .. 512))	---	неструктурированная информация
}		
IdentCardStructInfoReport ::= <b>SEQUENCE</b> {		
ident-card-serial <b>UTF8String</b> ( <b>SIZE</b> (1..16)),	---	серия удостоверения личности

```

ident-card-number  NumericString (SIZE (1..16)),
ident-card-description UTF8String (SIZE (1 .. 256))
}

abonentOrganization TAGGED ::= {
  OID { sorm-report-abonent-organization }
  DATA AbonentOrganization
}

AbonentOrganization ::= SEQUENCE {
  full-name      UTF8String (SIZE (1 .. 128)),
  inn            UTF8String (SIZE(1 .. 64)),
  contact [0]    UTF8String (SIZE (1 .. 128)) OPTIONAL,
  phone-fax [1] UTF8String (SIZE (1 .. 128)) OPTIONAL,
  internal-users [2] InternalUsers OPTIONAL,
}

заполнении
идентификаторов

bank [3]        UTF8String (SIZE(1 .. 256)) OPTIONAL,
связи)
bank-account [4] UTF8String (SIZE(1 .. 30)) OPTIONAL
оператором связи)
}

InternalUsers ::= SEQUENCE OF InternalUsersRecord
абонента –

InternalUsersRecord ::= SEQUENCE {
  user-name      UTF8String (SIZE (1 .. 64)),
  internal-number NumericString (SIZE (1 .. 8))
}

Standard ::= NetworkType --- перечень стандартов связи

reportService TAGGED ::= {
  OID sorm-report-abonent-service
  DATA SEQUENCE OF AbonentService
}

AbonentService ::= SEQUENCE {
  telco-id      TelcoID,
  service-id    INTEGER (0 .. 65535),
  idents        ReportedIdentifier OPTIONAL,
  contract      UTF8String (SIZE (1 .. 64)),
  begin-time    DateAndTime,
  end-time      DateAndTime,
  parameter     UTF8String (SIZE(1..256)) OPTIONAL
}

}

END

```

--- номер удостоверения личности  
--- когда и кем выдано

--- полное наименование  
--- ИНН  
--- контактное лицо  
--- контактные телефоны, факс  
--- список внутренних пользователей, выдается только при  
--- заполнении аналогичного поля в запросе формирования  
--- задачи на поиск идентификаторов абонентов или при

--- поля internal-number в запросе на принадлежность

--- операторам связи  
--- банк абонента (используемый при расчетах с оператором

--- счет абонента в банке (используемый при расчетах с

--- набор записей, описывающих внутренних пользователей

--- юридического лица

--- строка - описатель внутреннего пользователя  
--- внутренний номер

--- идентификатор оператора связи или филиала  
--- идентификатор услуги  
--- идентификаторы абонента  
--- номер договора  
--- дата и время начала оказания услуги  
--- дата и время окончания оказания услуги  
--- индивидуальные параметры настройки услуги абонента  
--- номер на который осуществляется преадресация

---

## Reports.asn

Reports DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS reportMessage,  
Acknowledgement;

IMPORTS TAGGED,  
sorm-message-report  
FROM Classification

```

Message, MessageID, DateAndTime
FROM Sorm

TaskID
FROM Tasks

DictionaryReport
FROM Dictionaries

ConnectionsReport
FROM ReportsConnections

LocationReport
FROM ReportsLocations

PaymentsReport
FROM ReportsPayments

PresenseReport
FROM ReportsPresense

NonFormalizedReport
FROM ReportsNonFormalized

AbonentsReport
FROM ReportsAbonents

DataContentReport
FROM ReportsDataContent;

reportMessage TAGGED ::= {
  OID { sorm-message-report }
  DATA CHOICE {
    report [0]      Report,          --- тип сообщения "отчет"
    ack [1]        Acknowledgement --- тип сообщения "подтверждение"
  }
}

-- Блок данных сообщения типа "отчет"
Report ::= SEQUENCE {
  request-id MessageID,          --- идентификатор запроса, запросивший отчет
  task-id TaskID,               --- идентификатор задачи, сгенерировавшей данный отчет
  total-blocks-number INTEGER (1 .. 1000000000000), --- общее количество блоков в отчете
  block-number INTEGER (1 .. 1000000000000), --- порядковый номер текущего блока
  report-block ReportDataBlock --- блок данных отчета
}

-- Подтверждение приёма блока, передаётся с номером сообщения соответствующему номеру сообщения блока отчёта
Acknowledgement ::= SEQUENCE {
  successful BOOLEAN,          --- признак успешного приёма блока
  broken-record INTEGER (1 .. 100000) OPTIONAL, --- номер записи в отчете, обработанной на ПУ с ошибкой
  error-description UTF8String (SIZE(1..1024)) OPTIONAL --- описание ошибки приёма блока в произвольной форме
}

ReportDataBlock ::= CHOICE {
  dictionary [0] DictionaryReport, --- отчеты задач пополнения справочников (нормативно-
  справочная информация)
  abonents [1] AbonentsReport, --- отчеты задач поисков по принадлежности абонентов
  connections [2] ConnectionsReport, --- отчеты задач поисков по соединениям абонентов
  locations [3] LocationReport, --- отчет задачи получения данных местоположения абонентов
  payments [4] PaymentsReport, --- отчеты задач поисков по совершенным платежам
  presense [6] PresenseReport, --- отчеты задач по запросу наличия в ИС СОПМ информации
  nonFormalized [7] NonFormalizedReport, --- отчеты задач по обработке неформализованных данных
  dataContent [10] DataContentReport --- отчеты задач по получению содержимого потоков
}

```

END

ReportsConnections **DEFINITIONS IMPLICIT TAGS ::=**  
**BEGIN**

**EXPORTS** ConnectionsReport, CallsRecords;

**IMPORTS** DateAndTime  
**FROM** Sorm

    DataContentID  
**FROM** Tasks

    Location  
**FROM** Locations

    ReportedIdentifier  
**FROM** ReportedIdentifiers

    Bunch,  
    NetworkPeerInfo,  
    DataNetworkEquipment,  
    DataVoipNumber,  
    IPAddress,  
    VoipProtocol,  
    IMProtocol,  
    HttpMethod  
**FROM** NetworkIdentifiers

    PhoneAbonentType,  
    TelcoID  
**FROM** Dictionaries

    TAGGED,  
    sorm-report-connection-pager,  
    sorm-report-connection-pstn,  
    sorm-report-connection-mobile,  
    sorm-report-connection-ipdr-header,  
    sorm-report-connection-aaa-login,  
    sorm-report-connection-resource,  
    sorm-report-connection-email,  
    sorm-report-connection-im,  
    sorm-report-connection-voip,  
    sorm-report-connection-file-transfer,  
    sorm-report-connection-term-access,  
    sorm-report-connection-raw-flows,  
    sorm-report-connection-address-translations  
**FROM** Classification;

ConnectionsReport ::= CallsRecords

CallsRecords ::= **SEQUENCE** {  
    id TAGGED.&id ({ReportedCallsVariants}),  
    data TAGGED.&Data ({ReportedCallsVariants} {@id})  
}

ReportedCallsVariants TAGGED ::= {  
    pagerRecord  
    | pstnRecord  
    | mobileRecord  
    | dataAAARRecord  
    | dataEmailRecord  
    | dataImRecord  
    | dataFileTransferRecord  
    | dataTermAccessRecord  
    | dataRawFlowsRecord  
    | dataResourceRecord  
    | dataVoipRecord  
    | dataAddressTranslationRecord

```

}

-- Детализированные записи отправки пэйджинг-сообщений
pagerRecord TAGGED ::= {
  OID { sorm-report-connection-pager }
  DATA SEQUENCE OF PagerRecordContent
}

PagerRecordContent ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  call-type-id  INTEGER (0 .. 4294967295), --- тип соединения
  connection-time DateAndTime,          --- дата и время передачи сообщения
  info          ReportedIdentifier,      --- идентификаторы абонента
  in-bytes-count INTEGER (0 .. 1024),    --- объем переданных данных
  term-cause    INTEGER (0 .. 16384)     --- причина завершения соединения
}

-- Детализированные записи звонков абонентов ТФОП, в т.ч. и неудавшиеся попытки соединений
pstnRecord TAGGED ::= {
  OID { sorm-report-connection-pstn }
  DATA SEQUENCE OF PstnRecordContent
}

PstnRecordContent ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  begin-connection-time DateAndTime,    --- дата и время начала соединения
  duration      INTEGER (0 .. 86399),    --- время соединения
  call-type-id  INTEGER (0 .. 4294967295), --- тип соединения
  supplement-service-id INTEGER (0 .. 4294967295), --- ДВО при соединении
  in-abonent-type PhoneAbonentType,     --- тип вызывающего абонента
  out-abonent-type PhoneAbonentType,     --- тип вызываемого абонента
  switch-id     UTF8String (SIZE (1 .. 128)), --- код коммутатора обслужившего соединение
  inbound-bunch INTEGER (0 .. 4294967295), --- входящий пучок
  outbound-bunch INTEGER (0 .. 4294967295), --- исходящий пучок
  term-cause    INTEGER (0 .. 16384),    --- причина завершения соединения

  phone-card-number [0] NumericString (SIZE (1.. 20)) OPTIONAL, --- номер телефонной карты
  in-info [1] ReportedIdentifier OPTIONAL, --- идентификаторы вызывающего абонента
  dialed-digits [2] UTF8String (SIZE (2 .. 32)), --- набранный номер вызываемого абонента
  out-info [3] ReportedIdentifier OPTIONAL, --- идентификаторы вызываемого абонента
  forwarding-identifier [4] UTF8String (SIZE (2 .. 32)) OPTIONAL, --- телефонный номер при переадресации
  border-switch-id [5] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- код пограничного коммутатора
  message [10] UTF8String OPTIONAL, --- текстовое содержание сообщения абонента
  ss7-opc [11] UTF8String (SIZE (1 .. 32)) OPTIONAL, --- SS7 код точки отправления
  ss7-dpc [12] UTF8String (SIZE (1 .. 32)) OPTIONAL, --- SS7 код точки назначения
  data-content-id [13] DataContentID OPTIONAL --- идентификатор потока
}

-- Детализированные записи звонков мобильных абонентов, в т.ч. и неудавшиеся попытки соединений
-- Должны содержать также записи об SMS, в т.ч. и неудавшиеся попытки отправки
mobileRecord TAGGED ::= {
  OID { sorm-report-connection-mobile }
  DATA SEQUENCE OF MobileRecordContent
}

MobileRecordContent ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  begin-connection-time DateAndTime,    --- дата и время начала соединения
  duration      INTEGER (0 .. 86399),    --- время соединения
  call-type-id  INTEGER (0 .. 4294967295), --- тип соединения
  supplement-service-id INTEGER (0 .. 4294967295), --- ДВО при соединении
  in-abonent-type PhoneAbonentType,     --- тип вызывающего абонента
  out-abonent-type PhoneAbonentType,     --- тип вызываемого абонента
  switch-id     UTF8String (SIZE (1 .. 128)), --- код коммутатора обслужившего соединения
  term-cause    INTEGER (0 .. 16384),    --- причина завершения соединения

  inbound-bunch [0] Bunch OPTIONAL, --- входящий пучок
  outbound-bunch [1] Bunch OPTIONAL, --- исходящий пучок
}

```

```

in-info [2]          ReportedIdentifier OPTIONAL,
in-end-location [3]  Location OPTIONAL,
ВЫЗОВА
in-begin-location [4] Location OPTIONAL,
ВЫЗОВА
out-info [5]         ReportedIdentifier OPTIONAL,
out-begin-location [6] Location OPTIONAL,
ВЫЗОВА
out-end-location [7] Location OPTIONAL,
ВЫЗОВА
forwarding-identifier [8] UTF8String (SIZE (2 .. 32)) OPTIONAL,
roaming-partner-id [9]  INTEGER (0 .. 4294967295) OPTIONAL,
border-switch-id [10]  UTF8String (SIZE (1 .. 128)) OPTIONAL,
message [40]           UTF8String OPTIONAL,
data-content-id [41]   DataContentID OPTIONAL
}

--- запись IPDR подключения/отключения абонента к сети связи
dataAAARecord TAGGED ::= {
  OID { sorm-report-connection-aaa-login }
  DATA SEQUENCE OF DataAAARecordContent
}

DataAAARecordContent ::= SEQUENCE {
  telco-id          TelcoID,
  point-id          INTEGER (0 .. 1000),
  данных, с которой получена запись о соединении
  aaa-connection-time DateAndTime,
  передачи данных
  aaa-login-type ENUMERATED {
    connect (0),
    disconnect (1),
    lac-update (2)
  },
  aaa-session-id    UTF8String (SIZE (1 .. 64)),
  aaa-allocated-ip  IPAddress,
  aaa-user-name     UTF8String (SIZE (1 .. 64)),
  aaa-connect-type  INTEGER (0 .. 65535),
  номер порта для TCP/UDP
  aaa-calling-number UTF8String (SIZE (2 .. 18)),
  aaa-called-number UTF8String (SIZE (2 .. 18)),
  aaa-nas           NetworkPeerInfo,
  aaa-in-bytes-count INTEGER (0 .. 18446744073709551615),
  aaa-out-bytes-count INTEGER (0 .. 18446744073709551615),
  aaa-user-password [0] UTF8String (SIZE (1 .. 16)) OPTIONAL,
  aaa-user-equipment [1] DataNetworkEquipment OPTIONAL,
  aaa-apn [2]        UTF8String (SIZE (1 .. 32)) OPTIONAL,
  aaa-sgsn-ip [3]    IPAddress OPTIONAL,
  aaa-ggsn-ip [4]     IPAddress OPTIONAL,
  aaa-service-area-code [5] INTEGER (0 .. 65535) OPTIONAL,
  aaa-location-start [6] Location OPTIONAL,
  (передача данных посредством подвижной сети связи)
  aaa-location-end [7] Location OPTIONAL,
  (передача данных посредством подвижной сети связи)
  phone-card-number [8] NumericString (SIZE (20)) OPTIONAL,
  aaa-imsi [9]         NumericString (SIZE (2 .. 18)) OPTIONAL,
  aaa-imei [10]       NumericString (SIZE (2 .. 18)) OPTIONAL,
  абонента
  aaa-esn [11]        NumericString (SIZE (10 .. 18)) OPTIONAL,
  абонента
  aaa-pool-number [12] UTF8String (SIZE (1 .. 64)) OPTIONAL,
  aaa-mcc [13]        UTF8String OPTIONAL,
  aaa-mnc [14]        UTF8String OPTIONAL
}

--- идентификаторы вызывающего абонента
--- местоположение вызывающего абонента на конец
--- местоположение вызывающего абонента на начало
--- идентификаторы вызываемого абонента
--- местоположение вызываемого абонента на начало
--- местоположение вызываемого абонента на конец
--- телефонный номер при переадресации
--- код роумингового партнера
--- код пограничного коммутатора
--- текстовое содержание сообщения абонента
--- идентификатор потока

--- идентификатор оператора связи или филиала
--- идентификатор точки подключения к сети передачи
--- дата и время подключения/отключения к сети
--- тип соединения
--- подключение к сети передачи данных
--- отключение от сети передачи данных

--- идентификатор сессии
--- выделенный динамический IP-адрес
--- имя пользователя (логин)
--- код протокола в соответствии с RFC1700 либо

--- вызывающий номер
--- вызываемый номер
--- IP-адрес/порт NAS-сервера
--- объем принятых данных
--- объем переданных данных
--- пользовательский пароль
--- идентификатор пользовательского оборудования
--- наименование точки доступа (Access Point Name)
--- IP-адрес GPRS/EDGE SGSN
--- IP-адрес GPRS/EDGE GGSN
--- код зоны обслуживания (SAC) GPRS/EDGE
--- базовая станция, через которую установлено соединение
--- базовая станция, через которую завершено соединение

--- номер телефонной карты
--- IMSI мобильного абонента
--- идентификатор мобильной станции
--- идентификатор мобильной станции
--- номер модемного пула

--- запись IPDR передачи почтового сообщения
dataEmailRecord TAGGED ::= {
  OID { sorm-report-connection-email }
}

```

```

DATA SEQUENCE OF DataEmailRecordContent
}

DataEmailRecordContent ::= CHOICE {
  mail-aaa [0] DataEmailRecordContentAAA,
  mail-ipdr [1] DataEmailRecordContentIPDR
}

DataEmailRecordContentIPDR ::= SEQUENCE {
  mail-cdr-header DataNetworkCdrHeader,           --- заголовок IPDR-соединения
  mail-event      EmailEvent,                     --- тип события
  mail-sender     UTF8String (SIZE (1 .. 256)),    --- отправитель почтового сообщения
  mail-receiver   EmailReceivers,                 --- получатель почтового сообщения
  mail-cc         EmailReceivers,                 --- получатель-копия почтового сообщения
  mail-subject    UTF8String (SIZE (1 .. 256)),    --- тема почтового сообщения
  mail-size       INTEGER (0 .. 4294967295),      --- размер почтового сообщения, включая прикрепленные
  файлы, байт
  attachements   INTEGER (0 ..1),                --- наличие прикрепленных файлов в письме (да/нет)
  mail-servers   EmailServers,                    --- список текстовых наименований почтовых серверов, через
  которые отправлено письмо (в т.ч. сервера веб-почты)
  mail-term-cause INTEGER (0 .. 16384),           --- причина завершения соединения

  mail-reply-to  UTF8String (SIZE (1 .. 256)) OPTIONAL, --- имя и адрес, куда следует адресовать ответы на письмо
  mail-protocol  ENUMERATED {                    --- протокол, при помощи которого отправлено сообщение
    smtp(0),
    pop3(1),
    imap(2),
    web-mail(3)
  } OPTIONAL,
  mail-abonent-id [0] UTF8String (SIZE (0 .. 64)) OPTIONAL, --- идентификатор абонента
  mail-message [10] UTF8String OPTIONAL,          --- текстовое содержание сообщения абонента
  mail-nat-info [11] SEQUENCE OF NetworkPeerInfo OPTIONAL, --- транслированные NAT IP/порт
  mail-location [12] Location OPTIONAL,          --- местоположение абонента
  mail-data-content-id [13] DataContentID OPTIONAL --- идентификатор потока
}

DataEmailRecordContentAAA ::= SEQUENCE {
  mail-cdr-header DataNetworkCdrHeader,           --- заголовок IPDR-соединения
  mail-event      EmailEvent,                     --- тип события
  mail-aaa-info   IP-AAAInformation,              --- пользовательские реквизиты входа
  mail-message [10] UTF8String OPTIONAL,          --- текстовое содержание сообщения абонента
  mail-nat-info [11] SEQUENCE OF NetworkPeerInfo OPTIONAL, --- транслированные NAT IP/порт
  mail-location [12] Location OPTIONAL,          --- местоположение абонента
  mail-data-content-id [13] DataContentID OPTIONAL --- идентификатор потока
}

EmailEvent ::= ENUMERATED
{
  email-send(1),
  email-receive(2),
  email-download(3),
  email-logon-attempt(4),
  email-logon(5),
  email-logon-failure(6),
  email-logoff(7),
  email-partial-download(8)
}

EmailReceivers ::= SEQUENCE {
  data SEQUENCE OF UTF8String (SIZE (1 .. 256))
}

EmailServers ::= SEQUENCE {
  data SEQUENCE OF UTF8String (SIZE (1 .. 256))
}

--- запись IPDR передачи мгновенных электронных сообщений между пользователями
dataImRecord TAGGED ::= {
  OID { sorm-report-connection-im }
}

```

DATA SEQUENCE OF DataImRecordContent  
}

```
DataImRecordContent ::= SEQUENCE {
  im-cdr-header      DataNetworkCdrHeader,          --- заголовок IPDR-соединения
  im-user-login      UTF8String (SIZE (1 .. 20)),    --- учетная запись пользователя при подключении
  im-user-password   UTF8String (SIZE (1 .. 16)),    --- пользовательский пароль при подключении
  im-sender-screen-name UTF8String (SIZE (1 .. 32)), --- общедоступное имя отправителя
  im-sender-uin      UTF8String (SIZE (1 .. 256)),   --- пользовательский идентификатор отправителя (в т.ч. для
  веб-чата)
  im-receivers       ImReceivers,                  --- список получателей сообщения
  im-size            INTEGER (0 .. 4294967295),     --- размер данных сессии, байт
  im-term-cause      INTEGER (0 .. 16384),          --- причина завершения соединения

  im-protocol [0]    IMProtocol OPTIONAL,
  im-abonent-id [1]  UTF8String (SIZE (0 .. 64)) OPTIONAL, --- идентификатор абонента
  im-event [5]       ImEvent OPTIONAL,
  im-message [10]    UTF8String OPTIONAL,           --- текстовое содержание сообщения абонента
  im-nat-info [11]   SEQUENCE OF NetworkPeerInfo OPTIONAL, --- транслированные NAT IP/порт
  im-location [12]   Location OPTIONAL,             --- местоположение абонента
  im-data-content-id [13] DataContentID OPTIONAL    --- идентификатор потока
}
```

ImReceivers ::= SEQUENCE OF ImReceiver

```
ImReceiver ::= SEQUENCE {
  im-receiver-screen-name UTF8String (SIZE (1 .. 256)), --- общедоступное имя получателя
  im-receiver-uin         UTF8String (SIZE (1 .. 256)) --- пользовательский идентификатор получателя (в т.ч. для веб-
  чата)
}
```

ImEvent ::= ENUMERATED

```
{
  im-undefined(0),
  im-send(1),
  im-receive(2)
}
```

--- запись IPDR передачи файловых данных

```
dataFileTransferRecord TAGGED ::= {
  OID { sorm-report-connection-file-transfer }
  DATA SEQUENCE OF DataFileTransferRecordContent
}
```

```
DataFileTransferRecordContent ::= SEQUENCE {
  file-cdr-header      DataNetworkCdrHeader,          --- заголовок IPDR-соединения
  file-server-name     UTF8String (SIZE (1 .. 256)),  --- название сервера
  file-user-name       UTF8String (SIZE (1 .. 64)),   --- имя пользователя, наименование учетной записи
  file-user-password   UTF8String (SIZE (1 .. 256)), --- пользовательский пароль
  file-server-type     BOOLEAN,                       --- режим работы файлового сервера (ACTIVE/PASSIVE)
  file-in-bytes-count  INTEGER (0 .. 18446744073709551615), --- объем переданных данных (включает как соединения
  управления так и передачи данных), байт
  file-out-bytes-count INTEGER (0 .. 18446744073709551615), --- объем принятых данных (включает как соединения
  управления так и передачи данных), байт
  file-term-cause      INTEGER (0 .. 16384),          --- причина завершения соединения
  file-abonent-id [0]  UTF8String (SIZE (0 .. 64)) OPTIONAL, --- идентификатор абонента
  file-nat-info [10]   SEQUENCE OF NetworkPeerInfo OPTIONAL, --- транслированные NAT IP/порт
  file-location [11]   Location OPTIONAL,             --- местоположение абонента
  file-data-content-id [12] DataContentID OPTIONAL    --- идентификатор потока
}
```

--- запись IPDR терминального доступа к оборудованию

```
dataTermAccessRecord TAGGED ::= {
  OID { sorm-report-connection-term-access }
  DATA SEQUENCE OF DataTermAccessRecordContent
}
```

```
DataTermAccessRecordContent ::= SEQUENCE {
  term-cdr-header      DataNetworkCdrHeader,          --- заголовок IPDR-соединения
```

```

term-in-bytes-count  INTEGER (0 .. 18446744073709551615),
соединения управления так и передачи данных), байт
term-out-bytes-count  INTEGER (0 .. 18446744073709551615),
управления так и передачи данных), байт
term-protocol  ENUMERATED {
    telnet(0),
    ssh(1),
    scp(2)
} OPTIONAL,
term-abonent-id [0]  UTF8String (SIZE (0 .. 64)) OPTIONAL,
term-nat-info [10]  SEQUENCE OF NetworkPeerInfo OPTIONAL,
term-location [11]  Location OPTIONAL,
term-data-content-id [12]  DataContentID OPTIONAL
}

```

```

--- объем переданных данных (включает как
--- объем принятых данных (включает как соединения
--- протокол удаленного доступа к оборудованию
--- идентификатор абонента
--- транслированные NAT IP/порт
--- местоположение абонента
--- идентификатор потока

```

```

--- запись IPDR прочих данных, принимаемых, получаемых пользователем при помощи закрытых протоколов обмена
dataRawFlowsRecord TAGGED ::= {
    OID { sorm-report-connection-raw-flows }
    DATA SEQUENCE OF DataRawFlowsRecordContent
}

```

```

DataRawFlowsRecordContent ::= SEQUENCE {
    flow-cdr-header  DataNetworkCdrHeader,
    flow-in-bytes-count  INTEGER (0 .. 18446744073709551615),
соединения управления так и передачи данных), байт
    flow-out-bytes-count  INTEGER (0 .. 18446744073709551615),
управления так и передачи данных), байт
    flow-protocol  ENUMERATED {
        ip(0),
        udp(1),
        tcp(2)
    } OPTIONAL,
    flow-abonent-id [0]  UTF8String (SIZE (0 .. 64)) OPTIONAL,
    flow-nat-info [10]  SEQUENCE OF NetworkPeerInfo OPTIONAL,
    flow-location [11]  Location OPTIONAL,
    flow-data-content-id [12]  DataContentID OPTIONAL
}

```

```

--- заголовок IPDR-соединения
--- объем переданных данных (включает как
--- объем принятых данных (включает как соединения
--- протокол передачи данных
--- идентификатор абонента
--- транслированные NAT IP/порт
--- местоположение абонента
--- идентификатор потока

```

```

--- запись IPDR HTTP-обращения к информационному ресурсу сети связи (сайт, портал)
dataResourceRecord TAGGED ::= {
    OID { sorm-report-connection-resource }
    DATA SEQUENCE OF DataResourceRecordContent
}

```

```

DataResourceRecordContent ::= SEQUENCE {
    res-cdr-header  DataNetworkCdrHeader,
    res-url  UTF8String (SIZE (1 .. 1024)),
    res-bytes-count  INTEGER (0 .. 18446744073709551615),
как соединения управления так и передачи данных), байт
    res-term-cause  INTEGER (0 .. 16384),
    res-aaa-info [0]  IP-AAAInformation OPTIONAL,
информационный ресурс
    res-http-method [1]  HttpMethod OPTIONAL,
    res-abonent-id [2]  UTF8String (SIZE (0 .. 64)) OPTIONAL,
    res-nat-info [10]  SEQUENCE OF NetworkPeerInfo OPTIONAL,
    res-location [11]  Location OPTIONAL,
    res-data-content-id [12]  DataContentID OPTIONAL
}

```

```

--- заголовок IPDR-соединения
--- Наименование информационного ресурса
--- объем принятых и переданных данных в соединении (включает
--- причина завершения соединения
--- пользовательские реквизиты входа в
--- http метод
--- идентификатор абонента
--- транслированные NAT IP/порт
--- местоположение абонента
--- идентификатор потока

```

```

--- запись IPDR голосовой связи посредством сети передачи данных
dataVoipRecord TAGGED ::= {
    OID { sorm-report-connection-voip }
    DATA SEQUENCE OF DataVoipRecordContent
}

```

```

DataVoipRecordContent ::= SEQUENCE {
    voip-cdr-header  DataNetworkCdrHeader,
    voip-session-id  UTF8String (SIZE (1..64)),
    voip-conference-id  UTF8String (SIZE (1..64)),

```

```

--- заголовок CDR-соединения
--- идентификатор сессии/call-id
--- идентификатор конференции

```

```

voip-duration    INTEGER (0 .. 86399),           --- длительность разговора, сек.
voip-originator-name UTF8String (SIZE (1 .. 32)), --- общедоступное имя инициатора связи
voip-service-type INTEGER (0 .. 255),           --- способ подключения
voip-calling-number DataVoipNumber,            --- номер вызывающего абонента
voip-called-number DataVoipNumber,             --- номер вызываемого абонента
voip-in-bytes-count INTEGER (0 .. 18446744073709551615), --- объем переданных данных (включает как
соединения управления так и передачи данных), байт
voip-out-bytes-count INTEGER (0 .. 18446744073709551615), --- объем принятых данных (включает как соединения
управления так и передачи данных), байт
voip-fax          BOOLEAN,                       --- была попытка передачи факсовой
информации (Т.38)
voip-term-cause   INTEGER (0 .. 16384),          --- причина завершения соединения

inbound-bunch [0] Bunch OPTIONAL,               --- входящий пучок
outbound-bunch [1] Bunch OPTIONAL,              --- исходящий пучок
voip-gateways [2] SEQUENCE OF IPAddress OPTIONAL, --- идентификаторы медиашлюзов, обслуживших
соединение

voip-protocol [3] VoipProtocol OPTIONAL,

supplement-service-id [4] INTEGER (0 .. 4294967295) OPTIONAL, --- ДВО при соединении
voip-abonent-id [5] UTF8String (SIZE (0 .. 64)) OPTIONAL, --- идентификатор абонента
voip-nat-info [10] SEQUENCE OF NetworkPeerInfo OPTIONAL, --- транслированные NAT IP/порт
voip-location [11] Location OPTIONAL,          --- местоположение абонента
voip-event [12] VoIPEvent OPTIONAL,           --- тип события
voip-data-content-id [13] DataContentID OPTIONAL
}

VoIPEvent ::= ENUMERATED {
  outgoing (0), --- исходящее
  incoming (1), --- входящее
  unknown (2) --- направление неизвестно
}

--- запись IPDR трансляции сетевых адресов
dataAddressTranslationRecord TAGGED ::= {
  OID { sorm-report-connection-address-translations }
  DATA SEQUENCE OF DataAddressTranslationRecordContent
}

DataAddressTranslationRecordContent ::= SEQUENCE {
  telco-id TelcoID, --- идентификатор оператора связи или филиала
  point-id INTEGER (0 .. 1000), --- идентификатор точки подключения к сети передачи данных,
с которой получена запись о соединении
  translation-time DateAndTime, --- дата и время трансляции
  record-type ENUMERATED { --- тип записи о трансляции сетевого адреса
    session-start (0), --- начало сессии трансляции
    session-end (1) --- окончание сессии трансляции
  },

  private-ip NetworkPeerInfo, --- внутренний адрес
  public-ip NetworkPeerInfo, --- внешний адрес
  destination-ip NetworkPeerInfo, --- адрес назначения
  translation-type ENUMERATED { --- тип трансляции сетевых адресов
    static-nat (0), --- статическая
    dynamic-nat (1), --- динамическая
    source-nat (2), --- источника
    destination-nat (3), --- получателя
    pat (4) --- адрес-порт
  }
}

--- Data network header
DataNetworkCdrHeader ::= SEQUENCE {
  id TAGGED.&id ({DataNetworkHeaderVariants}),
  data TAGGED.&Data ({DataNetworkHeaderVariants}){@id}
}

```

```
DataNetworkHeaderVariants TAGGED ::= { dataNetworkCdrHeader }
```

```
dataNetworkCdrHeader TAGGED ::= {
  OID { sorm-report-connection-ipdr-header }
  DATA DataNetworkCdrHeaderData
}
```

```
DataNetworkCdrHeaderData ::= SEQUENCE {
```

```
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  begin-connection-time DateAndTime,    --- дата и время начала соединения
  end-connection-time  DateAndTime,    --- дата и время завершения соединения
  client-info      NetworkPeerInfo,    --- информация о клиенте (IP/порт)
  server-info      NetworkPeerInfo,    --- информация о сервере (IP/порт)
  protocol-code    INTEGER (0 .. 65535), --- код протокола в соответствии с RFC1700 либо
  номер порта для TCP/UDP
  point-id        INTEGER (0 .. 1000) OPTIONAL --- идентификатор точки подключения к сети передачи
  данных, с которой получена запись о соединении
}
```

```
--- информация о входе в ресурс
```

```
IP-AAAInformation ::= SEQUENCE {
  username  UTF8String (SIZE (0..64)), --- пользовательский логин
  aaaResult IP-AAAResult OPTIONAL    --- результат операции входа
}
```

```
--- результат операции входа в ресурс
```

```
IP-AAAResult ::= ENUMERATED {
  aaaUnknown(1), --- результат неизвестен
  aaaFailed(2),  --- неудачная попытка входа
  aaaSucceeded(3) --- успешный вход
}
```

```
END
```

---

## ReportsLocations.asn

```
ReportsLocations DEFINITIONS IMPLICIT TAGS ::=
```

```
BEGIN
```

```
EXPORTS LocationReport;
```

```
IMPORTS DateAndTime
```

```
  FROM Sorm
```

```
  TelcoID
```

```
  FROM Dictionaries
```

```
  Location
```

```
  FROM Locations
```

```
  ReportedIdentifier
```

```
  FROM ReportedIdentifiers;
```

```
LocationReport ::= SEQUENCE OF ValidateLocationRecord
```

```
ValidateLocationRecord ::= SEQUENCE {
```

```
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  connection-time DateAndTime,          --- время определения местоположения
  ident         ReportedIdentifier,     --- идентификатор мобильного абонента
  connection-location Location          --- местоположение мобильного абонента
}
```

```
END
```

---

## ReportsNonFormalized.asn

```
ReportsNonFormalized DEFINITIONS IMPLICIT TAGS ::=
```

```
BEGIN
```

```
EXPORTS NonFormalizedReport;
```

**IMPORTS**

EntityId,  
 NonFormalizedEntityAttributeData  
**FROM** TasksNonFormalized

StandardInterval  
**FROM** ReportsPresense;

--- отчет задачи по обработке неформализованных данных

NonFormalizedReport ::= **CHOICE** {

nonformalized-report [0] NonFormalizedRecords,  
 nonformalized-presense [1] NonFormalizedPresenseInfo

--- отчет по задаче обработки неформализованных данных

--- отчет по наличию неформализованных данных заданного

вида

}

NonFormalizedRecords ::= **SEQUENCE OF** NonFormalizedRecord

NonFormalizedPresenseInfo ::= **SEQUENCE OF** StandardInterval  
 неформализованные данные

--- диапазоны времени за которые есть

--- запись неформализованных данных

NonFormalizedRecord ::= **SEQUENCE OF** NonFormalizedEntityAttributeData

END

---

**ReportsPayments.asn**

ReportsPayments **DEFINITIONS IMPLICIT TAGS** ::=

**BEGIN**

**EXPORTS** PaymentsReport;

**IMPORTS TAGGED,**

sorm-report-payment-bank-transaction,  
 sorm-report-payment-express-pays,  
 sorm-report-payment-terminal-pays,  
 sorm-report-payment-service-center,  
 sorm-report-payment-cross-account,  
 sorm-report-payment-telephone-card,  
 sorm-report-payment-balance-fillups,  
 sorm-report-payment-bank-division-transfer,  
 sorm-report-payment-bank-card-transfer,  
 sorm-report-payment-bank-account-transfer

**FROM** Classification

DateAndTime

**FROM** Sorm

TelcoID

**FROM** Dictionaries

Location

**FROM** Locations

ReportedIdentifier

**FROM** ReportedIdentifiers

ReportedAddress

**FROM** Addresses;

PaymentsReport ::= **SEQUENCE** {

id TAGGED.&id ({ReportedPaymentsVariants}),  
 data TAGGED.&Data ({ReportedPaymentsVariants}{@id})

}

--- варианты запрашиваемых параметров связей

ReportedPaymentsVariants TAGGED ::= {  
 bankTransactionReport

```

| expressCardReport
| publicTerminalReport
| serviceCenterReport
| crossAccountReport
| telephoneCardReport
| balanceFillupReport
| bankDivisionTransferReport
| bankCardTransferReport
| bankAccountTransferReport
}

```

--- отчет задачи на поиск пополнения баланса через банковский перевод

```

bankTransactionReport TAGGED ::= {
  OID { sorm-report-payment-bank-transaction }
  DATA SEQUENCE OF BankTransactionRecord
}

```

```

BankTransactionRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  device-id     ReportedIdentifier,     --- идентификатор абонента
  date-time-fillup DateAndTime,        --- время и дата пополнения баланса
  bank-account  UTF8String (SIZE (1 .. 64)), --- номер банковского счета, с которого совершен платеж
  bank-name     UTF8String (SIZE (1 .. 512)), --- наименование банка, со счета которого совершен перевод
  bank-address  ReportedAddress,        --- адрес банка, со счета которого совершен перевод
  amount       UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

```

--- отчет задачи на поиск пополнения баланса через карты экспресс-оплаты

```

expressCardReport TAGGED ::= {
  OID { sorm-report-payment-express-pays }
  DATA SEQUENCE OF ExpressPaysRecord
}

```

```

ExpressPaysRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  device-id     ReportedIdentifier,     --- идентификатор абонента
  date-time-fillup DateAndTime,        --- время и дата пополнения баланса
  card-number   UTF8String (SIZE (1 .. 64)), --- номер карты
  amount       UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

```

--- отчет задачи на поиск пополнения баланса через терминалы моментальных платежей

```

publicTerminalReport TAGGED ::= {
  OID { sorm-report-payment-terminal-pays }
  DATA SEQUENCE OF PublicTerminalRecord
}

```

```

PublicTerminalRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  device-id     ReportedIdentifier,     --- идентификатор абонента
  date-time-fillup DateAndTime,        --- время и дата пополнения баланса
  terminal-id   UTF8String (SIZE (1 .. 64)), --- идентификатор терминала
  terminal-number NumericString (SIZE (2 .. 20)), --- номер терминала
  terminal-address ReportedAddress,      --- адрес терминала
  amount       UTF8String (SIZE (1 .. 64)), --- сумма перевода
  location     Location OPTIONAL        --- адрес совершения платежа
}

```

--- отчет задачи на поиск пополнения баланса через центры обслуживания клиентов

```

serviceCenterReport TAGGED ::= {
  OID { sorm-report-payment-service-center }
  DATA SEQUENCE OF ServiceCenterRecord
}

```

```

ServiceCenterRecord ::= SEQUENCE {
  telco-id      TelcoID,                --- идентификатор оператора связи или филиала
  device-id     ReportedIdentifier,     --- идентификатор абонента
  date-time-fillup DateAndTime,        --- время и дата пополнения баланса
  center-id     UTF8String (SIZE (1 .. 64)), --- идентификатор центра обслуживания клиентов
}

```

```

center-address  ReportedAddress,          --- адрес центра обслуживания клиентов
amount         UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

--- отчет задачи на поиск пополнения баланса посредством снятия денег со счета другого абонента
crossAccountReport TAGGED ::= {
  OID { sorm-report-payment-cross-account }
  DATA SEQUENCE OF CrossAccountRecord
}

CrossAccountRecord ::= SEQUENCE {
  telco-id      TelcoID,          --- идентификатор оператора связи или филиала
  device-id     ReportedIdentifier, --- идентификатор абонента-получателя платежа
  date-time-fillup DateAndTime,   --- время и дата пополнения баланса
  donanted-id   ReportedIdentifier, --- идентификатор абонента-отправителя платежа
  amount       UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

--- отчет задачи на поиск пополнения баланса через телефонные карты
telephoneCardReport TAGGED ::= {
  OID { sorm-report-payment-telephone-card }
  DATA SEQUENCE OF ValidateTelephoneCardRecord
}

ValidateTelephoneCardRecord ::= SEQUENCE {
  telco-id      TelcoID,          --- идентификатор оператора связи или филиала
  activator-device-id ReportedIdentifier, --- идентификатор абонента, активировавшего карту
  date-time-fillup DateAndTime,   --- время и дата пополнения баланса
  card-number   NumericString (SIZE (2 .. 20)), --- номер телефонной карты
  amount       UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

--- отчет задачи на поиск пополнения баланса личного счета абонента
balanceFillupReport TAGGED ::= {
  OID { sorm-report-payment-balance-fillups }
  DATA SEQUENCE OF ValidateBalanceFillupRecord
}

ValidateBalanceFillupRecord ::= SEQUENCE {
  telco-id      TelcoID,          --- идентификатор оператора связи или филиала
  pay-type-id   INTEGER (0 .. 4294967295), --- тип платежа (способ оплаты)
  device-id     ReportedIdentifier, --- идентификатор абонента
  date-time-fillup DateAndTime,   --- время и дата пополнения баланса
  amount       UTF8String (SIZE (1 .. 64)), --- сумма перевода
  pay-parameters UTF8String (SIZE (1 .. 512)) OPTIONAL, --- "неструктурированная" информация
  card-number [0] NumericString (SIZE (2 .. 20)) OPTIONAL, --- номер телефонной карты
  activation-code [1] UTF8String (SIZE (2 .. 20)) OPTIONAL --- код активации телефонной карты
}

--- отчет задачи по переводу средств со счета абонента для их снятия в отделении банка
bankDivisionTransferReport TAGGED ::= {
  OID { sorm-report-payment-bank-division-transfer }
  DATA SEQUENCE OF ValidateBankDivisonTransferRecord
}

ValidateBankDivisonTransferRecord ::= SEQUENCE {
  telco-id      TelcoID,          --- идентификатор оператора связи или филиала
  donanted-id   ReportedIdentifier, --- идентификатор инициатора перевода средств
  date-time-fillup DateAndTime,   --- время и дата перевода средств
  person-recieved UTF8String(SIZE (1 .. 512)), --- получатель платежа (ФИО и прочая
неструктурированная информация)
  bank-name     UTF8String (SIZE (1 .. 256)), --- наименование банка получателя
  bank-division-name UTF8String (SIZE (1 .. 512)), --- наименование отделения банка получателя
  bank-division-address ReportedAddress, --- адрес отделения банка получателя
  amount       UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

```

```

--- отчет задачи по переводу средств со счета абонента на банковскую карту
bankCardTransferReport TAGGED ::= {
  OID { sorm-report-payment-bank-card-transfer }
  DATA SEQUENCE OF ValidateBankCardTransferRecord
}

ValidateBankCardTransferRecord ::= SEQUENCE {
  telco-id          TelcoID,          --- идентификатор оператора связи или филиала
  donated-id        ReportedIdentifier, --- идентификатор инициатора перевода средств
  bank-card-id      NumericString (SIZE (1 .. 12)), --- номер банковской карты получателя перевода
  date-time-fillup  DateAndTime,     --- время и дата перевода средств
  amount            UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

--- отчет задачи по переводу средств со счета абонента на счет в банке
bankAccountTransferReport TAGGED ::= {
  OID { sorm-report-payment-bank-account-transfer }
  DATA SEQUENCE OF ValidateBankAccountTransferRecord
}

ValidateBankAccountTransferRecord ::= SEQUENCE {
  telco-id          TelcoID,          --- идентификатор оператора связи или филиала
  donated-id        ReportedIdentifier, --- идентификатор инициатора перевода средств
  bank-name         UTF8String (SIZE (1 .. 256)), --- наименование банка получателя
  bank-account      UTF8String (SIZE (1 .. 64)), --- номер банковского счета получателя
  date-time-fillup  DateAndTime,     --- время и дата перевода средств
  amount            UTF8String (SIZE (1 .. 64)) --- сумма перевода
}

END

```

---

## ReportsPresense.asn

```

ReportsPresense DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS PresenseReport,
  StandardInterval;

```

```

IMPORTS FindRange,
  DateAndTime
FROM Sorm

```

```

  TelcoID
FROM Dictionaries

```

```

  Location
FROM Locations

```

```

  ReportedIdentifier
FROM ReportedIdentifiers

```

```

  NetworkType
FROM NetworkIdentifiers

```

```

  TAGGED,
  sorm-report-presense-abonents,
  sorm-report-presense-connections,
  sorm-report-presense-payments,
  sorm-report-presense-dictionaries,
  sorm-report-presense-locations
FROM Classification;

```

```

---отчет по запросу наличия информации
PresenseReport ::= SEQUENCE {
  id TAGGED.&id ({ReportedPresensesVariants}),
  data TAGGED.&Data ({ReportedPresensesVariants}){@id})

```

}

```
ReportedPresensesVariants TAGGED ::= {
  subsPresence
  | connectionsPresence
  | paymentsPresence
  | dictionariesPresence
  | locationPresence
}
```

--- отчет по наличию информации по абонентам и их идентификаторам.

--- Для каждого стандарта может быть указано более одного, либо ни

--- одного интервала (фактические периоды наличия информации);

```
subsPresence TAGGED ::= {
  OID { sorm-report-presense-abonents }
  DATA SEQUENCE OF StandardInterval
}
```

-- отчет по наличию информации по соединениям. Для каждого стандарта может быть указано более одного, либо ни одного, интервала (фактические периоды наличия информации);

```
connectionsPresence TAGGED ::= {
  OID { sorm-report-presense-connections }
  DATA SEQUENCE OF ConnectionsPresenseRecord
}
```

ConnectionsPresenseRecord ::= **SEQUENCE** {

standard-interval StandardInterval,

data-type **ENUMERATED** {

есть в ИС СОРМ

telephone-pstn (0),

telephone-mobile (1),

pager (2),

data-aaa (3),

data-resource (4),

(сайт, портал)

data-email (5),

data-im (6),

пользователями

data-voip (7),

data-file (8),

data-term (9),

data-raw (10),

при помощи закрытых протоколов обмена

data-address-translations (11)

}

}

--- вид соединений передачи данных, информация по которым

--- телефонные ТФОП-соединения

--- телефонные соединения мобильной связи

--- соединения сети персонального радиовызова

--- подключения/отключения абонента к сети связи

--- HTTP-обращения к информационному ресурсу сети связи

--- передача почтовых сообщений

--- передача мгновенных электронных сообщений между

--- голосовая связь посредством сети передачи данных

--- передача файловых данных

--- терминальный доступ к оборудованию

--- прочие данные, принимаемые, получаемые пользователем

--- трансляции сетевых адресов

paymentsPresense TAGGED ::= {

OID { sorm-report-presense-payments }

DATA SEQUENCE OF StandardInterval

абонентов

}

--- описание имеющейся информации по пополнениям балансов

--- отчет о наличии информации справочников в ИС. Если какой-либо из справочников не публикуется ИС СОРМ, запись о нем отсутствует

```
dictionariesPresence TAGGED ::= {
  OID { sorm-report-presense-dictionaries }
  DATA SEQUENCE OF DictionaryInfo
}
```

--- запись отчета о наличии информации справочников

DictionaryInfo ::= **SEQUENCE** {

telco-id TelcoID,

dict ObjectDescriptor,

count **INTEGER** (1 .. 4294967295),

--- идентификатор оператора связи или филиала

--- тип справочника, по которому есть информация

--- (идентификатор запроса справочника) (Requested...)

--- количество записей в справочнике

```

change-dates FindRange          --- минимальное и максимальное дата/время изменения записей
в справочнике
}

--- отчет по наличию информации по местоположению абонентов
locationPresence TAGGED ::= {
  OID { sorm-report-presense-locations }
  DATA SEQUENCE OF StandardInterval
}

--- интервал времени, на котором имеются данные по абонентам, соединениям и платежам, в рамках стандарта связи
StandardInterval ::= SEQUENCE {
  telco-id TelcoID,          --- идентификатор оператора связи или филиала
  standard NetworkType,     --- стандарт связи, информация по которому имеется в ИС
  range FindRange,         --- интервал времени, на который имеются данные в ИС
  count INTEGER OPTIONAL    --- количество записей
}

END

```

---

### RequestedAbonents.asn

```

RequestedAbonents DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS RequestedAbonent;

IMPORTS TAGGED,
  sorm-request-abonent-person,
  sorm-request-abonent-organization
  FROM Classification

  RequestedAddress FROM Addresses;

RequestedAbonent ::= SEQUENCE {
  id TAGGED.&id ({RequestedAbonentVariants}),
  data TAGGED.&Data ({RequestedAbonentVariants}){@id}
}

--- варианты запрашиваемых идентификаторов
RequestedAbonentVariants TAGGED ::= {
  requestedPerson |          --- физическое лицо
  requestedOrganization      --- юридическое лицо - организация
}

--- поля параметра запроса на физическое лицо
requestedPerson TAGGED ::= {
  OID { sorm-request-abonent-person }
  DATA RequestedPerson
}

RequestedPerson ::= SEQUENCE {
  given-name [0] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- имя
  initial [1] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- отчество
  family-name [2] UTF8String (SIZE (1 .. 128)) OPTIONAL, --- фамилия
  passport-info [3] RequestedPassport OPTIONAL, --- паспортные данные
  address [4] RequestedAddress OPTIONAL, --- адресные данные
  icc [5] NumericString (SIZE (10 .. 20)) OPTIONAL, --- идентификатор SIM-карты абонента
  contract [6] UTF8String (SIZE (1 .. 64)) OPTIONAL --- номер договора
}

--- поля паспортных данных
RequestedPassport ::= SEQUENCE {
  doc-type-id [0] INTEGER (0 .. 65535) OPTIONAL, --- идентификатор типа документа, удостоверяющего
личность
  passport-serial [1] UTF8String (SIZE (1..16)) OPTIONAL, --- серия паспорта
  passport-number [2] NumericString (SIZE (1..16)) OPTIONAL --- номер паспорта
}

```

```

-- поля параметра запроса на юридическое лицо
requestedOrganization TAGGED ::= {
  OID { sorm-request-abonent-organization }
  DATA RequestedOrganization
}

RequestedOrganization ::= SEQUENCE {
  full-name [0] UTF8String (SIZE (1 .. 128)) OPTIONAL,      --- полное наименование организации
  address [1] RequestedAddress OPTIONAL,                    --- юридический адрес организации
  inn [2] NumericString (SIZE (1 .. 64)) OPTIONAL,          --- ИНН
  internal-user [3] UTF8String (SIZE (1 .. 64)) OPTIONAL,   --- внутренний пользователь
  contract [4] UTF8String (SIZE (1 .. 64)) OPTIONAL         --- номер договора
}

```

END

---

### RequestedConnections.asn

```

RequestedConnections DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS RequestedConnection;

```

```

IMPORTS TAGGED,

```

```

  sorm-request-connection-pstn,
  sorm-request-connection-mobile,
  sorm-request-connection-entrance,
  sorm-request-connection-aaa-login,
  sorm-request-connection-resource,
  sorm-request-connection-email,
  sorm-request-connection-im,
  sorm-request-connection-voip,
  sorm-request-connection-file-transfer,
  sorm-request-connection-term-access,
  sorm-request-connection-raw-flows,
  sorm-request-connection-address-translations
  FROM Classification

```

```

  requestedPagerIdentifier,
  requestedPstnIdentifier,
  requestedGsmIdentifier,
  requestedCdmaIdentifier
  FROM RequestedIdentifiers

```

```

  PhoneAbonentType
  FROM Dictionaries

```

```

  Bunch,
  DataNetworkEquipment,
  IPAddress,
  NetworkPeerInfo,
  DataVoipNumber,
  VoipProtocol,
  IMProtocol,
  HttpMethod
  FROM NetworkIdentifiers

```

```

  Location
  FROM Locations;

```

```

RequestedConnection ::= SEQUENCE {
  id TAGGED.&id ({RequestedConnectionVariants}),
  data TAGGED.&Data ({RequestedConnectionVariants}{@id})
}

```

--- варианты запрашиваемых параметров связей

```

RequestedConnectionVariants TAGGED ::= {
  requestedPagerIdentifier
  | requestedConnectionPstn
  | requestedConnectionMobile
связи
  | requestedConnectionEntrance
  | requestedAAALogin
  | requestedResource
  | requestedEmail
  | requestedIm
  | requestedVoip
  | requestedFileTransfer
  | requestedTermAccess
  | requestedRawFlows
  | requestedAddressTranslations
}

```

--- параметры соединений абонента ТФОП  
 --- параметры соединений абонента сети мобильной  
 --- параметры соединений подключения к сети связи

```

requestedConnectionEntrance TAGGED ::= {
  OID { sorm-request-connection-entrance }
  DATA CHOICE {
    directory-number [0] UTF8String (SIZE (2 .. 32)),
    imsi [1] NumericString (SIZE (2 .. 18)),
    ip-address [2] IPAddress,
    mac [3] OCTET STRING (SIZE (6))
  }
}

```

--- телефонный номер в международном формате  
 --- идентификатор мобильного абонента  
 --- IP-адрес абонента  
 --- MAC-адрес устройства абонента

--- параметры соединений абонента ТФОП

```

requestedConnectionPstn TAGGED ::= {
  OID { sorm-request-connection-pstn }
  DATA CHOICE {
    duration [0] INTEGER (0 .. 86399),
    call-type-id [1] INTEGER (0 .. 4294967295),
    in-abonent-type [2] PhoneAbonentType,
    out-abonent-type [3] PhoneAbonentType,
    switch-id [4] UTF8String (SIZE (1 .. 128)),
    inbound-bunch [5] INTEGER (0 .. 4294967295),
    outbound-bunch [6] INTEGER (0 .. 4294967295),
    border-switch-id [7] UTF8String (SIZE (1 .. 128)),
    term-cause [8] INTEGER (0 .. 16384),
    supplement-service-id [9] INTEGER (0 .. 4294967295),
    phone-card-number [10] NumericString (SIZE (1.. 20)),
    in-info [11] RequestedConnectionPstnIdentifier,
    out-info [12] RequestedConnectionPstnIdentifier,
    forwarding-identifier [13] UTF8String (SIZE (2 .. 32)),
    message [20] UTF8String
  }
}

```

--- время соединения  
 --- тип соединения  
 --- тип вызывающего абонента  
 --- тип вызываемого абонента  
 --- код коммутатора обслужившего вызов  
 --- входящий пучок  
 --- исходящий пучок  
 --- код пограничного коммутатора  
 --- причина завершения соединения  
 --- ДВО при соединении  
 --- номер телефонной карты  
 --- идентификаторы вызывающего абонента  
 --- идентификаторы вызываемого абонента  
 --- телефонный номер при переадресации  
 --- текстовое содержание сообщения абонента

-- Идентификаторы PSTN

```

RequestedConnectionPstnIdentifier ::= SEQUENCE {
  id TAGGED.&id ({RequestedConnectionPstnIdentifierVariants}),
  data TAGGED.&Data ({RequestedConnectionPstnIdentifierVariants}){@id}
}

```

```

RequestedConnectionPstnIdentifierVariants TAGGED ::= {
  requestedPstnIdentifier
}

```

```

requestedConnectionMobile TAGGED ::= {
  OID { sorm-request-connection-mobile }
  DATA CHOICE {
    duration [0] INTEGER (0 .. 86399),
    call-type-id [1] INTEGER (0 .. 4294967295),
    supplement-service-id [2] INTEGER (0 .. 4294967295),
    in-abonent-type [3] PhoneAbonentType,
    out-abonent-type [4] PhoneAbonentType,
    switch-id [5] UTF8String (SIZE (1 .. 128)),
  }
}

```

--- время соединения  
 --- тип соединения  
 --- ДВО при соединении  
 --- тип вызывающего абонента  
 --- тип вызываемого абонента  
 --- код коммутатора обслужившего соединения

```

inbound-bunch [6]      Bunch,
outbound-bunch [7]    Bunch,
border-switch-id [8]  UTF8String (SIZE (1 .. 128)),
roaming-partner-id [9] INTEGER (0 .. 4294967295),
term-cause [10]       INTEGER (0 .. 16384),
in-info [11]          RequestedConnectionMobileIdentifier,
in-end-location [12]  Location,
in-begin-location [13] Location,
dialed-digits [14]    UTF8String (SIZE (2 .. 32)),
out-info [15]         RequestedConnectionMobileIdentifier,
out-begin-location [16] Location,
out-end-location [17] Location,
forwarding-identifier [18] UTF8String (SIZE (2 .. 32)),
message [40]          UTF8String
}
}

-- Идентификаторы мобильных абонентов
RequestedConnectionMobileIdentifier ::= SEQUENCE {
  id TAGGED.&id ({RequestedConnectionMobileIdentifierVariants}),
  data TAGGED.&Data ({RequestedConnectionMobileIdentifierVariants}){@id}
}

RequestedConnectionMobileIdentifierVariants TAGGED ::= {
  requestedGsmIdentifier |
  requestedCdmaIdentifier
}

requestedAAALogin TAGGED ::= {
  OID { sorm-request-connection-aaa-login }
  DATA CHOICE {
    point-id [0]  INTEGER (0 .. 1000),
    login-type [1] ENUMERATED {
      connect (0),
      disconnect (1),
      update (2)
    },
    user-equipment [2]  DataNetworkEquipment,
    allocated-ip [3]    IPAddress,
    user-name [4]       UTF8String (SIZE (1 .. 64)),
    user-password [5]   UTF8String (SIZE (1 .. 16)),
    connect-type [6]    INTEGER (1 .. 65535),
    calling-number [7]  UTF8String (SIZE (2 .. 18)),
    called-number [8]  UTF8String (SIZE (2 .. 18)),
    nas [9]             NetworkPeerInfo,
    apn [10]            UTF8String (SIZE (1 .. 32)),
    sgsn-ip [11]        IPAddress,
    ggsn-ip [12]        IPAddress,
    service-area-code [13] INTEGER (0 .. 65535),
    location-start [14] Location,
    location-end [15]  Location,
    phone-card-number [16] NumericString (SIZE (20)),
    imsi [17]           NumericString (SIZE (2 .. 18)),
    imei [18]           NumericString (SIZE (2 .. 18)),
    esn [19]            NumericString (SIZE (2 .. 18)),
    pool-number [20]    UTF8String (SIZE (1 .. 64)),
    mcc [21]            UTF8String,
    mnc [22]            UTF8String
  }
}

для TCP/UDP
--- или номер SMS центра если SMS
--- входящий пучок
--- исходящий пучок
--- код пограничного коммутатора
--- код роумингового партнера
--- причина завершения соединения
--- идентификаторы вызывающего абонента
--- местоположение вызывающего абонента на конец вызова
--- местоположение вызывающего абонента на начало вызова
--- набранный номер вызываемого абонента
--- идентификаторы вызываемого абонента
--- местоположение вызываемого абонента на начало вызова
--- местоположение вызываемого абонента на конец вызова
--- телефонный номер при переадресации
--- текстовое содержание сообщения абонента

--- идентификатор точки подключения к сети передачи данных,
с которой получены записи
--- тип соединения
--- подключение к сети передачи данных
--- отключение от сети передачи данных

--- идентификатор пользовательского оборудования
--- выделенный динамический IP-адрес
--- имя пользователя (логин)
--- пользовательский пароль
--- код протокола в соответствии с RFC1700 либо номер порта

--- вызывающий номер
--- вызываемый номер
--- IP-адрес/порт NAS-сервера
--- наименование точки доступа (Access Point Name)
--- IP-адрес GPRS/EDGE SGSN
--- IP-адрес GPRS/EDGE GGSN
--- код зоны обслуживания (SAC) GPRS/EDGE
--- базовая станция, через которую установлено соединение

--- базовая станция, через которую завершено соединение

--- номер телефонной карты
--- IMSI мобильного абонента
--- идентификатор мобильной станции абонента
--- идентификатор мобильной станции абонента
--- номер модемного пула

requestedResource TAGGED ::= {
  OID { sorm-request-connection-resource }
  DATA CHOICE {

```

point-id [0]	<b>INTEGER</b> (0 .. 1000),	---	идентификатор точки подключения к сети передачи данных,
с которой получены записи			
client-info [1]	NetworkPeerInfo,	---	идентификатор абонента сети передачи данных
server-info [2]	NetworkPeerInfo,	---	идентификатор сервера сети передачи данных
url [3]	<b>UTF8String</b> (SIZE (1 .. 1024)),	---	Наименование информационного ресурса
term-cause [4]	<b>INTEGER</b> (0 .. 16384),	---	причина завершения соединения
http-method [5]	HttpMethod,		
abonent-id [6]	<b>UTF8String</b> (SIZE (0 .. 64)),		
nat-info [10]	NetworkPeerInfo,	---	транслированные NAT IP/порт
location [11]	Location	---	местоположение абонента
}			
}			
requestedEmail TAGGED ::= {			
OID { sorm-request-connection-email }			
DATA CHOICE {			
point-id [0]	<b>INTEGER</b> (0 .. 1000),	---	идентификатор точки подключения к сети передачи данных,
с которой получены записи			
client-info [1]	NetworkPeerInfo,	---	идентификатор абонента сети передачи данных
server-info [2]	NetworkPeerInfo,	---	идентификатор сервера сети передачи данных
sender [3]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	отправитель почтового сообщения
receiver [4]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	получатель почтового сообщения
cc [5]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	получатель-копия почтового сообщения
subject [6]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	тема почтового сообщения
attachements [7]	<b>BOOLEAN</b> ,	---	наличие прикрепленных файлов в письме (да/нет)
mail-server [8]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	текстовое наименование сервера, через который отправлено
письмо (в т.ч. сервер веб-почты)			
term-cause [9]	<b>INTEGER</b> (0 .. 16384),	---	причина завершения соединения
abonent-id [10]	<b>UTF8String</b> (SIZE (0 .. 64)),		
message [20]	<b>UTF8String</b> ,	---	текстовое содержание сообщения абонента
nat-info [21]	NetworkPeerInfo,	---	транслированные NAT IP/порт
location [22]	Location	---	местоположение абонента
}			
}			
requestedIm TAGGED ::= {			
OID { sorm-request-connection-im }			
DATA CHOICE {			
point-id [0]	<b>INTEGER</b> (0 .. 1000),	---	идентификатор точки подключения к сети передачи данных,
с которой получены записи			
client-info [1]	NetworkPeerInfo,	---	идентификатор абонента сети передачи данных
server-info [2]	NetworkPeerInfo,	---	идентификатор сервера сети передачи данных
user-login [3]	<b>UTF8String</b> (SIZE (1 .. 20)),	---	учетная запись пользователя при подключении
user-password [4]	<b>UTF8String</b> (SIZE (1 .. 16)),	---	пользовательский пароль при подключении
sender-screen-name [5]	<b>UTF8String</b> (SIZE (1 .. 32)),	---	общедоступное имя отправителя
sender-uin [6]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	пользовательский идентификатор отправителя (в т.ч. для
веб-чата)			
receiver-screen-name [7]	<b>UTF8String</b> (SIZE (1 .. 32)),	---	общедоступное имя получателя
receiver-uin [8]	<b>UTF8String</b> (SIZE (1 .. 256)),	---	пользовательский идентификатор получателя (в т.ч. для веб-
чата)			
protocol [9]	IMProtocol,		
term-cause [10]	<b>INTEGER</b> (0 .. 16384),	---	причина завершения соединения
abonent-id [11]	<b>UTF8String</b> (SIZE (0 .. 64)),		
message [20]	<b>UTF8String</b> ,	---	текстовое содержание сообщения абонента
nat-info [21]	NetworkPeerInfo,	---	транслированные NAT IP/порт
location [22]	Location	---	местоположение абонента
}			
}			
requestedVoip TAGGED ::= {			
OID { sorm-request-connection-voip }			
DATA CHOICE {			
point-id [0]	<b>INTEGER</b> (0 .. 1000),	---	идентификатор точки подключения к сети передачи данных,
с которой получены записи			
client-info [1]	NetworkPeerInfo,	---	идентификатор абонента сети передачи данных
server-info [2]	NetworkPeerInfo,	---	идентификатор сервера сети передачи данных
duration [3]	<b>INTEGER</b> (0 .. 86399),	---	длительность разговора, сек.
originator-name [4]	<b>UTF8String</b> (SIZE (1 .. 32)),	---	общедоступное имя инициатора связи
call-type-id [5]	<b>INTEGER</b> (0 .. 4294967295),	---	способ подключения

```

voip-calling-number [6] DataVoipNumber,
voip-called-number [7] DataVoipNumber,
inbound-bunch [8] Bunch,
outbound-bunch [9] Bunch,
conference-id [10] UTF8String (SIZE (1..64)),
protocol [11] VoipProtocol,
term-cause [12] INTEGER (0 .. 16384),
abonent-id [13] UTF8String (SIZE (0 .. 64)),
nat-info [20] NetworkPeerInfo,
location [21] Location
}
}

requestedFileTransfer TAGGED ::= {
  OID { sorm-request-connection-file-transfer }
  DATA CHOICE {
    point-id [0] INTEGER (0 .. 1000),
с которой получены записи
    client-info [1] NetworkPeerInfo,
    server-info [2] NetworkPeerInfo,
    server-name [3] UTF8String (SIZE (1 .. 256)),
    user-name [4] UTF8String (SIZE (1 .. 64)),
    user-password [5] UTF8String (SIZE (1 .. 256)),
    term-cause [6] INTEGER (1 .. 16384),
    abonent-id [7] UTF8String (SIZE (0 .. 64)),
    nat-info [20] NetworkPeerInfo,
    location [21] Location
  }
}

requestedTermAccess TAGGED ::= {
  OID { sorm-request-connection-term-access }
  DATA CHOICE {
    point-id [0] INTEGER (0 .. 1000),
с которой получены записи
    client-info [1] NetworkPeerInfo,
    server-info [2] NetworkPeerInfo,
    abonent-id [3] UTF8String (SIZE (0 .. 64)),
    nat-info [10] NetworkPeerInfo,
    location [11] Location
  }
}

--- идентификаторы для соединений передачи данных (закрытые протоколы обмена)
requestedRawFlows TAGGED ::= {
  OID { sorm-request-connection-raw-flows }
  DATA CHOICE {
    point-id [0] INTEGER (0 .. 1000),
с которой получены записи
    protocol-code [1] INTEGER (0 .. 65535),
    client-info [2] NetworkPeerInfo,
    server-info [3] NetworkPeerInfo,
    abonent-id [4] UTF8String (SIZE (0 .. 64)),
    nat-info [10] NetworkPeerInfo,
    location [11] Location
  }
}

--- идентификаторы для соединений передачи данных (закрытые протоколы обмена)
requestedAddressTranslations TAGGED ::= {
  OID { sorm-request-connection-address-translations }
  DATA CHOICE {
    point-id [0] INTEGER (0 .. 1000),
с которой получена запись о соединении
    record-type [2] ENUMERATED {
      session-start (0),
      session-end (1)
    },
    private-ip [3] NetworkPeerInfo,

```

--- номер вызывающего абонента  
--- номер вызываемого абонента  
--- входящий пучок  
--- исходящий пучок  
--- идентификатор конференции  
--- причина завершения соединения  
--- транслированные NAT IP/порт  
--- местоположение абонента

--- идентификатор точки подключения к сети передачи данных,  
--- идентификатор абонента сети передачи данных  
--- идентификатор сервера сети передачи данных  
--- название сервера  
--- имя пользователя, наименование учетной записи  
--- пользовательский пароль  
--- причина завершения соединения  
--- транслированные NAT IP/порт  
--- местоположение абонента

--- идентификатор точки подключения к сети передачи данных,  
--- идентификатор абонента сети передачи данных  
--- идентификатор сервера сети передачи данных  
--- транслированные NAT IP/порт  
--- местоположение абонента

--- идентификатор точки подключения к сети передачи данных,  
--- код протокола в соответствии с RFC1700  
--- идентификатор абонента сети передачи данных  
--- идентификатор сервера сети передачи данных  
--- транслированные NAT IP/порт  
--- местоположение абонента

--- идентификатор точки подключения к сети передачи данных,  
--- тип записи о трансляции сетевого адреса  
--- начало сессии трансляции  
--- окончание сессии трансляции  
--- внутренний адрес

```

public-ip [4]    NetworkPeerInfo,          --- внешний адрес
destination-ip [5] NetworkPeerInfo,       --- адрес назначения
translation-type [6] ENUMERATED {      --- тип трансляции сетевых адресов
    static-nat (0),
    dynamic-nat (1),
    source-nat (2),
    destination-nat (3),
    pat (4)
}
}
}
END

```

---

## RequestedIdentifiers.asn

```

RequestedIdentifiers DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

### EXPORTS

```

    RequestedIdentifier,
    requestedPagerIdentifier,
    requestedPstnIdentifier,
    requestedGsmIdentifier,
    requestedCdmaIdentifier
;

```

### IMPORTS TAGGED,

```

    sorm-request-identifier-pager,
    sorm-request-identifier-pstn,
    sorm-request-identifier-gsm,
    sorm-request-identifier-cdma,
    sorm-request-identifier-data-network

```

### FROM Classification

```

    DataNetworkEquipment,
    IPAddress

```

### FROM NetworkIdentifiers

```

;

```

```

RequestedIdentifier ::= SEQUENCE {
    id TAGGED.&id ({RequestedIdentifierVariants}),
    data TAGGED.&Data ({RequestedIdentifierVariants}@id)
}

```

```

--- варианты запрашиваемых идентификаторов

```

```

RequestedIdentifierVariants TAGGED ::= {
    requestedPagerIdentifier |          --- идентификатор сети персонального
радиовызова
    requestedPstnIdentifier |         --- идентификатор ТФОП
    requestedGsmIdentifier |          --- идентификатор GSM
    requestedCdmaIdentifier |         --- идентификатор CDMA
    requestedDataNetworkIdentifier |  --- идентификатор сети передачи данных
}

```

```

--- идентификатор сети персонального радиовызова

```

```

requestedPagerIdentifier TAGGED ::= {
    OID { sorm-request-identifier-pager }
    DATA RequestedPagerIdentifier
}

```

```

RequestedPagerIdentifier ::= NumericString (SIZE (2 .. 18))

```

```

--- идентификатор телефонной сети общего пользования

```

```

requestedPstnIdentifier TAGGED ::= {
    OID { sorm-request-identifier-pstn }
    DATA RequestedPstnIdentifier
}

```

```

RequestedPstnIdentifier ::= SEQUENCE {
  directory-number UTF8String (SIZE (2 .. 32)),          --- телефонный номер в международном
  формате
  internal-number NumericString (SIZE (1 .. 32)) OPTIONAL --- дополнительный внутренний номер, если
  есть
}

-- идентификатор абонента GSM
requestedGsmIdentifier TAGGED ::= {
  OID { sorm-request-identifier-gsm }
  DATA RequestedGsmIdentifier
}

RequestedGsmIdentifier ::= CHOICE {
  directory-number [0] UTF8String (SIZE (2 .. 32)),      --- телефонный номер в международном формате
  imsi [1]        NumericString (SIZE (2 .. 18)),      --- идентификатор мобильного абонента
  imei [2]        NumericString (SIZE (2 .. 18))      --- идентификатор мобильной станции
}

-- идентификатор абонента CDMA
requestedCdmaIdentifier TAGGED ::= {
  OID { sorm-request-identifier-cdma }
  DATA RequestedCdmaIdentifier
}

RequestedCdmaIdentifier ::= CHOICE {
  directory-number [0] UTF8String (SIZE (2 .. 32)),      --- телефонный номер в международном формате
  imsi [1]        NumericString (SIZE (2 .. 18)),      --- идентификатор мобильного абонента
  esn [2]        NumericString (SIZE (2 .. 18)),      --- идентификатор мобильной станции
  min [3]        NumericString (SIZE (2 .. 18))      --- идентификатор мобильного абонента (CDMA)
}

-- Идентификатор сети передачи данных
requestedDataNetworkIdentifier TAGGED ::= {
  OID { sorm-request-identifier-data-network }
  DATA RequestedDataNetworkIdentifier
}

RequestedDataNetworkIdentifier ::= CHOICE {
  user-equipment [0] DataNetworkEquipment,            --- идентификатор пользовательского оборудования
  login [1]       UTF8String (SIZE (1 .. 128)),        --- имя пользователя - login
  ip-address [2]  IPAddress,                          --- IP адрес
  e-mail [3]     UTF8String (SIZE (1 .. 128)),        --- адрес электронной почты
  voip-phone-number [5] UTF8String (SIZE (2 .. 32))  --- номер телефона
}

END

```

---

## Sessions.asn

```

Sessions DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS sessionMessage;

IMPORTS TAGGED
    ,sorm-message-session
    FROM Classification;

sessionMessage TAGGED ::= {
  OID { sorm-message-session }
  DATA CHOICE {
    connect [0]      ConnectRequest,                  --- запрос на открытие сессии
    connect-response [1] ConnectResponse,            --- ответ на запрос открытия сессии

    adjustment [2]   AdjustmentRequest,              --- согласование поддерживаемых типов со стороны
    ПУ
    adjustment-response [3] AdjustmentResponse,      --- ответ на запрос согласования данных
  }
}

```

```

disconnect [4]      DisconnectRequest,
disconnect-response [5] DisconnectResponse
}
}

--- запрос создания сессии
ConnectRequest ::= SEQUENCE {
  session-timeout      INTEGER (60 .. 2592000),
  max-data-length      INTEGER (10 .. 100000),
  data-packet-window-size  INTEGER (4 .. 256),
  быть
  data-load-timeout    INTEGER (1 .. 60),
  request-response-timeout  INTEGER (1 .. 60),
  data-packet-response-timeout  INTEGER (1 .. 60)
отчета
}

-- ответ на запрос создания сессии
ConnectResponse ::= SEQUENCE {
  confirmed-data-packet-window-size  INTEGER (4 .. 256),
  в ConnectRequest
  confirmed-session-timeout          INTEGER (60 .. 2592000),
  переданному в ConnectRequest
  confirmed-data-load-timeout        INTEGER (1 .. 60),
отчетов
  переданному в ConnectRequest
  confirmed-request-response-timeout  INTEGER (1 .. 60),
  переданному в ConnectRequest
  supports SEQUENCE OF ObjectDescriptor
запросов, типов отчётов
}

--- согласование поддерживаемых типов со стороны ПУ
AdjustmentRequest ::= SEQUENCE {
  supports SEQUENCE OF ObjectDescriptor
отчётов
  список в сообщении ConnectRequest
}

-- ответ на согласование списка поддерживаемых типов
AdjustmentResponse ::= NULL

--- запрос завершения сессии
DisconnectRequest ::= NULL

--- ответ на запрос завершения сессии
DisconnectResponse ::= NULL

```

--- запрос на закрытие сессии  
--- ответ на запрос закрытия сессии

--- максимальное время неактивности  
--- максимальная длина блока отчета (в строках)  
--- окно канала передачи данных  
--- максимальное число блоков данных, которое может

--- отправлено без подтверждения приема  
--- таймаут начала передачи блоков отчетов  
--- таймаут ответа на запрос  
--- таймаут подтверждения приема блока данных

--- подтвержденное окно передачи данных  
--- то окно, которое может обеспечить ИС СОРМ  
--- должно быть меньше или равно окну, переданному

--- подтвержденное максимальное время неактивности  
--- должно быть больше или равно значению,

--- подтвержденный таймаут начала передачи блоков  
--- должен быть больше или равен значению,

--- подтвержденный таймаут ответа на запрос  
--- должен быть больше или равен значению,

--- весь список поддерживаемых СОРМ типов

--- список поддерживаемых ПУ типов запросов, типов  
--- данный список должен быть меньшим либо равным

--- ответ на запрос согласования данных

END

Sorm.asn

Sorm DEFINITIONS IMPLICIT TAGS ::= BEGIN

EXPORTS DateAndTime,  
FindRange,  
MessageID,  
Message;

```

IMPORTS TAGGED FROM Classification
    sessionMessage FROM Sessions
    trapMessage FROM Traps
    taskMessage FROM Tasks
    reportMessage FROM Reports
    managementMessage FROM Management
    unformattedMessage FROM Unformatted
    filterMessage FROM Filters;

```

```

Version ::= PrintableString
vers Version ::= "2.25.0" --- текущая версия протокола

-- Оболочка сообщения COPM --
Message ::= SEQUENCE {
    version Version DEFAULT vers, --- версия протокола
    message-id MessageID, --- номер запроса
    message-time DateAndTime, --- время и дата запроса
    operator-name PrintableString (SIZE (1 .. 128)) OPTIONAL, --- наименование оператора связи

    id TAGGED.&id ({SormPDUs}), --- идентификтор блока данных
    data TAGGED.&Data({SormPDUs}{@id}) --- данные блока данных
}

-- Блок данных сообщения
SormPDUs TAGGED ::= {
    sessionMessage --- сообщения организации сессии
    | trapMessage --- сообщения сигналов
    | taskMessage --- сообщения работы с задачами
    | reportMessage --- сообщения работы с отчётами
    | managementMessage --- сообщения канала передачи мониторинга (КПМ)
    | unformattedMessage --- сообщения канала передачи неформатированных данных
} (КПНФ)
| filterMessage --- сообщения установки/снятия фильтров записываемого
содержимого соединений сети передачи данных
}

--- Общие данные

-- Номер сообщения
MessageID ::= INTEGER (0 .. 4294967295)

-- Дата и время
DateAndTime ::= UTCTime

-- Диапазон поиска
FindRange ::= SEQUENCE {
    begin-find [0] DateAndTime OPTIONAL, --- время и дата начала поиска информации
    end-find [1] DateAndTime OPTIONAL --- время и дата окончания поиска информации
}

END

```

---

### TasksAbonents.asn

```

TasksAbonents DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS AbonentsTask;

```

#### **IMPORTS**

```

    LogicalOperation FROM Tasks
    RequestedIdentifier FROM RequestedIdentifiers
    RequestedAbonent FROM RequestedAbonents;

```

```

AbonentsTask ::= CHOICE {
    validate-abonents-task [0] ValidateAbonentsTask, --- задача на поиск информации о принадлежности
    --- идентификаторов абонентов сети оператора связи

    validate-identifiers [1] ValidateIdentifiersTask, --- задача на поиск информации об идентификаторах
}

```

```

зарегистрированных
validate-services [2] ValidateServicesTask
видам услуг связи
}

--- абонентов сети оператора связи
--- на физическое или юридическое лицо
--- задача на поиск информации о доступных абоненту

--- задача на поиск информации о принадлежности идентификаторов абонентов сети оператора связи
ValidateAbonentsTask ::= RequestedIdentifiers
--- запрашиваемые идентификаторы

RequestedIdentifiers ::= SEQUENCE OF RequestedIdentifierParameters
-- последовательность из идентификаторов и
логических операций

RequestedIdentifierParameters ::= CHOICE {
  separator [0] LogicalOperation,
  find-mask [1] RequestedIdentifier
}
--- логическая операция или скобка
--- параметр - идентификатор

--- задача на поиск информации об идентификаторах абонентов сети оператора связи зарегистрированных на физическое или
юридическое лицо
ValidateIdentifiersTask ::= RequestedAbonents
--- запрашиваемые абоненты

RequestedAbonents ::= SEQUENCE OF RequestedAbonentsParameters
--- последовательность логических операций и
параметров

RequestedAbonentsParameters ::= CHOICE {
  separator [0] LogicalOperation,
  find-mask [1] RequestedAbonent
}
--- логический оператор связи
--- информация запроса об абоненте

--- задача на поиск истории услуг связи абонента
--- запрос по маске запрещён, идентификатор должен быть указан полностью
ValidateServicesTask ::= SEQUENCE OF ValidateServicesParameters
ValidateServicesParameters ::= CHOICE {
  separator [0] LogicalOperation,
  find-mask [1] ValidateServicesParameter
}
--- логическая операция или скобка
--- параметр - идентификатор

ValidateServicesParameter ::= CHOICE {
  contract [0] UTF8String (SIZE (1 .. 64)),
  identifier [1] RequestedIdentifier
}
--- номер договора

```

END

---

## Tasks.asn

```

Tasks DEFINITIONS IMPLICIT TAGS ::=
BEGIN

EXPORTS taskMessage,
  TaskID,
  LogicalOperation,
  CreateTaskResponse,
  DataContentID;

IMPORTS TAGGED,
  sorm-message-task
FROM Classification

  FindRange,
  MessageID
FROM Sorm

  TelcoList,
  DictionaryTask

```

```

FROM Dictionaries

    AbonentsTask
FROM TasksAbonents

    ConnectionsTask
FROM TasksConnections

    LocationTask
FROM TasksLocation

    PaymentsTask
FROM TasksPayments

    PresenseTask
FROM TasksPresense

    DataContentTask
FROM TasksContentTask

    NonFormalizedTaskRequest,
    NonFormalizedTaskResponse
FROM
    TasksNonFormalized;

taskMessage TAGGED ::= {
  OID {sorm-message-task}
  DATA CHOICE {
    data-ready-request [0]      DataReadyRequest,           --- запрос готовности данных
    data-ready-response [1]     DataReadyResponse,         --- ответ на запрос готовности данных
    data-load-request [2]       DataLoadRequest,           --- запрос загрузки данных
    data-load-response [3]      DataLoadResponse,         --- ответ на запрос загрузки данных
    data-drop-request [4]       DataDropRequest,          --- запрос удаления данных
    data-drop-response [5]      DataDropResponse,         --- ответ на запрос удаления данных
    data-interrupt-request [6]  DataInterruptRequest,     --- запрос прерывания загрузки данных
    data-interrupt-response [7] DataInterruptResponse,    --- ответ на запрос прерывания загрузки данных
    create-task-request [8]     CreateTaskRequest,        --- запрос на создание задачи по обработке
    информации
    create-task-response [9]    CreateTaskResponse,       --- ответ на запрос создания задачи
    non-formalized-task-request [10] NonFormalizedTaskRequest, --- запрос на создание задачи по обработке
    неформализованных данных
    non-formalized-task-response [11] NonFormalizedTaskResponse --- ответ на запрос создания задачи по обработке
    неформализованных данных
  }
}

--- в этом запросе не параметров
DataReadyRequest ::= NULL

--- запрос загрузки данных конкретной задачи
DataLoadRequest ::= TaskID

--- запрос удаления данных конкретной задачи
DataDropRequest ::= TaskID

--- запрос прерывания загрузки данных
DataInterruptRequest ::= TaskID

--- запрос на создание задачи поиска
CreateTaskRequest ::= SEQUENCE {
  telcos [0]      TelcoList OPTIONAL,           --- список операторов связи
  range [1]       FindRange OPTIONAL,         --- временной диапазон поиска
  report-limit [2] INTEGER (1 .. 10000000) OPTIONAL, --- ограничение на максимальное количество
  возвращаемых записей
  task [3] CHOICE {
    dictionary [0]      DictionaryTask,          --- задачи пополнения справочников (нормативно-справочная
    информация)
    abonents [1]        AbonentsTask,           --- задачи поисков по принадлежности абонентов
    connections [2]     ConnectionsTask,       --- задачи поисков по соединениям абонентов
  }
}

```

```

        location [3]      LocationTask,          --- задача получения данных местоположения
абонентов
        payments [4]     PaymentsTask,         --- задачи поисков по совершенным платежам
        presense [6]     PresenseTask,        --- задачи предоставления сведений о наличии данных
        data-content [9]  DataContentTask      --- задачи получения содержимого потоков
    }
}

-----

-- последовательность записей о готовности данных задач
DataReadyResponse ::= SEQUENCE OF DataReadyTaskRecord

DataReadyTaskRecord ::= SEQUENCE {
    task-id TaskID,          --- идентификатор задачи
    result TaskResult       --- результат выполнения задачи
}

TaskResult ::= SEQUENCE {
    result ENUMERATED {
        data-not-ready (0), --- данные не готовы, задача еще выполняется
        data-ready (1),    --- данные есть, задача выполнена
        data-not-found (2), --- данных нет, задача выполнена
        error (3)         --- в процессе выполнения задачи произошла ошибка
    },
    report-records-number [0] INTEGER (0 .. 99999999999) OPTIONAL, --- для выполненной задачи - количество
                                                                    --- записей в отчете
    report-limit-exceeded [1] BOOLEAN OPTIONAL, --- количество записей превысило лимит, заданный при
создании задачи
    error-description [2] UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание произошедшей ошибки,
                                                                    --- если обнаружена
}

DataLoadResponse ::= SEQUENCE {
    task-id TaskID,          --- идентификатор задачи, сгенерировавшей данный
отчет
    data-exists BOOLEAN, --- признак существования результатов исполнения
задачи
                                                                    --- (есть данные или нет)
    data-blocks-number INTEGER (0 .. 99999999999) OPTIONAL, --- количество блоков в отчете
    error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если
обнаружена
}

DataDropResponse ::= SEQUENCE {
    task-id TaskID,          --- идентификатор задачи, данные которой будут
удалены
    successful BOOLEAN, --- признак успешного выполнения запроса
    error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если
обнаружена
}

DataInterruptResponse ::= SEQUENCE {
    request-id MessageID, --- идентификатор прерванного запроса загрузки
данных
    successful BOOLEAN, --- признак успешного выполнения запроса
    data-blocks-available INTEGER (0 .. 99999999999) OPTIONAL, --- количество оставшихся переданными
блоков
    error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если обнаружена
}

CreateTaskResponse ::= SEQUENCE {
    task-id TaskID OPTIONAL, --- идентификатор задачи
    successful BOOLEAN, --- признак успешного выполнения запроса
    error-description UTF8String (SIZE (1 .. 256)) OPTIONAL --- краткое описание ошибки, если
обнаружена
}

-----

```

```

-- идентификатор задачи
TaskID ::= INTEGER (0 .. 4294967295)

-- идентификатор потока
DataContentID ::= UTF8String (SIZE (1 .. 512))

LogicalOperation ::= ENUMERATED {
  operation-open-bracket (0), -- открывающая скобка - "("
  operation-close-bracket (1), -- закрывающая скобка - ")"
  operation-or (2),           -- логическое "или"
  operation-and (3),         -- логическое "и"
  operation-not (4)         -- логическое "не"
}

```

END

---

### TasksConnections.asn

```

TasksConnections DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS ConnectionsTask;

```

#### IMPORTS

```

  LogicalOperation FROM Tasks
  RequestedConnection FROM RequestedConnections
;

```

```

ConnectionsTask ::= CHOICE {

```

```

  validate-connections [0] ValidateConnectionsTask,      --- задача на поиск телефонных соединений между
  абонентами
  validate-data [1]   ValidateDataTask,                  --- задача на поиск соединений между абонентами
  сети передачи данных
  validate-entrance [2] ValidateEntranceTask             --- задача на поиск информации о появлении абонента
  в сети связь (выхода на связь)
}

```

```

-- Зачем разделение ValidateConnectionsTask ValidateDataTask

```

```

--- задача на поиск по соединениям абонентов

```

```

ValidateConnectionsTask ::= RequestedConnectionIdentifiers --- запрашиваемые идентификаторы, указываются все,
кроме RequestedDataNetworkIdentifier
ValidateDataTask ::= RequestedConnectionIdentifiers        --- запрашиваемые идентификаторы, указываются
RequestedDataNetworkIdentifier
ValidateEntranceTask ::= RequestedConnectionIdentifiers   --- задача на поиск информации о появлении абонента
в сети связь (выхода на связь)

```

```

RequestedConnectionIdentifiers ::= SEQUENCE OF RequestedConnectionParameter

```

```

RequestedConnectionParameter ::= CHOICE {

```

```

  separator [0] LogicalOperation, --- логический оператор связи
  find-mask [1] RequestedConnection --- параметр запроса
}

```

END

---

### TasksLocation.asn

```

TasksLocation DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS LocationTask;

```

#### IMPORTS

```

  LogicalOperation FROM Tasks
  RequestedConnection FROM RequestedConnections

```

```

  IPAddress
  FROM NetworkIdentifiers;

```

--- задача получения данных местоположения абонентов

LocationTask ::= RequestedLocationIdentifier --- запрашиваемые единичные идентификаторы для определения местоположения

RequestedLocationIdentifier ::=

```

CHOICE {
    directory-number [0] UTF8String (SIZE (2 .. 32)), --- телефонный номер в международном формате
    imsi [1] NumericString (SIZE (2 .. 18)), --- идентификатор мобильного абонента
    ip-address [2] IPAddress, --- IP-адрес абонента
    imei [3] NumericString (SIZE (2 .. 18)) --- идентификатор мобильной станции
}

```

END

---

## TasksNonFormalized.asn

TasksNonFormalized **DEFINITIONS IMPLICIT TAGS** ::= **BEGIN**

**EXPORTS**

```

EntityId,
NonFormalizedTaskRequest,
NonFormalizedTaskResponse,
NonFormalizedEntityAttributeData;

```

**IMPORTS** TelcoList  
**FROM** Dictionaries

```

FindRange,
MessageID,
DateAndTime
FROM Sorm

```

```

Location
FROM Locations

```

```

LogicalOperation,
CreateTaskResponse
FROM Tasks

```

;

```

-- TaskID,
-- LogicalOperation;

```

```

NonFormalizedTaskRequest ::= CHOICE {
    get-entities [0] GetEntities, --- тип сообщения "запрос получения списка типов сущностей"
    get-attributes [1] GetEntityAtttributes, --- тип сообщения "запрос получения списка атрибутов
    сущности"
    validate-task [2] ValidateNonFormalizedTask, --- тип сообщения "задача поиска неформализованных данных"
    validate-presense [3] NonFormalizedPresenseTask --- тип сообщения "задача предоставления сведений о наличии
    неформализованных данных"
}

```

```

NonFormalizedTaskResponse ::= CHOICE {
    entities [0] GetEntitiesResponse, --- ответ на запрос получения списка типов сущностей
    entity-attributes [1] GetEntityAtttributesResponse, --- ответ на запрос получения списка атрибутов сущности
    validate-task [2] ValidateNonFormalizedTaskResponse, --- ответ на запрос задачи поиска неформализованных данных
    validate-presense [3] NonFormalizedPresenseTaskResponse --- ответ на запрос задач предоставления сведений о наличии
    неформализованных данных
}

```

```

--- тип сообщения "запрос получения списка типов сущностей"
GetEntities ::= NULL

```

```

--- тип сообщения "запрос получения списка атрибутов сущности"
GetEntityAtttributes ::= EntityId

```

```

--- тип сообщения "задача поиск неформализованных данных"
ValidateNonFormalizedTask ::= SEQUENCE {
  entity-id EntityId,          --- сущность для поиска по неформализованным данным
  parameters NonFormalizedParameters, --- критерии поиска по неформализованным данным
  range FindRange OPTIONAL, --- временной диапазон поиска
  report-limit INTEGER (1 .. 10000000) OPTIONAL --- ограничение на максимальное количество возвращаемых
записей
}

--- тип сообщения "задача предоставления сведений о наличии неформализованных данных"
NonFormalizedPresenseTask ::= EntityId

NonFormalizedParameters ::= SEQUENCE OF NonFormalizedParameter

NonFormalizedParameter ::= CHOICE {
  separator [0] LogicalOperation, --- логическая операция
  find-mask [1] NonFormalizedEntityCondition --- условие
}

NonFormalizedEntityCondition ::= SEQUENCE {
  attribute NonFormalizedEntityAttribute, --- атрибут сущности
  operation MathOperation, --- операция
  attribute-value NonFormalizedEntityAttributeData --- значение атрибута
}

NonFormalizedEntityAttribute ::= SEQUENCE {
  attribute-name UTF8String (SIZE (1 .. 256)), --- текстовое наименование атрибута сущности
  attribute-type AttributeType --- тип данных атрибута
}

NonFormalizedEntityAttributeData ::= CHOICE {
  datetime [0] DateAndTime, --- дата и время с точностью до секунд
  integer [1] INTEGER, --- целочисленный
  string [2] UTF8String, --- строковый
  boolean [3] BOOLEAN, --- булевый
  float [4] REAL, --- с плавающей запятой
  location [5] Location, --- местоположение
  empty [6] NULL --- пустое значение (null)
}

--- математические операции сравнения
MathOperation ::= ENUMERATED {
  equal (0), --- равно
  less (1), --- меньше
  greater (2), --- больше
  not-equal (3), --- не равно
  less-or-equal (4), --- меньше или равно
  greater-or-equal (5) --- больше или равно
}

-----
GetEntitiesResponse ::= SEQUENCE OF NonFormalizedEntity

NonFormalizedEntity ::= SEQUENCE {
  entity-id EntityId, --- уникальный идентификатор сущности
  entity-name UTF8String (SIZE (1 .. 256)) --- текстовое наименование сущности
}

GetEntityAtttributesResponse ::= SEQUENCE {
  entity-id EntityId, --- уникальный идентификатор сущности
  entity-attributes SEQUENCE OF NonFormalizedEntityAttribute --- атрибуты сущности
}

ValidateNonFormalizedTaskResponse ::= CreateTaskResponse

NonFormalizedPresenseTaskResponse ::= CreateTaskResponse

```

```

--- типы данных атрибутов
AttributeType ::= ENUMERATED {
  date-time (0), --- дата и время с точностью до секунд
  integer (1), --- целочисленный
  string (2), --- строковый
  boolean (3), --- булевый
  float (4), --- с плавающей запятой
  location (5), --- местоположение
  empty (6) --- пустое значение
}

```

```

--- Идентификатор сущности
EntityId ::= INTEGER (0 .. 4294967296)

```

```

END

```

## TasksPayments.asn

```

TasksPayments DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS PaymentsTask;

```

```

IMPORTS LogicalOperation
FROM Tasks

```

```

    RequestedConnection
FROM RequestedConnections

```

```

    RequestedIdentifier
FROM RequestedIdentifiers

```

```

    RequestedAddress
FROM Addresses

```

```

    TAGGED,
    sorm-request-payment-bank-transaction,
    sorm-request-payment-express-pays,
    sorm-request-payment-terminal-pays,
    sorm-request-payment-service-center,
    sorm-request-payment-cross-account,
    sorm-request-payment-telephone-card,
    sorm-request-payment-balance-fillups,
    sorm-request-payment-bank-division-transfer,
    sorm-request-payment-bank-card-transfer,
    sorm-request-payment-bank-account-transfer
FROM Classification;

```

```

PaymentsTask ::= SEQUENCE {
  id TAGGED.&id ({RequestedPaymentsVariants}),
  data TAGGED.&Data ({RequestedPaymentsVariants}{@id})
}

```

```

--- варианты запрашиваемых параметров связей

```

```

RequestedPaymentsVariants TAGGED ::= {
  bankTransactionTask
| expressCardTask
| publicTerminalTask
| serviceCenterTask
| crossAccountTask
| telephoneCardTask
| balanceFillupTask
| bankDivisionTransferTask
| bankCardTransferTask
| bankAccountTransferTask
}

```

--- задача на поиск пополнения баланса через банковский перевод

```
bankTransactionTask TAGGED ::= {
  OID { sorm-request-payment-bank-transaction }
  DATA RequestedBankTransactionPays
}
```

RequestedBankTransactionPays ::= **SEQUENCE OF** RequestedBankTransactionPaysParameters

--- последовательность логических операций и параметров

RequestedBankTransactionPaysParameters ::= **CHOICE** {

```
  separator [0] LogicalOperation, --- логический оператор связи
  bank-account [1] UTF8String (SIZE (1 .. 64)), --- номер банковского счета, с
  котрого совершен платеж
  bank-name [2] UTF8String (SIZE (1 .. 512)) --- наименование банка, со счета
  котрого совершен перевод
}
```

--- задача на поиск пополнения баланса через карты экспресс-оплаты

```
expressCardTask TAGGED ::= {
  OID { sorm-request-payment-express-pays }
  DATA RequestedExpressPays
}
```

RequestedExpressPays ::= **SEQUENCE OF** RequestedExpressPaysParameters

--- последовательность логических

операций и параметров

RequestedExpressPaysParameters ::= **CHOICE** {

```
  separator [0] LogicalOperation, --- логический оператор связи
  express-card [1] NumericString (SIZE (2 .. 20)) --- номер карты экспресс-оплаты
}
```

--- задача на поиск пополнения баланса через терминалы моментальных платежей

```
publicTerminalTask TAGGED ::= {
  OID { sorm-request-payment-terminal-pays }
  DATA RequestedTerminalPays
}
```

RequestedTerminalPays ::= **SEQUENCE OF** RequestedTerminalPaysParameters

--- последовательность логических

операций и параметров

RequestedTerminalPaysParameters ::= **CHOICE** {

```
  separator [0] LogicalOperation, --- логический оператор связи
  terminal-id [1] UTF8String (SIZE (1 .. 64)), --- идентификатор терминала
  terminal-number [2] NumericString (SIZE (2 .. 20)), --- номер терминала
  terminal-address [3] RequestedAddress --- адрес терминала
}
```

--- задача на поиск пополнения баланса через центры обслуживания клиентов (ЦОК)

```
serviceCenterTask TAGGED ::= {
  OID { sorm-request-payment-service-center }
  DATA RequestedServiceCenterPays
}
```

RequestedServiceCenterPays ::= **SEQUENCE OF** RequestedServiceCenterPaysParameters

--- последовательность логических

операций и параметров

RequestedServiceCenterPaysParameters ::= **CHOICE** {

```
  separator [0] LogicalOperation, --- логический оператор связи
  center-id [1] UTF8String (SIZE (1 .. 64)), --- идентификатор центра обслуживания клиентов
  center-address [2] RequestedAddress --- адрес центра обслуживания клиентов
}
```

--- задача на поиск пополнения баланса посредством снятия денег со счета другого абонента

```
crossAccountTask TAGGED ::= {
  OID { sorm-request-payment-cross-account }
  DATA RequestedCrossAccountPays
}
```

```

RequestedCrossAccountPays ::= SEQUENCE OF RequestedCrossAccountPaysParameters    --- последовательность логических
операций и параметров
RequestedCrossAccountPaysParameters ::= CHOICE {
  separator [0] LogicalOperation,          --- логический оператор связи
  source-identifier [1] RequestedIdentifier, --- идентификатор абонента, со счета которого
переводятся средства
  dest-identifier [2] RequestedIdentifier    --- идентификатор абонента, на счет которого
переводятся средства
}

--- задача на поиск пополнения баланса через телефонные карты
telephoneCardTask TAGGED ::= {
  OID { sorm-request-payment-telephone-card }
  DATA RequestedTelephoneCardPays
}

RequestedTelephoneCardPays ::= SEQUENCE OF RequestedTelephoneCardPaysParameters --- последовательность логических
операций и параметров
RequestedTelephoneCardPaysParameters ::= CHOICE {
  separator [0] LogicalOperation,          --- логический оператор связи
  card-number [1] NumericString (SIZE (2 .. 20)) --- номер телефонной карты
}

--- общая задача на поиск пополнения баланса личного счета абонента
balanceFillupTask TAGGED ::= {
  OID { sorm-request-payment-balance-fillups }
  DATA RequestedBalanceFillups          --- параметры запроса
}

RequestedBalanceFillups ::= SEQUENCE OF RequestedBalanceFillupsParameters --- последовательность логических операций
и параметров
RequestedBalanceFillupsParameters ::= CHOICE {
  separator [0] LogicalOperation,          --- логический оператор связи
  identifier [1] RequestedIdentifier       --- идентификатор абонента
}

--- задача на поиск перевода средств со счета абонента для их снятия в отделении банка
bankDivisionTransferTask TAGGED ::= {
  OID { sorm-request-payment-bank-division-transfer }
  DATA RequestedBankDivisionTranferPays --- параметры запроса
}

RequestedBankDivisionTranferPays ::= SEQUENCE OF RequestedTranferParameters --- последовательность логических операций
и параметров

--- задача на поиск перевода средств со счета абонента на банковскую карту
bankCardTransferTask TAGGED ::= {
  OID { sorm-request-payment-bank-card-transfer }
  DATA RequestedBankCardTranferPays     --- параметры запроса
}

RequestedBankCardTranferPays ::= SEQUENCE OF RequestedTranferParameters --- последовательность логических операций
и параметров

--- задача на поиск перевода средств со счета абонента на счет в банке
bankAccountTransferTask TAGGED ::= {
  OID { sorm-request-payment-bank-account-transfer }
  DATA RequestedBankAccountTranferPays  --- параметры запроса
}

RequestedBankAccountTranferPays ::= SEQUENCE OF RequestedTranferParameters --- последовательность логических операций
и параметров

```

```

RequestedTransferParameters ::= CHOICE {
  separator [0] LogicalOperation,          --- логический оператор связи
  source-identifier [1] RequestedIdentifier --- идентификатор абонента инициатора перевода
}

```

END

---

### TasksPresense.asn

```

TasksPresense DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS PresenseTask;

```

#### IMPORTS

```

  TAGGED,
  sorm-request-presense
  FROM Classification;

```

```

PresenseTask ::= SEQUENCE {
  id TAGGED.&id ({PresenseListVariants}),
  data TAGGED.&Data ({PresenseListVariants}@id)
}

```

```

PresenseListVariants TAGGED ::= { presenseInfo }

```

```

presenseInfo TAGGED ::= {
  OID { sorm-request-presense }
  DATA ENUMERATED {
    subscribers (0), --- запрос наличия информации по абонентам и их идентификаторам
    connections (1), --- запрос наличия информации по соединениям
    payments (2),    --- запрос наличия имеющейся информации по платежам
    dictionaries (3), --- запрос наличия справочников
    locations (4)    --- запрос наличия информации по местоположениям абонентов
  }
}

```

END

---

### Traps.asn

```

Traps DEFINITIONS IMPLICIT TAGS ::=
BEGIN

```

```

EXPORTS trapMessage;

```

#### IMPORTS

```

  TAGGED,
  sorm-message-trap
  FROM Classification

```

```

  MessageID
  FROM Sorm
;

```

```

trapMessage TAGGED ::= {
  OID { sorm-message-trap }
  DATA CHOICE {
    trap [0] Trap,          --- тип сообщения "сигнал"
    trap-ack [1] TrapAck   --- тип сообщения "подтверждение сигнала"
  }
}

```

```

-- Блок данных сообщения типа "сигнал"

```

```

Trap ::= SEQUENCE {
  trap-type TrapType,      -- тип сообщения
  trap-message UTF8String (SIZE (1 .. 256)) OPTIONAL, -- описание сообщения
  reference-message MessageID OPTIONAL, -- номер сообщения к которому относится данный сигнал
}

```

```

прерывании передачи)
}

TrapType ::= ENUMERATED {
  heartbeat (0),           -- тестовый пакет
  restart-software (1),   -- перезапуск ПО
  unauthorized-access (2), -- попытка несанкционированного доступа
  critical-error (3),     -- критическая ошибка ПО, потеря данных, дальнейшая работа
  невозможна
  major-error (4),       -- серьезная ошибка ПО, потеря данных, но дальнейшая работа
  возможна
  minor-error (5)        -- незначительная ошибка ПО, данные не потеряны,
  дальнейшая работа возможна
}

-- Блок данных сообщения типа "подтверждение сигнала"
-- Номер сообщения TrapAck должен соответствовать номеру сообщения Trap
TrapAck ::= NULL

```

END

---

### Unformatted.asn

Unformatted **DEFINITIONS IMPLICIT TAGS** ::=

**BEGIN**

**EXPORTS** unformattedMessage;

**IMPORTS TAGGED,**

    sorm-message-unformatted

**FROM** Classification

    TelcoList

**FROM** Dictionaries

    Acknowledgement

**FROM** Reports

    CallsRecords

**FROM** ReportsConnections

    DateAndTime,

    MessageID

**FROM** Sorm;

unformattedMessage TAGGED ::= {

    OID { sorm-message-unformatted }

    DATA CHOICE {

        request [0] RawRequest,

        response [1] RawResponse,

        report [2] RawReport,

        report-ack [3] RawAcknowledgement

    }

}

RawRequest ::= **SEQUENCE** {

    telcos TelcoList,

    --- список операторов связи

    raw-task RawRequestTask

    --- запрос получения неформатированных данных

}

RawRequestTask ::= **CHOICE** {

    data-types-request [0] DataTypesRequest,

    --- запрос проверки наличия вида неформатированных данных в ИС

    COPM

    data-start-request [1] DataStartRequest,

    --- запрос на начало передачи неформатированных данных

    data-stop-request [2] DataStopRequest

    --- запрос на остановку передачи неформатированных данных

}

```

--- типы данных, передаваемых ИС СОРМ
RawDataType ::= ENUMERATED {
  data-reports (0),          --- записи о соединениях в формате сообщений кпд2
  raw-cdr (1),              --- "сырые" CDR-файлы
  raw-ip (2)                --- записанные сессии сетевого трафика в TCPDUMP-формате
}

ControlCommandType ::= ENUMERATED {
  start (0),                --- команда начала передачи ИС СОРМ данных
  stop (1)                  --- команда прерывания передачи ИС СОРМ данных
}

DataTypesRequest ::= RawDataType

DataStartRequest ::= SEQUENCE {
  time-from DateAndTime,    --- начало временного периода в буфере, с которого необходимо получить данные
  time-to DateAndTime,      --- конец временного периода в буфере, с которого необходимо получить данные
  raw-type RawDataType      --- тип неформатированных данных передачи
}

DataStopRequest ::= NULL

RawResponse ::= CHOICE {
  data-types-response [0]   DataTypesResponse,    --- ответ на запрос проверки наличия неформатированных вида данных в
  ИС СОРМ
  data-start-response [1]   DataStartResponse,    --- ответ на запрос начала передачи неформатированных данных
  data-stop-response [2]   DataStopResponse      --- ответ на запрос остановки передачи неформатированных данных
}

DataTypesResponse ::= SEQUENCE {
  successful BOOLEAN,       --- признак наличия в ИС СОРМ запрошенного вида неформатированных данных
  selected-type RawDataType, --- выбранный вид данных для передачи
  time-from DateAndTime,    --- начало временного периода в буфере, начиная с которого накоплены данные
  time-to DateAndTime      --- конец временного периода в буфере, по которому накоплены данные
}

DataStartResponse ::= BOOLEAN --- признак успешности выполнения команды
DataStopResponse ::= BOOLEAN --- признак успешности выполнения команды

RawReport ::= SEQUENCE {
  request-id MessageID,     --- идентификатор запроса
  stream-id UTF8String (SIZE (1 .. 256)), --- идентификатор потока в сессии
  total-blocks-number INTEGER (0 .. 999999999999), --- общее количество блоков в отчете
  block-number INTEGER (1 .. 1000000000000), --- порядковый номер текущего блока
  report-block RawDataBlock --- блок данных отчета
}

RawDataBlock ::= CHOICE {
  reports [0] CallsRecords, --- записи отчетов о соединениях в формате КПД
  raw-cdr [1] RawBytesBlock
}

RawBytesBlock ::= SEQUENCE OF RawBytes
RawBytes ::= OCTET STRING (SIZE (1 .. 4096))

RawAcknowledgement ::= Acknowledgement

```

END

---

## ReportsDataContent.asn

ReportsDataContent DEFINITIONS IMPLICIT TAGS ::=

**BEGIN**

**EXPORTS** DataContentReport;

**IMPORTS TAGGED,**

sorm-report-data-content-raw  
**FROM** Classification;

```
DataContentReport ::= SEQUENCE {
  id TAGGED.&id ({ReportedDataContentVariants}),
  data TAGGED.&Data ({ReportedDataContentVariants}{@id})
}
```

```
ReportedDataContentVariants TAGGED ::= {
  reportDataContentRaw
}
```

```
reportDataContentRaw TAGGED ::= {
  OID { sorm-report-data-content-raw }
  DATA SEQUENCE {
    successful          BOOLEAN,          --- признак успешного формирования блока данных
    data [0]           OCTET STRING (SIZE (1 .. 1048576)) OPTIONAL, --- содержимое блока (в случае если блок успешно
сформирован)
    error [1]          UTF8String (SIZE (1 .. 4096)) OPTIONAL, --- описание ошибки (в случае если не удалось
сформировать блок)
    codec-info [2]     UTF8String (SIZE (1 .. 4096)) OPTIONAL, --- описание способа кодирования в формате SDP в
соответствии RFC 2327
    direction [3]      DataContentRawDirection OPTIONAL --- направление передачи
  }
}
```

```
DataContentRawDirection ::= ENUMERATED {
  client-server (0), --- от клиента к серверу (от инициатора)
  server-client (1) --- от сервера клиенту (инициатору)
}
END
```

---

### TasksContentTask.asn

TasksContentTask **DEFINITIONS IMPLICIT TAGS** ::= **BEGIN**

**EXPORTS** DataContentTask;

**IMPORTS**

DataContentID **FROM** Tasks;

DataContentTask ::= DataContentID

**END**

---

### Filters.asn

Filters **DEFINITIONS IMPLICIT TAGS ::=**  
**BEGIN**

**EXPORTS** filterMessage;

**IMPORTS TAGGED,**  
sorm-message-filter  
**FROM** Classification

NetworkPeerInfo,  
IPMask,  
PortRange  
**FROM** NetworkIdentifiers;

filterMessage TAGGED ::= {  
  OID {sorm-message-filter}  
  DATA CHOICE {  
    create-filter-request [0] CreateFilterRequest,  
    create-filter-response [1] CreateFilterResponse,  
    drop-filter-request [2] DropFilterRequest,  
    drop-filter-response [3] DropFilterResponse  
  }  
}

CreateFilterRequest ::= **SEQUENCE** {  
  filter-id FilterID,  
  filter-data FilterData  
}

CreateFilterResponse ::= **SEQUENCE** {  
  filter-id FilterID,  
  successful **BOOLEAN**,  
  error-description **UTF8String (SIZE (1 .. 256)) OPTIONAL**  
}

DropFilterRequest ::= FilterID

DropFilterResponse ::= **SEQUENCE** {  
  filter-id FilterID,  
  successful **BOOLEAN**,  
  error-description **UTF8String (SIZE (1 .. 256)) OPTIONAL**  
}

FilterData ::= **SEQUENCE** {  
  filter-criteria FilterCriteria, --- основной критерий фильтрации  
  additional-criteria FilterCriteria **OPTIONAL** --- вспомогательный фильтр, объединяется с основным по "И"  
}

FilterCriteria ::= **CHOICE** {  
  ip-port [0] NetworkPeerInfo, --- IP адрес и опционально порт  
  network-mask [1] IPFilterMask, --- маска IP/сети  
  port-range [2] PortRange, --- диапазон TCP/UDP портов  
  vlan [3] **INTEGER (SIZE (0 .. 4096))**, --- номер VLAN  
  mac [4] **OCTET STRING (SIZE (6))**, --- MAC-адрес  
  mpls-tag [5] **INTEGER**, --- MPLS-метка  
  protocol-code [6] **INTEGER (0 .. 65535)** --- код протокола в соответствии с RFC1700  
}

IPFilterMask ::= **SEQUENCE** {  
  mask IPMask, --- маска IP/сети  
  mask-length **INTEGER** --- длина маски  
}

FilterID ::= **INTEGER**

**END**

---

Приложение № 10  
к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающих выполнение установленных действий при проведении оперативно-розыскных мероприятий

**Соответствие накапливаемой ИС БД ОРМ видов информации о соединении, справочной информации и лицензий на оказание услуг связи**

1. Группы видов информации о соединениях:
  - 1.1. Группа 1 (Г1) включает:
    - 1) в фиксированной телефонной связи;
  - 1.2. Группа 2 (Г2) включает:
    - 1) в подвижной телефонной связи;
    - 2) изменения местоположения.
  - 1.3. Группа 3 (Г3) включает:
    - 3) подключения/отключения абонента к сети передачи данных (ААА);
    - 4) НТТР-соединения с информационными ресурсами сети передачи данных;
    - 5) передачу почтовых e-mail сообщений;
    - 6) электронные сообщения между пользователями;
    - 7) голосовая информация посредством сети передачи данных;
    - 8) передача файловых данных;
    - 9) терминальный доступ к оборудованию;
    - 10) передача прочих видов данных;
    - 11) трансляция сетевых адресов;
    - 12) изменение местоположения.
2. Группы видов справочной информации:
  - 2.1. Группа 1 (С1) включает:
    - 13) операторы связи, обслуживаемые ИС БД ОРМ;
    - 14) пучки соединительных линий;
    - 15) коммутаторы;
    - 16) типы вызовов;
    - 17) список услуг связи;
    - 18) способы оплаты (пополнения баланса);
    - 19) причины завершения соединения;
    - 20) план телефонной номерной емкости;
    - 21) специальные номера оператора связи;
    - 22) типы документов, удостоверяющих личность;
    - 23) карта связей пучков соединительных линий;

2.2. Группа 2 (С2) включает:

- 1) операторы связи, обслуживаемые ИС БД ОРМ;
- 2) пучки соединительных линий;
- 3) коммутаторы;
- 4) типы вызовов;
- 5) список услуг связи;
- 6) способы оплаты (пополнения баланса);
- 7) причины завершения соединения;
- 8) план телефонной номерной емкости;
- 9) специальные номера оператора связи;
- 10) типы документов, удостоверяющих личность;
- 11) карта связей пучков соединительных линий;
- 12) базовые станции;
- 13) план нумерации идентификаторов мобильных телефонных абонентов;
- 14) роуминговые партнеры;

2.3. Группа 3 (С3) включает:

- 1) операторы связи, обслуживаемые ИС БД ОРМ;
- 2) справочник с планами IP-адресации;
- 3) идентификаторы точек подключения к сети передачи данных;
- 4) IP шлюзы;
- 5) специальные номера оператора связи;
- 6) типы документов, удостоверяющих личность;
- 7) способы оплаты (пополнения баланса);
- 8) причины завершения соединения;
- 9) список услуг связи.

В Таблице № 1 приведены виды накапливаемой информации в ИС БД ОРМ и соответствующие им виды лицензий на оказание услуг связи.

Таблица № 1. Виды накапливаемой информации в ИС БД ОРМ и соответствующие им виды лицензий на оказание услуг связи.

№	Наименование услуг связи	Виды накапливаемой информации о соединениях	Виды справочной информации
1	услуги местной телефонной связи, за исключением услуг местной телефонной связи с использованием таксофонов и средств коллективного доступа	Г1	С1
2	услуги междугородной и международной телефонной связи	Г1	С1
3	услуги телефонной связи в выделенной сети связи	Г1	С1
4	услуги внутризонавой	Г1	С1

	телефонной связи		
5	услуги местной телефонной связи с использованием таксофонов	Г1	С1
6	услуги местной телефонной связи с использованием средств коллективного доступа	Г2	С1
7	услуги подвижной радиосвязи в сети связи общего пользования	Г2	С2
8	услуги подвижной радиосвязи в выделенной сети связи	Г2	С2
9	услуги подвижной радиотелефонной связи в сети связи общего пользования	Г2	С2
10	услуги подвижной спутниковой радиосвязи	Г2	С2
11	услуги связи по предоставлению каналов связи	Г3 за исключением информации о смене местоположения	С3
12	услуги связи в сети передачи данных, за исключением передачи голосовой информации	Г3 за исключением передачи голосовой информации посредством сети передачи данных	С3
13	услуги связи по передаче голосовой информации в сети передачи данных	голосовая информация посредством сети передачи данных	С3
14	телематические услуги связи	Г3 за исключением: – голосовая информация посредством сети передачи данных; – изменение местоположения.	С3

## Приложение № 11

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

**Требования к интерфейсам точек консолидации информации передаваемой в телефонных сетях и сетях передачи данных для сбора текстовых сообщений пользователей, голосовой информации, изображений, звуков, иных сообщений пользователей**

1. Интерфейсы с использованием контроля несущей и обнаружением коллизий, включая:

оптические интерфейсы 10GBASE-S;  
оптические интерфейсы 10GBASE-L;  
оптические интерфейсы 10GBASE-E;  
оптические интерфейсы 10GBASE-LX4;  
электрические интерфейсы 10GBASE-CX4;  
оптические интерфейсы 1000 BASE-X;  
электрический интерфейс GBE;  
оптические интерфейсы 100BASE-X;  
электрические интерфейсы 100BASE-T;  
оптические интерфейсы 10BASE-F;  
электрические интерфейсы EtherNet.

2. Оптические интерфейсы оборудования синхронной цифровой иерархии (SDH):

интерфейс 1-го уровня SDH (STM-1);  
интерфейс 4-го уровня SDH (STM-4);  
интерфейс 16-го уровня SDH (STM-16);  
интерфейс 64-го уровня SDH (STM-64).

3. Оптические интерфейсы оборудования плезиохронной цифровой иерархии (PDH):

интерфейс 34 Мбит/с (E3);  
интерфейс 140 Мбит/с (E4).

4. Электрические интерфейсы оборудования плезиохронной (PDH) и синхронной (SDH) цифровых иерархий:

интерфейс 2 Мбит/с (E1);  
интерфейс 34 Мбит/с (E3);  
интерфейс 140 Мбит/с (E4);

интерфейс 155 Мбит/с (STM-1).

5. Электрических интерфейсов оборудования передачи данных,  
включая:

интерфейс V.24/V.28;

интерфейс X.21/V.11;

интерфейс V.35/V.28;

интерфейс V.36/V.11.

---

## Приложение № 12

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

**Требования к форматам записываемой информации текстовых сообщений пользователей, голосовой информации, изображений, звуков, иных сообщений пользователей услугами связи**

1. Для текстовых сообщений пользователей – в кодировке UTF-8 (RFC 3629). В случае если в силу передачи по сети связи сообщение было разбито на несколько фрагментов текстовое сообщение записывается и хранится в виде единого целого.

2. Для голосовой информации – в соответствии с фактически использованной сетью связи кодированием (кодеком) при доставке/приеме абонентом голосовой информации без внесения изменений и ухудшения качества голосовой информации, в виде RTP потока (RFC 3550). При использовании протоколов сигнализации SIP, MGCP или других, использующих сообщения SDP для описания параметров передачи медиаданных, передаются атрибуты m и a из сообщения с типом application/sdp (описание сессии) согласно RFC 2327. Для остальных протоколов сигнализации также передаются атрибуты описания сессии m и a в формате, описанном RFC 2327, но сформированные на стороне TC CORM.

Пример значения поля Value (кавычки не включаются):

```
“m=audio 49230 RTP/AVP 96 97 98
```

```
a=rtpmap:96 L8/8000
```

```
a=rtpmap:97 L16/8000
```

```
a=rtpmap:98 L16/11025/2”
```

3. Для изображений, звуков, иных сообщениям пользователей услугами связи – в виде прикладной сессии протоколов – в виде потока информации прикладного уровня (собранных сессий прикладного уровня (ГОСТ Р ИСО/МЭК 7498-1-99) протоколов TCP/UDP, RFC 793 и RFC 768 соответственно).

4. Для факсимильных сообщений - в формате TIFF, в т.ч. многостраничного.

---

## Приложение № 13

к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

**Требования к организации точек консолидации содержимого текстовых сообщений пользователей, голосовой информации, изображений, звуков, иных сообщений пользователей услугами связи**

1. ИС БД ОРМ выполняет сбор информации содержимого текстовых сообщений пользователей, голосовой информации, иных сообщений пользователей услугами из точек консолидации.

2. Требования к точкам консолидации в соответствии с Таблицей № 1. Таблица № 1. Виды лицензий на оказание услуг связи и соответствующие им требования к организации точек консолидации информации.

№	Наименование услуг связи	Требования к организации точек консолидации информации
1	услуги связи по предоставлению каналов связи	Интерфейсы оборудования связи точек пропуска входящего и исходящего трафика с предоставленных каналов связи
2	услуги связи в сети передачи данных, за исключением передачи голосовой информации	Интерфейсы оборудования связи включающие полный объем входящего и исходящего сетевого трафика с абонентских устройств, в т.ч. межабонентский трафик до выполнения преобразования (трансляции) сетевых адресов абонентов
3	телематические услуги связи	Интерфейсы оборудования связи с сетевым трафиком содержащим полный объем принимаемой и доставляемой на телематический сервис информации от пользователей а также сетевой трафик обмена информацией с другими телематическими сервисами

Оператор связи организывает подачу в точки консолидации информации всей доступной в сети информации.

Приложение № 14  
к Правилам применения технических и программных средств информационных систем, содержащих базы данных абонентов оператора связи и предоставленных им услугах связи, обеспечивающие выполнение установленных действий при проведении оперативно-розыскных мероприятий

Справочно

### Список использованных сокращений и терминов

АПК	–	Аппаратно-программный комплекс
БД	–	База данных
Время выполнения задачи в ИС БД ОРМ	–	Временной интервал между началом исполнения поисковой задачи в ИС БД ОРМ и временем завершения формирования результата в ИС БД ОРМ
ИС БД ОРМ	–	Информационные системы, содержащие базы данных операторов связи, необходимые для обеспечения проведения оперативно-розыскных мероприятий на сетях электросвязи
Источник информации	–	Аппаратно-программные средства оператора связи, обеспечивающие автоматический или автоматизированный ввод в ИС БД ОРМ информации о пользователях услуг связи, предоставленных им услугах связи, платежах и иной информации, подлежащей накоплению
КТС	–	Комплекс технических средств
НСД	–	Несанкционированный доступ
ПО	–	Программное обеспечение
ПУ		Пульт управления
Сессия (сеанс связи)		Любое авторизованное подключение к сети связи, в рамках которого пользователь получает услуги связи
Соединение		Взаимодействие между средствами связи, позволяющее абоненту передавать и/или получать данные
Несостоявшееся соединение и попытка соединения		Любое регистрируемое сетью связи действие, совершенное пользователем услуг связи при обращении к сети связи, которое не привели к

установлению соединения/сеанса связи.

К несостоявшимся соединениям и попыткам соединения относятся:

1) для услуг телефонной связи:

- а) набор неполного или не корректный набор номера телефона вызываемого абонента;
- б) недоступность вызываемого абонента в сети;
- в) сброс вызова или неответ вызываемым абонентом;
- г) вызываемый абонент находился в состоянии разговора на момент поступления нового вызова;
- д) сбой сети связи в ходе установления соединения;

2) для услуг, предоставляемых в сети передачи данных:

- невозможность и/или отказ вызываемой стороны принять входящий вызов для установления соединения любого уровня.

Прочие виды соединений абонентов/пользователей посредством услуг сети передачи данных

Недекодированные соединения, получаемые и передаваемые при помощи закрытых протоколов обмена (не относящихся к информации о: подключениях/отключениях абонентов к сети передачи данных, HTTP-соединениям с информационными ресурсами, передачи почтовых e-mail сообщений, электронным сообщениям между пользователями, голосовой связи посредством сети передачи данных, передачи файловых данных, терминальному доступу к оборудованию для удаленного управления, трансляции сетевых адресов пользователей, информации об изменении местоположения абонентов)

---