ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО/МЭК 19831— 2017

МОДЕЛЬ И ПРОТОКОЛ ИНТЕРФЕЙСА УПРАВЛЕНИЯ ОБЛАЧНОЙ ИНФРАСТРУКТУРОЙ (CIMI)

Интерфейс для управления облачной инфраструктурой

(ISO/IEC 19831:2015, IDT)

Издание официальное

Москва Стандартинформ 2017

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Обществом с ограниченной ответственностью «Информационно-аналитический вычислительный центр» (ООО ИАВЦ) на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 22 «Информационные технологии»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 22 февраля 2017 г. № 68-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/МЭК 19831:2015 «Модель и протокол интерфейса управления облачной инфраструктурой (CIMI). Интерфейс для управления облачной инфраструктурой» (ISO/IEC 19831:2015 «Cloud Infrastructure Management Interface (CIMI) Model and RESTful HTTP-based Protocol — An Interface for Managing Cloud Infrastructure», IDT).

ИСО/МЭК 19831 разработан подкомитетом 38 «Платформы и сервисы для распределенных приложений» Совместного технического комитета JTC 1 «Информационные технологии» Международной организации по стандартизации (ISO) и Международной электротехнической комиссии (IEC).

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

5 ВВЕДЕН ВПЕРВЫЕ

6 Некоторые положения международного стандарта, указанного в пункте 4, могут являться объектом патентных прав. ИСО и МЭК не несут ответственности за идентификацию подобных патентных прав

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© Стандартинформ, 2017

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
1.1 Структура стандарта 1
1.2 Схема управления версиями стандарта 1
1.3 Типографские соглашения 1
2 Нормативные ссылки 2
3 Термины и определения 3
4 Протокол, основанный на HTTP 4
4.1 Введение 4
4.2 Операции протокола 11
4.3 Поддержка OVF 17
5 Модель 17
5.1 Обертки Ресурсов 17
5.2 Расширяемость 18
5.3 Идентификаторы 18
5.4 Ограничения атрибута 18
5.5 Типы данных и их сериализация 19
5.6 Единицы 26
5.7 Семантика отношений 26
5.8 Операции 27
5.9 Альтернативные форматы модели 27
5.10 Ресурсы 27
5.11 Метаданные Ресурса 29
5.12 Точка входа в облако 39
5.13 Ресурс System и отношения 44
5.14 Ресурс Machine и отношения 66
5.15 Ресурсы Volume и отношения 98
5.16 Сетевые ресурсы и отношения 109
5.17 Ресурсы мониторинга и отношения 138
6 Вопросы безопасности 165
Приложение А (обязательное) Поддержка OVF в CIMI 166
Приложение В (справочное) XML-схема 167
Приложение С (справочное) История изменений 168
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов национальным стандартам 169

Введение

Настоящий стандарт подготовлен Рабочей группой по управлению облачными вычислениями DMTF[1)]. Стандарт определяет логическую модель для менеджмента ресурсов в категории «Инфраструктура как услуга».

[1)] DMTF — некоммерческая ассоциация, объединяющая представителей промышленности для продвижения средств управления предприятиями и системами и обеспечения совместимости.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

МОДЕЛЬ И ПРОТОКОЛ ИНТЕРФЕЙСА УПРАВЛЕНИЯ ОБЛАЧНОЙ ИНФРАСТРУКТУРОЙ (CIMI)

Интерфейс для управления облачной инфраструктурой

Cloud infrastructure management interface (CIMI) model and RESTful HTTP-based protocol. An interface for managing cloud infrastructure

Дата введения — 2018—01—01

1 Область применения

Настоящий стандарт распространяется на модель и протокол для административных взаимодействий между поставщиком службы облачных вычислений категории «Инфраструктура как услуга» (далее — служба IaaS) и потребителем службы IaaS. В стандарте представлены модели основных ресурсов службы IaaS (машины, хранилище и сеть) для обеспечения административного доступа потребителя к реализации службы IaaS и содействия переносимости между реализациями, соответствующими указанным в стандарте. Стандарт определяет протокол в стиле REST над HTTP. Однако базовая модель не зависит от HTTP, поэтому существует возможность отобразить ее на другие протоколы.

CIMI определяет управление жизненным циклом инфраструктуры, предоставляемой Поставщиком. CIMI не распространяется за пределы управления инфраструктурой на управление приложениями и службами, которые Потребитель выполняет на инфраструктуре, предоставленной как служба Поставщиком. Хотя CIMI может в некоторой степени применяться к другим моделям службы облачных вычислений, таким как «Платформа как услуга» (PaaS) или «Хранение как услуга» (SaaS), эти сценарии использования не рассматривались при проектировании CIMI.

1.1 Структура стандарта

В настоящем стандарте определены модель и протокол, основанный на RESTful HTTP.

Сначала определяются базовые шаблоны REST, а после определения каждого ресурса для него определяется любая дополнительная информация, специфичная для HTTP.

1.2 Схема управления версиями стандарта

Схема управления версиями — согласно 6.3 DMTF DSP4004.

Поскольку в стандарте могут быть внесены изменения в течение времени, некоторые характеристики могут устареть. Каждая из этих характеристик обозначена в пункте настоящего стандарта, в котором она определена, и не должна поддерживаться.

1.3 Типографские соглашения

В настоящем стандарте применены следующие соглашения:

В тексте стандарта:

- основной текст набран шрифтом Arial;
- выделены существительные CIMI, такие как наименования ресурсов, атрибутов, операций;
- текст, обозначающий наименование или значение определенного понятия (например, атрибут «value constraints», графа «Наименование Ресурса», значение «false»), взят в кавычки. В таких случаях текст в кавычках всегда относится к понятию, экземпляром которого он является;
- наименования понятий CIMI, которые представляют собой общепринятые слова, но имеют конкретное значение в CIMI, набраны обычным шрифтом, но начинаются с прописной буквы, например Поставщик, Потребитель, Ресурс, Набор.

Издание официальное

Слова, имеющие общепринятые смысловые значения, набраны со строчной буквы. Однако модельные понятия CIMI, имеющие общепринятые смысловые значения, набраны строчными буквами, например атрибут, операция.

В таблицах, описывающих модель данных Ресурсов:

- шрифт основного изложения стандарта используется для всех терминов, поскольку структура таблицы позволяет отличить их от обычного текста;
- текстовые описания набраны с применением правил изложения основного текста стандарта;
- имена, служащие заполнителями для фактических имен, которые могут меняться в каждом экземпляре модели, взяты в угловые скобки <> (например, <componentTemplate>).

При описании сериализации Ресурсов используется нотация псевдосхемы с применением следующих соглашений:

- значения, выделенные курсивом, указывают на типы данных, а не литералы;
- к элементам добавляются символы для указания на количество элементов:
  - «?» (0 или 1);
  - «*» (0 или более);
  - «+» (1 или более);
- вертикальный разделитель «|» обозначает альтернативу. Например «a|b» означает выбор между «a» и «b»;
- круглые скобки «(«и»)» используются для указания области действия операторов «?», «*», «+» и «|»;
- многоточие («...») указывает на точки расширяемости.

Примечание — Отсутствие многоточия не означает, что не существует возможного расширения; скорее всего оно не представлено в явном виде (как правило, для краткости).

Наименования операций Create, Update, Delete, Read обозначают абстрактные операции, которые передают семантику соответствующих конкретных операций, таких как методы HTTP или URI операций CIMI.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты. Для датированных документов используется только указанные издания (включая любые исправления или версии обновления DMTF). Для недатированных документов используются самые последние издания (с учетом всех изменений) (включая любые исправления или версии обновления DMTF).

DMTF DSP0223, Generic Operations 1.0 (Общие операции 1.0)

DMTF DSP0243, Open Virtualization Format Specification 1.1 (Спецификация открытого формата виртуализации 1.1)

DMTF DSP1001, Management Profile Specification Usage Guide 1.1 (Руководство по использованию спецификации профиля менеджмента 1.1)

DMTF DSP4004, DMTF Release Process 2.4 (Процесс подготовки публикаций DMTF 2.4)

IANA HTTP Header Registry (Реестр заголовков HTTP)

IEC 80000-13:2008, International Organization for Standardization, Geneva, Switzerland, Quantities and units – Part 13: Information science and technology (Международная организация по стандартизации, Женева, Швейцария, Величины и единицы. Часть 13: Информатика и информационная технология)

IETF RFC2616, R. Fielding et al, Hypertext Transfer Protocol — HTTP/1.1,Р. Филдинг и др. (Протокол передачи гипертекста —HTTP/1.1)

IETF RFC3986, T.Berners-Lee et al, *Uniform Resource Identifiers (URI): Generic* Syntax, Т.Бернерс-Ли и др. (Унифицированные идентификаторы ресурса (URI). Общий синтаксис)

IETF RFC4627, D. Crockford, The application/json Media Type for JavaScript Object Notation (JSON) (Д. Крокфорд, Тип медиа application/json для объектной нотации JavaScript (JSON)

IETF RFC5246, T. Dierks and E. Rescorla, The Transport Layer Security (TLS) Protocol Version 1.2 (Т. Диркси Э. Рескорла, Протокол безопасности транспортного уровня, Версия 1.2 (TLS)

ISO 8601:20044, International Organization for Standardization, Geneva, Switzerland, Data elements and interchange formats — Information interchange — Representation of dates and times (Международная организация по стандартизации, Женева, Швейцария, Элементы данных и форматы для обмена информацией. Обмен информацией. Представление дат и времени, март 2008 г.)

ISO/IEC 14977:1996, Roger S. Scowen, Extended BNF — A generic base standard. Software Engineering Standards Symposium 1993 (Информационная технология. Синтаксический метаязык. Расширенная форма Бэкуса-Наура (Extended BNF)

ISO/IEC Directives Part 2, Rules for the structure and drafting of International Standards (Директивы ИСО/МЭК. Часть 2. Правила построения и формулирования международных стандартов)

NIST 800-145,NIST Special Publication 800-145, Peter Mell and Timothy Grance, The NIST Definition of Cloud Computing (Питер Мелл и Тимоти Грэнс. Определение облачных вычислений, сентябрь 2011 г.)

NIST Special Publication 500-292, Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger and Dawn Leaf, NIST Cloud Computing Reference Architecture, NIST 500-292(Фан Лиу, Чжин Тонг, Цзянь Мао, Роберт Бон, Джон Мессина, Ли Бэджер и Дон Лиф, Эталонная архитектура облачных вычислений NIST, сентябрь 2011 г.)

Representational State Transfer, Roy Fielding, Doctoral dissertation, University of California, Architectural Styles and the Design of Network-based Software Architectures (Рой Филдинг, Стили архитектуры и проект сетевой программной архитектуры (глава 5), Докторская диссертация, Калифорнийский университет, 2000 г.)

XML Schema — Part 1, World Wide Web Consortium (W3C) Recommendation, H. Thompson, et al., Editors, XML Schema Part 1: Structures Second Edition (Рекомендация Консорциума World Wide Web (W3C), Х. Томпсон, и др. (редакторы), XML-схема часть 1: Структуры (вторая редакция), 28 октября 2004 г.)

XML Schema — Part 2, World Wide Web Consortium (W3C) Recommendation, P. Biron, A. Malhotra, Editors, XML Schema Part 2: Data types (Second Edition) (Рекомендация Консорциума World Wide Web (W3C), П. Бирон, А. Мэлхотра (редакторы), XML-схема часть 2.Типы данных (вторая редакция), 28 октября 2004 г.)

3 Термины и определения

В настоящем стандарте некоторые термины, определенные в данном разделе, помимо общепринятого смыслового значения также имеют иное значение.

Термины «должен» («требуется»), «не должен», «следует» («рекомендуется»), «не следует» («не рекомендуется»), «не нуждается» («не требуется»), «может» и «не может» в настоящем стандарте, должны интерпретироваться по [13]. Термины, приведенные в круглых скобках, являются альтернативным вариантом для термина, указанного перед скобкой для использования в исключительных случаях, когда предыдущий термин не может быть использован по лингвистическим причинам.

Примечание — В [13], приложение Н определены дополнительные альтернативы, которые в случае их использования следует интерпретировать в общепринятом значении.

Термины «пункт», «подпункт», «абзац» и «приложение» следует интерпретироваться, как описано в [13], раздел 5.

Термины «обязательный» и «справочный» в настоящем стандарте следует интерпретировать согласно [13], раздел 3. В настоящем стандарте пункты, подпункты или приложения, имеющие справочное значение, не содержат обязательных требований. Примечания и примеры всегда являются справочными элементами.

В настоящем стандарте применены термины по [4], [1] и [3], а также следующие термины с соответствующими определениями:

3.1 **аутентификация** (authentication): Процесс проверки заявления, предъявленного субъектом, о том, что он имеет право разрешить действовать от имени данного принципала (человека, службы и т.д.), типичными механизмами которой являются использование комбинации имени пользователя/пароля или пар открытый/закрытый ключ.

3.2 **авторизация** (Контроль Доступа) (authorization): Процесс проверки наличия разрешения у проверяемого принципала (человека, службы и т.д.) на выполнение определенных операций (например, чтение, обновление) на определенных Ресурсах.

3.3 **облако** (cloud): Синоним термина «облачные вычисления», определенного в разделе [14].

3.4 **Потребитель Службы облачных вычислений** (Cloud Service Consumer): Категория агентов, которая включает в себя Бизнес-менеджера Потребителя (утверждает деловые и финансовые расходы для использованных служб, счета на используемые реализации службы, устанавливает деловые отношения, устанавливает учетные записи, бюджет, условия и т.д.), Администратора службы Потребите-

ля (запрашивает реализации службы и изменения в реализациях службы, закупает службы в рамках деловых отношений, создает Пользователей служб (включая политики), выделяет ресурсы, такие как компьютер и хранилище, готовит отчеты, такие как отчет об использовании и т.д.) и Пользователи службы (использует реализации службы, предоставленные Поставщиком службы облачных вычислений). Если данное действие или деятельность включат в себя более одного из вышеуказанных агентов, то используется термин «Потребитель». В случаях, когда различие между агентами данной категории существенно, используются уточненные термины.

Примечание — Термин «Потребитель Службы облачных вычислений» эквивалентен термину «Потребитель Облачных вычислений», определенному в эталонной архитектуре NIST [15].

3.5 **Поставщик Службы облачных вычислений** (Cloud Service Provider): Категория агентов, которая включает в себя Менеджера операций Службы (управляет технической инфраструктурой, необходимой для предоставления службы облачных вычислений, проводит мониторинг и измеряет производительность и использование в соответствии с Соглашением об уровне обслуживания, предоставляет отчеты по мониторингу и измерениям и т.д.), Бизнес-менеджера Службы (предлагает все типы служб, созданных разработчиками службы облачных вычислений, учитывает услуги, потенциально предложенные самими Поставщиками службы и предлагаемые от имени разработчиков службы облачных вычислений, формирует портфель деловых отношений, устанавливает учетные записи и условия для Потребителей и т.д.) и Менеджера перехода Службы (позволяет потребителю использовать службу облачных вычислений, включая первичное подключение, интеграцию и адаптацию процесса, определяет и создает службы на основе Шаблонов и Конфигураций, которые могут использоваться Потребителями и заноситься в каталог и т.д.). Если данное действие или деятельность могут включать в себя более одного из вышеуказанных агентов, то используется термин «Поставщик». В случаях, когда различие между агентами данной категории существенно, используются уточненные термины.

3.6 **Набор** (Collection): Особый вид Ресурса, содержащий набор других Ресурсов и снабженный представлением и сериализацией в соответствии с настоящим стандартом и являющийся синонимом термина «набора CIMI».

3.7 **Конфигурация** (Configuration): Совокупность метаданных, значения которых служат параметрами отдельной конфигурации определенного типа виртуального ресурса.

3.8 **Инфраструктура как услуга; служба IaaS** (Infrastructure as a Service (IaaS)): Модель службы облачных вычислений, определенная в [14, раздел 2].

3.9 **конфиденциальность сообщения** (message confidentiality): Свойство сообщения, обеспечивающее просмотр его содержания только для конкретного(ых) получателя(ей).

3.10 **целостность сообщения** (message integrity): Свойство сообщения, позволяющее получателю этого сообщения определить, изменялось ли содержание сообщения после его создания.

3.11 **Ресурс** (Resource): Представление объекта, управляемого Поставщиком [Службы облачных вычислений], который обычно доступен Потребителю [Службы облачных вычислений] для доступа или операций с помощью интерфейса, описанного в настоящем стандарте, является синонимом термина «ресурс CIMI».

3.12 **Шаблон** (Template): Ресурс, представляющий собой совокупность метаданных и инструкций, используемых для создания экземпляра другого Ресурса (например, Machine Template используется для создания экземпляров Machine); может включать в себя другие Ресурсы метаданных, такие как другие Шаблоны, Конфигурации и Образы, например Machine Template ссылается на Machine Configuration и MachineImage является синонимом термина «шаблон CIMI».

4 Протокол, основанный на HTTP

4.1 Введение

Все операции основаны на Протоколе передачи гипертекста (HTTP), версия 1.1 [7]. Каждый запрос должен быть отправлен с помощью глаголов HTTP, например таких, как PUT, GET, DELETE, HEAD или POST, и включает в себя тело сообщения в формате JSON или XML. Каждый ответ использует стандартный код состояния HTTP, семантика которого интерпретируются в контексте конкретного выполненного запроса. У каждого Ресурса в модели есть тип MIME, который дополнительно контекстуализирует полезную нагрузку запросов и ответов.

Ресурсы в модели идентифицируются с помощью URI и представление каждого Ресурса должно содержать атрибут «ID» типа URI, который действует как «указатель на себя». Этот URI должен быть уникальным в рамках контекста реализации Поставщика. Разыменовывание (ссылка на указанный объект) (через HTTP GET) URI Ресурса приводит к представлению Ресурса, содержащего атрибуты и ссылки на связанные Ресурсы. Для того чтобы начать операцию, клиент должен знать URI главной точки входа Поставщика, также известной как Ресурс «Точка входа в облако». Тогда все остальные Ресурсы в пределах окружения должны поддаваться обнаружению путем итеративного перехода по ссылкам к связанным Ресурсам в пределах каждого полученного Ресурса.

4.1.1 Развитие протокола и ожидания клиентов

Будущие версии структуры настоящего стандарта будут изменяться таким образом, чтобы клиенты, использующие более ранние версии настоящего стандарта, могли продолжать работу и эволюция протокола не оказывала на них никакого отрицательного влияния. При этом клиенты должны знать несколько правил, чтобы гарантировать эту совместимость:

1 Клиенты должны знать, что схемы сериализации ответов в настоящем стандарте не являются полными. В частности, клиенты должны принимать ответы, содержащие смесь данных и представленных в настоящем стандарте сериализаций, и должны игнорировать такие данные. Однако в соответствии с 4.2.1.3 клиенты должны включать неизвестные данные в запросы PUT для обновления Ресурсов.

2 Клиенты не должны иметь знаний об операциях, которые поддерживает сервер. Предполагают, что они обнаружат поддерживаемые (и допустимые) операции при обходе Ресурсов от Точки входа в облако. Сериализации обнаруженных Ресурсов указывают на то, какие операции поддерживаются сервером.

4.1.2 Пространства имен XML

В таблице 1 приведены пространства имен XML, используемых в настоящем стандарте. Выбор любого префикса пространства имен произволен и не имеет семантического значения.

Таблица 1 — Пространства имен XML

Префикс	Пространства имен XML	Стандарт
cimi	gost.e-ecolog.ru	Настоящий стандарт
xs	gost.e-ecolog.ru	XML Схема, Часть 2 [18]

4.1.3 Пространство URI

Несмотря на то что URI, возвращенные Поставщиками, должны рассматриваться как непрозрачные для Потребителей. Потребители не должны знать конкретную структуру URI для Ресурсов, Потребители могут изменить URI, добавив некоторые четко определенные параметры запроса, которые поддерживаются Поставщиком, в соответствии с 4.1.6.

Демонстрационные URI, используемые в настоящем стандарте, не являются нормативными, и используемые шаблоны не должны интерпретироваться как руководство для реализаций. Например любой из следующих URI может использоваться Поставщиками для ссылки на конкретный Ресурс Machine:

gost.e-ecolog.ru

gost.e-ecolog.ru

gost.e-ecolog.ru

gost.e-ecolog.ru

4.1.4 Типы медиа

В настоящем стандарте представления Ресурсов и ответов закодированы в формате JSON в соответствии с [9], либо в XML. Если сериализация происходит в формате JSON, типом медиа для ресурсов CIMI должен быть «application/json». Если сериализация происходит в формате XML, типом медиа должен быть «application/xml».

В сериализации JSON представлений CIMI, отправленных Поставщиками, должен быть дополнительный атрибут корневого объекта, имеющий наименование «resourceURI» и содержащий уникальный URI, связанный с типом сериализуемого ресурса CIMI.

Данное требование применяется в том случае, даже если используется атрибут select для выделения подмножества запрашиваемого ресурса.

В сериализации XML представлений Набора, отправленного Поставщиками, должен присутствовать атрибут resourceURI, как показано на примере сериализации XML Наборов в 5.5.12.

Данный атрибут необязателен для его использования Потребителями. Если он присутствует, значение этого атрибута должно соответствовать атрибуту «typeURI» соответствующего Ресурса ResourceMetadata (см. 5.11), если ResourceMetadata поддерживается. Это значение также должно быть эквивалентно объемлющему элементу сериализации XML; другими словами, конкатенации пространства имен объемлющего элемента, знака «косая черта» («/») и его localName.

У любого ресурса CIMI, реализованного Поставщиком, должны быть представления в JSON и XML. Клиентская реализация может таким образом использовать либо JSON, либо XML в запросах с любой реализацией сервера и может запросить определенную сериализацию, используя процедуру согласования содержания сервером (используя заголовок запроса Accept).

4.1.5 Заголовки запроса

Для передачи метаданных сообщения в сообщениях запроса следует использовать общие заголовки, заголовки запроса и заголовки объекта в соответствии с [7]. Приложения, использующие сообщения, определенные в настоящем стандарте, должны использовать заголовки, соответствующие требованиям [7].

4.1.6 Параметры запроса

Поставщики могут принять решение включить параметры запроса как часть URI, возвращаемых Потребителям. Потребители должны включать данные параметры запроса при отправке сообщений в эти URI. Если Поставщики принимают решение определить собственные параметры запроса, то таким параметрам необходимо уделить отдельное внимание, чтобы избежать конфликтов с параметрами запроса, определенными CIMI.

Чтобы модифицировать поведение Поставщика при обработке сообщений запроса, Потребители могут расширить запрос URI в соответствии с указаниями, приведенными далее в настоящем пункте. В соответствии с 4.1.3 URI, возвращаемые Поставщиками, должны рассматриваться Потребителями как непрозрачные. Однако Потребитель несет ответственность за поддержку параметров запроса, определенных далее в настоящем пункте, и гарантию корректности при формировании запроса.

Параметры запроса, которые не поддерживаются или неизвестны, должны быть просто проигнорированы Поставщиками. Потребители могут исследовать возможности Ресурса CloudEntryPoint для определения того, предоставляется ли поддержка этих параметров запроса.

4.1.6.1 Фильтрация Наборов

Если Потребители получают представление Набора, они могут включать в себя параметр запроса $filter для сокращения числа возвращенных записей Набора на основании данных в рамках записей Набора. Поставщики должны интерпретировать и обрабатывать параметр запроса $filter в соответствии с описанием, приведенным далее. Параметр $filter должен иметь форму:

? $filter=expression,

где «expression» — математическое выражение, обозначающее, как следует фильтровать атрибуты верхнего уровня Ресурсов в Наборе. Выражение определено следующей грамматикой EBNF:

```
Filter ::= AndExpr ( 'or' Filter )* ;
AndExpr ::= Comp ( 'and' AndExpr )*
Comp ::= Attribute Op Value
| Value Op Attribute
| PropExpr
| '(' Filter ')'
Op ::= '<' | '<=' | '=' | '>=' | '>' | '!='
Attribute ::= ? наименование атрибута ресурса ?
Value ::= IntValue | DateValue | StringValue | BoolValue
IntValue ::= /[0-9]+/
DateValue ::= ? в соответствии с определением XML Схемы ?
StringValue ::= "..." | '...'
BoolValue ::= 'true' | 'false'
PropExpr ::= 'property[' StringValue ']' Op StringValue
```

PropExpr используется для нахождения Ресурсов, содержащих свойство с определенной комбинацией ключ/значение. Ключ — StringValue в квадратных скобках ([ ]), а значение — StringValue после

Op. Считается, что Ресурс удовлетворяет критериям поиска, если какое-либо из свойств в Ресурсах будет соответствовать указанному PropExpr.

Каждый из них должен быть соответствующим образом закодирован в URL с использованием процентов.

Выбор оператора (включая «and» и «or») ограничен на основании типа значения и атрибута. Допустимыми операторами являются:

«or», «and» : Булевское значение/атрибут;

'<', '<=', '=', '>=', «>', '!=' : Значение/атрибут целого типа или дата;

'=', '!=' : Строковое значение/атрибут.

Потребители могут включать несколько фильтров в один URI. Поставщик должен рассматривать набор фильтров как последовательность выражений, объединенных оператором «and», и запись из Набора должна быть включена в ответное сообщение, только если она удовлетворяет всем выражениям фильтра.

Примеры

В данных примерах используются следующие демонстрационные ссылки URI.

URI к MachineCollection Точки входа в облако следующее:

/Machines

URI экземпляра Machine:

/Machines/123

URI DiskCollection экземпляра Machine:

/Machines/123/disks

URI MachineVolumeCollection экземпляра Machine:

/Machines/123/Volumes

Чтобы отфильтровать MachineCollection так, чтобы возвращались только экземпляры Machine с атрибутом "name" равным "mine", используется следующий фильтр:

GET / Machines?$filter=name='mine'

Чтобы отфильтровать DiskCollection у экземпляра Machine так, чтобы были возвращены только экземпляры Disk с форматом "ntfs", используется следующий фильтр:

GET /Machines/123/disks?$filter=format='ntfs'

Если используется атрибут $filter, то атрибут "count" Набора должен содержать число Ресурсов, соответствующих выражению фильтра.

4.1.6.2 Подмножества Наборов

Получая представление Набора Потребители могут включать параметры запроса для определения подмножества объектов Набора, которые должны быть возвращены. Поставщики должны интерпретировать и обработать эти параметры запроса в соответствии с требованиями настоящего раздела. В предыдущем подпункте было указано, как выполнить фильтрацию данных Набора. Данный подпункт использует порядковое положение в рамках Набора для выделения подмножества.

В настоящем стандарте определены два параметра запроса, которые при их использовании должны указывать на первое и последнее порядковые положения объектов в пределах возвращенного Набора. Параметры запроса должны иметь форму:

?$first=*число*

?$last=*число,*

где "$first" указывает на порядковое положение (начиная с 1) первого объекта Набора для включения в ответ, а "$last" указывает на порядковое положение (начиная с 1) последнего объекта Набора для включения в ответ. Потребители не обязаны использовать их одновременно. Если $first будет определен, а $last не будет, то подразумеваемым значением $last должно быть числовое значение последнего объекта в Наборе. С другой стороны, если $last будет определен, а $first не будет, то подразумеваемым значением $first должна быть 1.

Если любая часть диапазона, выраженного с помощью $first и $last, выйдет за пределы границ Набора, то будут возвращены только те Ресурсы Набора (если они имеются), которые содержатся в пределах этого диапазона. Если какая-либо часть или весь выраженный диапазон будут за пределами Набора, сообщение об ошибке не должно генерироваться.

Примечание — Если значение $first более $last, диапазон должен представлять собой пустой массив, поэтому никакие Ресурсы не будут возвращаться.

Если $first или $last определены и выражение фильтра (в соответствии с 4.1.6.1) также определено, то первым должно быть обработано выражение фильтра, а затем следует применять порядковые ограничения $first и $last.

4.1.6.3 Выделение подмножества Ресурса

Запрашивая представление Ресурса Потребители могут включать параметр запроса $select, чтобы определить подмножество Ресурса, с которыми будут взаимодействовать. Поставщики должны интерпретировать и обработать этот параметр запроса в соответствии с требованиями настоящего раздела. Данное подмножество должно иметь семантическую эквивалентность ссылки на другой Ресурс, атрибутами которого является подмножество исходного Ресурса в соответствии с именами атрибутов, перечисленными в параметре запроса $select. Формат параметра запроса $select:

? $select=наименование Атрибута...

Значение параметра запроса $select должно выглядеть как перечень наименований атрибутов верхнего уровня Ресурса, разделенных запятой, возможно включая строку «operations» в этом случае, необходимо выбрать операции, доступные Потребителю для этого Ресурса. Любое наименование атрибута, ошибочно появляющееся в списке, который не является частью Ресурса, должно быть проигнорировано Поставщиком. Наименование атрибута «*» эквивалентно определению всех атрибутов Ресурса, включая его операции. Если наименование атрибута явно появляется в URI более одного раза, его второе (и последующие появления) должны быть проигнорированы.

В URI параметр запроса $select может появиться несколько раз. Это семантически эквивалентно появлению всех имен атрибутов как значения единичного параметра запроса $select. Например:

?$select=name&$select=state

эквивалентно:

? $select=name,state

В целях сериализации порядок имен атрибутов в параметре запроса $select не имеет значения. Атрибуты сериализуются в соответствии с правилами/порядком сериализации, указанным в определении Ресурса.

Примечание — Как указано в 4.1.4, если представление Ресурса отправляется Поставщиком, оно должно всегда включать в себя атрибут resourceURI, даже если это не определено в параметре запроса $select.

Например чтобы ограничить подмножество перечня атрибутов объектов Machine, с которыми Потребитель собирается взаимодействовать, только атрибутами «name» и «description», используется следующий параметр запроса:

?$select=name,description

Информация о влиянии этого параметра запроса при обновлении Ресурса приведена в 4.2.1.3.1.

Если для Ресурса Набора в URI используется $select, то подмножества должны применяться к атрибутам самого Набора, как для любого другого Ресурса. Например, следующее определение подмножества Ресурса Набора приведёт к передаче только числа элементов и операций, доступных для данного Набора:

?$select=count,operations

Однако в случае Ресурсов Набора, если некоторый атрибут, указанный в списке $select, не является атрибутом верхнего уровня Ресурса Набора, а вместо этого является атрибутом объектов, которые являются элементами Набора, то подмножество должно относиться к каждому элементу Набора относительно этого атрибута. Например при получении DiskCollection следующий параметр запроса ?$select=name,capacity возвращает набор экземпляров Disk, связанных с некоторым экземпляром Machine, но каждый объект Набора содержит только атрибуты name и capacity, а также атрибуты operation или id.

Опционально реализация также может поддерживать альтернативную нотацию наименования атрибута <collectionName>/<attributeName> для выделения подмножества элементов в наборе. Например следующее подмножество из элементов Набора DiskCollection эквивалентно тому, которое было выполнено в предыдущем примере; кроме того, включается перечисление операций самого ресурса Набора (а не его элементов):

?$select=disks/name,disks/capacity,operations

Если поддерживается эта нотация (см. возможность «QueryPathNotation» в 5.11.2), она позволяет разрешать неоднозначности подмножеств, если одно и то же наименование атрибута можно найти как в самом Наборе, так и для каждого элемента в Наборе (это всегда верно для id и operations).

4.1.6.4 Разворачивание ссылок

Запрашивая представление Ресурса Потребители могут включать параметр запроса $expand для определения, какие атрибуты верхнего уровня Ресурса, на которые указывают ссылки, должны быть добавлены в представление. Поставщики должны интерпретировать и обработать этот параметр запроса в соответствии с требованиями настоящего раздела. Разворачивание ссылки означает, что атрибуты Ресурса, на который указывает ссылка, должны быть включены в сериализацию этого атрибута. Это функция позволяет оптимизировать получение Ресурсов.

Сериализация должна быть выполнена следующим образом:

Сериализация JSON:

"name": {"href": string}

должна быть расширена, и стать:

"name": {

"href": string,

... атрибуты ресурса, на который указывает ссылка...

}

Сериализация XML:

<namehref = "xs:anyURI"/>

должна быть расширена, и представлять собой:

<name href = "xs:anyURI">

... атрибуты ресурса, на который указывает ссылка...

</name>

Примечание — В случае XML вложенные элементы не должны содержать элемент-обертку Ресурса, на который ссылаются (например, <Machine> в случае ссылки на Ресурс Machine).

Формат параметра запроса $expand должен быть следующим:

? $expand=наименование Атрибута...

Значение параметра запроса $expand —перечень наименований атрибутов, разделенных запятой. Любое наименование атрибута, ошибочно появляющееся в списке, которое не является частью Ресурса или ссылкой, должно быть проигнорировано Поставщиком. Наименование атрибута «*» или полное отсутствие перечня наименований атрибутов эквивалентно перечислению всех атрибутов. Если наименование атрибута явно появляется в URI более одного раза, его второе (и последующие появления) должны быть проигнорированы.

Параметр запроса $expand может появиться в URI несколько раз. Это семантически эквивалентно появлению всех наименований атрибутов как значения единичного параметра запроса $expand.

Если запрашиваемым Ресурсом является Набор, то наименования атрибутов, перечисленные в $expand, должны применяться к атрибутам ресурсов, содержащихся в Наборе. Например определение ? $expand=volumes при запросе к MachineCollection имеет тот же самый итоговый эффект, что и применение семантики «расширения» к указанному атрибуту (в данном примере «Volumes») каждого Ресурса Machine в пределах Набора. При этом $expand действует на атрибуты Ресурсов в Наборе, а не на атрибуты самого Ресурса Набора.

4.1.6.5 Определение формата Ресурса

Для определения стиля кодирования ответа при запросе представления Ресурса используется заголовок HTTP Accept. Несмотря на то, что Потребителям рекомендуется использовать заголовок Accept, могут возникнуть ситуации, когда Потребители будут не в состоянии управлять значениями, определенными в данном заголовке. В этих случаях Потребители могут использовать параметр запроса $format, чтобы переопределить значения заголовка Accept. Поставщики должны интерпретировать и обрабатывать параметр запроса $format, в соответствии с требованиями настоящего раздела.

Параметр $format должен иметь форму ? $format=encoding, где "encoding" — требуемое представление ответа. Настоящий стандарт определяет два возможных значения: «json» и «xml». Поставщик может поддерживать также другие значения. Значение параметра запроса $format не зависит от регистра.

Если в сообщении запроса будут присутствовать заголовок Accept и параметр запроса $format, то значение $format является приоритетным. Если параметр запроса $format появляется более одного раза, то второе и последующие его появления должны быть проигнорированы.

4.1.6.6 Сортировка Наборов

При запросе представления Набора Потребителям допускается включать параметр запроса $orderby для сортировки возвращаемых записей Набора на основании различных атрибутов и в различном порядке (убывания). Поставщики должны интерпретировать и обрабатывать параметр запроса $orderby в соответствии с требованиями настоящего раздела.

Параметр $orderby должен иметь форму

?$orderby=наименованиеАтрибута*[:asc|:desc], …*

Выражение $orderby может включать в себя несколько наименований атрибутов, разделенных запятой. Кроме того, каждое наименование атрибута может сразу сопровождаться двоеточием и ключевыми словами «asc» для обозначения порядка по возрастанию (значение по умолчанию) или «desc» для обозначения порядка по убыванию для данного атрибута. Если ни asc, ни desc не заданы, то порядок должен быть возрастающим.

Атрибуты, включенные в $orderby, должны иметь следующие типы в соответствии с 5.5: boolean, dateFormat, duration, integer или string.

Сортировка зависит от типа атрибута.

Следующие правила относятся к сортировке по возрастанию:

- boolean — значение «false» должно быть расположено перед значением "true";
- dateTime — более ранние значения дата/время должны быть расположены перед более поздними значениями дата/время;
- duration — более короткая продолжительность должна быть расположена перед более длительной продолжительностью;
- integer — меньшее целое число должно быть расположено перед большими целыми числами. Отрицательные целые числа должны быть расположены перед положительными целыми числами;
- string — порядок основан на порядке сортировки Unicode/UTF-8.

Порядок сортировки desc (по убыванию) должен быть противоположным указанному выше.

Примеры

Для сортировки набора результатов Ресурса MachinesCollection по атрибуту "created" в порядке убывания, используют следующее выражение:

GET / Machines?$orderby=created:desc

Для сортировки набора результатов Ресурса MachinesCollection по атрибуту "cpu" в порядке убывания, а затем по атрибуту "memory" в порядке возрастания, используют следующее выражение:

GET /Machines?$orderby=cpu:desc,memory:asc

4.1.6.7 Заголовки ответа

В соответствии с [7] для передачи метаданных сообщения в ответных сообщениях в настоящем стандарте использованы заголовки общего назначения, заголовки ответа и заголовки объекта (entity), чтобы предоставить метаданные о сообщении. В приложениях, в которых использованы сообщения, определенные в настоящем стандарте, следует использовать заголовки, совместимые с Реестром заголовков HTTP [5].

4.1.6.8 Заголовок Job

Если сервер поддерживает Ресурс Job, то ответные сообщения должны включать в себя заголовок, определенный в настоящем стандарте, чтобы указать на URI задания, созданного для обработки связанного с ним сообщения запроса:

CIMI-Job-URI = "CIMI-Job-URI" ":" string

В тех случаях, когда во время обработки запроса происходит ошибка, Поставщик должен включить представление Ресурса Job, описывающего состояние неудавшейся операции. Представление Job должно быть включено даже в тех случаях, если Поставщик обычно не поддерживает Ресурсы Job, чтобы гарантировать, что Потребителям предоставлена достаточная информация согласованным способом относительно причины неудачи, независимо от того, поддерживает ли Поставщик Ресурсы Job. Если Ресурсы Job в целом не поддерживаются, то любая из ссылок в представлении Job (например, «id» или «href» для nestedJobs) должна быть представлена пустыми путями (т. е.«»), и массив nestedJob должен быть развернут (см. 4.1.6.4), чтобы встроить представление псевдоподчиненных заданий Job.

4.1.6.9 Поддержка завершающего тега (ETag)

Заголовок ETag может быть предоставлен Поставщиком с каждым Ресурсом в соответствии с [7]. Если Поставщик действительно предоставляет заголовок ETag, то он должен также поддержать обработку заголовка If-Match от имени Потребителя.

4.2 Операции протокола

В настоящем подразделе определена совокупность общих операций HTTP, которые могут быть предоставлены Поставщиком. Его ядро составляют четыре основные операции CRUD — Create (Создать), Read (Читать), Update (Обновить) и Delete (Удалить). В рамках модели способ, которым они используются, является совместимым для всех Ресурсов, поэтому их использование определено один раз и должно унифицировано применяться. Некоторые Ресурсы поддерживают специализированные операции, которые не полностью подходят для стиля операций CRUD; такие операции тем не менее следуют общему высокоуровневому шаблону, но в каждой операции допускается вносить небольшие изменения, чтобы приспособить ее для определенных потребностей. Специфические особенности этих отдельных операций детально описаны в пункте, который определяет соответствующий Ресурс.

При необходимости некоторые представления Ресурса включают в себя атрибут «operations». Поставщики должны включать атрибут «operations», только в том случае, если указанные операции доступны для клиента этого конкретного Ресурса. Это означает, что при каждой сериализации Ресурса может возвращаться различная совокупность операций, что связано с многочисленными факторами (например, права авторизации клиентов, текущее состояние Ресурса и т.д.). Каждая операция должна включать в себя поля "rel" и "href". Поле "rel" должно однозначно определять наименование операции (например, "add", "edit"), в то время как поле "href" представляет собой URI, на который должно отправляться сообщение запрос операции.

Примечание — Поле URI "href" может отличаться от URI самого Ресурса.

Атрибут операций должен быть сериализован следующим образом:

Сериализация JSON:

```
{"operations": [
                {"rel": "string", "href": "string"}, +
      ]
}
```

Сериализация XML:

```
<Resource xmlns = "gost.e-ecolog.ru                    ">
   <operation rel="xs:anyURI" href ="xs:anyURI"/> *
</Resource>
```

Например операция "edit" выглядит следующим образом:

Сериализация JSON:

```
{"operations": [
                {"rel": "edit", "href": "<editURI>"}
      ]
}
```

Сериализация XML:

```
<Resource xmlns="gost.e-ecolog.ru                    ">
   <operation rel="edit" href = "<editURI>"/>
</Resource>
```

Поставщики могут определить дополнительные значения «rel», которые должны быть полностью квалифицированными URI, а не относительными URI.

4.2.1 Общие операции CRUD

Каждый из Ресурсов, поддерживаемых данным протоколом, должен придерживаться сценариев взаимодействия, определенных далее.

4.2.1.1 Создание нового Ресурса

Чтобы создать новый экземпляр типа Ресурса, на этот тип Ресурса отсылают запрос POSTHTTP к определенному "addURI". Во многих случаях ресурс Набора, который поддерживает или группирует все реализации этого типа Ресурса, включает операцию "add". Операция "add" ссылается на addURI, который должен использоваться.

Запрос POSTHTTP должен включать в себя:

- сериализацию CIMI запроса на создание нового Ресурса в теле HTTP;
- заголовок HTTPContent-Type;
- заголовок HTTPContent-Length.

Например запрос может быть следующим:

HTTP/1.1 POST<addURI>

Host: <hostname>

Accept: application / (json|xml)

Content-Type: application / (json|xml)

Content-Length: <*length*>

<*сериализация запроса создания нового ресурса*>

В данном примере есть заголовок Acceptc одним из типов медиа, поддерживаемых CIMI: application/json или application/xml. Если Поставщик принимает решение включить в ответ сериализацию, то данная сериализация должна иметь указанный тип медиа. Отсутствие заголовка Accept позволяет Поставщику включить в ответ сериализацию любого типа медиа. Если Ресурс будет содержат атрибут "state", то его значение должно быть "CREATING" в то время, когда Поставщик будет обрабатывать эту операцию.

Многие запросы create определены таким образом, чтобы передавать Шаблон нового Ресурса. Такие запросы create допускают, чтобы Шаблон передавался по ссылке или по значению. Например создание новой Machine может выглядеть следующим образом (в данном примере используется XML):

<MachineCreate xmlns = "gost.e-ecolog.ru "> <name>

xs:string </наименование>?

<description> xs:string </description>?

<property key= "xs:string"> xs:string </property> *

<MachineTemplate href = "xs:anyURI"?>

... атрибутышаблона...?

</MachineTemplate>

</MachineCreate>

Примечание — В случае XML создание новой Machine требует наличия элемента обертки под названием MachineCreate в соответствии с правилами, определенными в 5.5.12.1.

Создание нового Ресурса осуществляется в соответствии с одним из двух сценариев сериализации (данный пример приведен в JSON):

(1) Создание ресурса путем передачи Шаблона по значению:

{"resourceURI": "gost.e-ecolog.ru ",

"name": "myResourceName"?

"description": "Мое описание ресурса"?

"properties": {*"prop1name":"prop1value"*, +}?

"resourceTemplate": {

<в этом случае шаблон передан значением>

}

}

(2) Создание ресурса с передачей шаблона по ссылке:

{"resourceURI": "gost.e-ecolog.ru ",

"name": "myResourceName"?

"description": "Мое описание ресурса"?

"properties": {*"prop1name":"prop1value"*, +}?

"resourceTemplate": {"href": *строка*,

<в этом случае могут быть добавлены некоторые пары атрибут/значение шаблона для переопределения значения в шаблоне, на который указывает ссылка>

}

}

В случае, если созданный Ресурс сам является Шаблоном, то применяется только первый сценарий создания — по значению. В сценариях (1) и (2) атрибут resourceURI определяет операцию, которую в общем случае можно идентифицировать как "ResourceCreate", например MachineCreate.

В сценариях (1) и (2) элемент, соответствующий Шаблону Ресурса (идентифицированному как «resourceTemplate», например MachineTemplate), определяет Шаблон, который будет использоваться либо по значению (1), либо по ссылке (2).

Прямая установка атрибутов в новом Ресурсе:

В запросе создания допускается установить значения некоторых атрибутов созданного Ресурса независимо от того, какие значения были заданы в экземпляре Шаблона, если использовался только этот экземпляр. В созданном Ресурсе могут быть установлены три общих атрибута: name (наименование), description (описание) и properties (свойства).

Семантика должна быть такой же, как и при частичном обновлении Ресурса для этих атрибутов (описанном в следующем подразделе), последовавшем непосредственно за созданием Ресурса только из одного Шаблона.

Определение или ссылка на Шаблон Ресурса:

В сценарии (1) Поставщик может принять решение о создании Ресурса Шаблона из переданного значения, но такой экземпляр будет временным. Поставщик не должен представлять такой промежуточный Ресурс Потребителю, и ни один такой промежуточный Ресурс не должен быть включен в какие-либо результаты запроса в ответе Потребителю.

В сценарии (2) дополнительные пары наименование атрибута/значение атрибута могут передаваться внутри элемента ResourceTemplate, который также содержит ссылку на внешний (существующий ранее) Шаблон, чтобы переопределить подобные атрибуты, определенные в Шаблоне:

- любой атрибут верхнего уровня составного или простого типа в Шаблоне, на который ссылается операция, должен быть переопределен путем присвоения ему пары наименование/значение в запросе create в элементе resourceTemplate и сразу под ним. Для атрибута верхнего уровня составного типа (например, массивы, Наборы, структуры), предоставленное составное значение должно также устанавливать все вложенные атрибуты, например элементы массива;

- семантика должна быть такой же, как при изменении (переопределении) частей Шаблона, на который ссылается операция, непосредственно перед его использованием для создания экземпляра Ресурса, но эти переопределения не должны сохраняться в Шаблоне, на который указывает ссылка и должны иметь отношение только к этому экземпляру.

В сценарии (2) Потребители могут стереть любые атрибуты Шаблона, определяя либо

«attribute»: null

для атрибута в сериализации JSON, либо

<attribute/>

в сериализации XML для этого атрибута.

Примеры

Пример сценария создания (1) с использованием MachineTemplate по значению (в JSON):

{ "resourceURI": "gost.e-ecolog.ru ***",***

"name": "myMachine123",

"description": "Машина, которая будет подключена к существующей сети",

"machineTemplate": {

<в этом случае шаблон передан по значению, т. е. парами атрибут/значение для Шаблона MachineTemplate. Нижеприведенпримердля networkInterfaces>

"networkInterfaces": [

{ "addresses": [{"address": {"href": "gost.e-ecolog.ru ***,***

{"address": {"href": "gost.e-ecolog.ru ***],***

"network": {"href": "gost.e-ecolog.ru ***,***

"state": "ACTIVE"}

]

}

}

В данном примере:

Атрибуты name и description являются параметрами уровня экземпляра Ресурса, потому что они находятся вне элемента MachineTemplate (т. е. они устанавливают значения атрибута в новом созданном Ресурсе, а не в Шаблоне, используемом для создания такого Ресурса). Наименование нового экземпляра Machine — "myMachine123".

Этот экземпляр Machine подключен к существующей сети (экземпляр Network), обозначенному ссылкой (gost.e-ecolog.ru), указанной в составном атрибуте Шаблона.

Пример

Пример сценария создания (2) с использованием MachineTemplate по ссылке:

```
{ "resourceURI": "gost.e-ecolog.ru                                                    ",
"name": "myMachine456",
 "description": "Машина подключена к существующему тому",
"machineTemplate": {
"href": "gost.e-ecolog.ru                                                    ",
Credential": {"href": "gost.e-ecolog.ru
"networkInterfaces": [
{ "addresses": [{"address": {"href": "gost.e-ecolog.ru                                                    ,
{"address": {"href": "gost.e-ecolog.ru                                                    ],
"network": {"href": "gost.e-ecolog.ru                                                    ,
"state": "ACTIVE"}
]
}
}
```

В данном примере создана новая машина, названная "myMachine456", которая также связана с существующей сетью Network, как и в примере (1), но с иной совокупностью адресов Addresses. В настоящем примере во время создания присваиваются значения двум видам атрибутов:

- установка атрибута уровня экземпляра Ресурса: эти атрибуты должны быть незамедлительно обновлены в созданном Ресурсе, в данном случае name и description;

- переопределение атрибутов Шаблона: MachineTemplate, на который указывает ссылка, используется для создания Machine, но атрибут Credential в этом Шаблоне переопределен учетными данными, предоставленными в запросе создания, также как и массив networkInterfaces. В случае, если такие атрибуты не присутствовали в Шаблоне по ссылке, их добавляют (временно) только для создания этого экземпляра Machine.

Некоторые запросы на создание допускают передавать конфигурационные параметры Ресурсов по ссылке или по значению, например Credential в операции создания Machine. Правила обработки, определенные выше, применяют также и в этих случаях.

Если ответ имеет код статуса 201, то ответ должен включать в себя:

- заголовок HTTP Location со ссылкой на новый Ресурс.

Если ответ на запрос включает в себя сериализацию нового Ресурса, то ответ должен дополнительно содержать:

- заголовок HTTP Content-Type;
- заголовок HTTP Content-Length.

Например ответ может быть следующим:

HTTP/1.1 201 CreatedLocation: <местоположение>
Content-Type: application/(json|xml)
Content-Length: <длина>
<сериализация нового ресурса>

4.2.1.3 Обновление ресурса

Для обновления состояния Ресурса по адресу editURI, заданного для этого типа Ресурса, направляют запрос PUTHTTP, содержащий полное обновленное представление. Потребители должны включать в запрос PUT все непустые атрибуты Ресурса, включая те, которые они, возможно, не поддерживают или не понимают, которые возвращались в ответе GET. Это нужно для того, чтобы гарантировать, что клиент не изменяет (стирает) непреднамеренно данные в Ресурсе, исключив их из полного представления Ресурса.

Во многих случаях данный editURI совпадает с URI самого Ресурса. При получении представление Ресурса должно включать в себя операцию «edit», содержащую editURI, который должен использоваться, если запрашивающей стороне разрешено изменять Ресурс.

Если при обработке запроса PUT сервер обнаруживает, что предпринимается попытка обновить атрибут, предназначенный только для чтения или неизменяемый атрибут, он должен проигнорировать этот запрос обновления атрибута, не генерируя сообщения об ошибке. Это правило также относится к частичному обновлению Ресурса.

Из-за потенциальных конфликтов, которые могут произойти вследствие нескольких параллельных обновлений, Потребители должны использовать механизм частичного обновления, определенный в 4.2.1.3.1, чтобы уменьшить возможности ошибочного обновления атрибутов устаревшими данными.

Запрос PUT HTTP должен включать в себя:
- сериализацию CIMI обновленного Ресурса в теле сообщения HTTP;
- заголовок HTTP Content-Type;
- заголовок HTTP Content-Length.

Например запрос может быть следующим:
PUT<editURI>HTTP/1.1
Host: <наименование узла>
Accept: application / (json|xml)
Content-Type: application / (json|xml)
Content-Length: <длина>
<сериализация запроса с целью обновить ресурс>

Если ответ будет включать в себя сериализацию обновленного Ресурса и иметь код статуса 200, то данный ответ должен включать в себя:
- заголовок HTTP Content-Type;
- заголовок HTTP Content-Length.

Например ответ может быть следующим:
HTTP/1.1 200 OK
Content-Type: application / (json|xml)
Content-Length: <длина>
<сериализация обновленного ресурса>

4.2.1.3.1 Частичное обновление Ресурса

В данном подпункте определено, как следует использовать параметр запроса $select (см. 4.1.6.3) для определения подмножества Ресурса с целью работы только на выбранной совокупности атрибутов верхнего уровня.

Для обновления конкретных атрибутов Ресурса только верхнего уровня Потребитель может включать в представление Ресурса в теле запроса HTTP только измененные атрибуты. Для выполнения такого запроса URI для Ресурса должно включать в себя атрибуты, которые будут изменены как перечень значений параметров запроса, разделенных запятой, URI должен иметь следующий вид:

gost.e-ecolog.ru ...

Изменению подлежат только те атрибуты, которые перечислены в параметрах запроса URI, атрибуты, не перечисленные в URI, не должны напрямую изменяться в результате запроса.

Примечание — Это обстоятельство не устраняет возможность возникновения побочных эффектов в результате обновления атрибута, которые приведут к изменению атрибута, отсутствующего в параметрах запроса.

Любой атрибут, перечисленный в URI, но не включенный в запросHTTP, должен быть переустановлен на значение, характерное для Ресурса (например, удален).

С точки зрения HTTP частично обновленный Ресурс является индивидуальным. Семантика обычного HTTPPUT сохраняется, что является полной заменой указанного Ресурса. С точки зрения Потребителя частичное обновление интерпретируется и выполняется Поставщиком Службы Облачных Вычислений, при этом изменяется некоторая часть Ресурса.

В соответствии с общей семантикойPUT, определенной ранее, любой атрибут исходного (полного) Ресурса, включенный в запрос HTTP, должен привести к ошибке, если этот атрибут не перечислен в параметре запроса $select (см. 5.4).

Примечание — Это происходит вследствие того, что такие атрибуты неизвестны частичному Ресурсу.

В следующем примере показано, как запрос обновляет только атрибуты name и description Machine:
PUT /Machines/myMachine?$select=name,description HTTP/1.1
Host: <наименование узла>
Accept: application/xml
Content-Type: application/xml
Content-Length: <длина>

<Machine>
<name> My New Machine </name>
</Machine>
Атрибут name установлен в "My New Machine", а атрибут description удален.

4.2.1.4 Удаление ресурса

Для удаления Ресурса передают запрос HTTPDELETE по адресу deleteURI , заданный для этого типа Ресурса. Во многих случаях deleteURI совпадает с URI самого Ресурса. При получении представление Ресурса должно включать в себя операцию delete, содержащую deleteURI, который должен использоваться, если запрашивающей стороне разрешено удалять Ресурс.

Например запрос может быть следующим:

DELETE<*deleteURI*>HTTP/1.1

Host:<наименование узла>

Если Ресурс содержит атрибут state, то его значением должно быть "DELETING" в то время, когда Поставщик обрабатывает данную операцию.

Например ответ может быть следующим:

HTTP/1.1 200 OK

4.2.1.5 Другие операции

Несмотря на то, что как правило модификация Ресурса в модели может осуществляться путем простой операции обновления (PUT) Ресурса по адресу editURI, иногда требуется более сложная совокупность действий. В этих случаях операции должны моделироваться как запросы POSTHTTP, направленные по адресам URI, определенным для Ресурса.

Для каждого из Ресурсов, которые определяют дополнительные операции, предоставляется описание запроса HTTP и содержание ответа. Однако общее взаимодействие HTTP происходит в соответствии с приведенным далее.

Запрос должен иметь следующий вид:

HTTP/1.1 POST<URI операции>

Host: <наименование узла>

Accept: application/(json|xml)

Content-Type: application/(json|xml)

Content-Length: <длина>

<*сериализация запроса операции*>

Форма ответа варьируется в зависимости от операции и определена самой операцией.

Примечание — Определение операции CREATE (см. 4.2.1.1) должно соответствовать этому шаблону.

4.2.1.6 Синхронные операции

Если Поставщик поддерживает Ресурс Job, то каждый входящий запрос PUT, DELETE, POST должен приводить к созданию Ресурса Job и абсолютная ссылка URI на этот Ресурс Job должна возвращаться обратно клиенту в заголовке CIMI-Job-URI в ответном сообщении HTTP:

CIMI-Job-URI: <*uri-to-Job*>

В этом случае требуемая операция должна быть завершена и JobURI должен указать на выполненное задание. Если задание не будет завершено, то сервер должен вернуть код ответа 202 и следовать инструкциям для выполнения асинхронных операций.

4.2.1.7 Асинхронные операции

В некоторых случаях выполнение операции, затребованной клиентом, может занять неопределенный промежуток времени для своего завершения. Например создание нового экземпляра Machine или запуск существующего экземпляра Machine могут занять относительно много времени до своего завершения. В таких случаях непрактично завершать эти операции в пределах таймаута запроса HTTP, поэтому Поставщик должен вернуть код ответа HTTP «202 Accepted».

Как и в случае с синхронными операциями, если Поставщик поддерживает Ресурс Job, он должен создать Ресурс Job для входящего запроса и вернуть ссылку на этот Ресурс Job обратно клиенту в заголовке CIMI-Job-URI в ответном сообщении HTTP. Кроме того, в случае, если кодом ответа является «202 Accepted», Поставщик может также вернуть в теле ответа HTTP любое из следующего:

- представление Ресурса Job, если такой был создан;

- частичное представление ответного сообщения, как если бы операция была синхронной. Например при создании нового экземпляра Machine ответное сообщение может включать в себя частичное представление нового экземпляра Machine в ответном сообщении. Возвращаемый перечень атрибутов Ресурса определяется реализацией и основывается на том, сколько информации доступно в то время, пока генерируется ответное сообщение, но он должен быть совместим с представлением полного

Ресурса. При выполнении операции create Поставщик может также включать заголовок HTTP Location, содержащий ссылку на создаваемый Ресурс, если он известен;

- пустое тело ответа.

Примечание — Решение по поводу того, является ли какая-то конкретная операция синхронной или асинхронной, принимает сервер.

4.3 Поддержка OVF

В спецификации открытого формата виртуализации (OVF) по [2] описан открытый, безопасный, переносимый, эффективный и расширяемый формат для упаковки и распространения программного обеспечения, которое будет выполняться в виртуальных машинах. Поддержка OVF в CIMI позволяет использовать пакеты OVF для создания ресурсов менеджмента CIMI путем импорта пакета. Кроме того, ресурсы менеджмента CIMI могут быть экспортированы в пакет OVF. Фактическая поддержка пакета OVF обычно реализуется гипервизором, которым управляет поставщик CIMI. Импорт пакета OVF предоставляет доступ к набору конструкций и параметров, определяемых CIMI, без изменений исходного пакета OVF. Таким образом, ресурсы CIMI, созданные в результате импорта, формируют «представление» того, что сделано гипервизором. Однако другая (не имеющая отображение в CIMI) информация от пакета OVF может быть использована гипервизором при его импорте. Такая информация определяется реализацией и далее не рассматривается в настоящем стандарте.

Пакет OVF может поддерживать единичные виртуальные машины (далее — ВМ), соответствующие единичному экземпляру CIMI Machine или MachineTemplate(см. 5.14.1), либо может поддерживать сложную иерархию ВМ и связанных с ними ресурсов, соответствующих CIMISystem или SystemTemplate (см. 5.13.1), и связанных ресурсов менеджмента CIMI.

Описание поддержки OVF более подробно приведено в приложении А.

5 Модель

Данная модель предполагает, что между Потребителем и Поставщиком уже установлены деловые отношения. Эти отношения могут включать в себя финансовые условия, приводящие к созданию отдельно администрируемых облаков, за которые платит организация-потребитель, и установлению учетных данных аутентификации для получения доступа к административной точке входа для каждого облака. Областью применения данной модели является отдельно администрируемое облако.

Описание модели CIMI приводится в виде таблицы. Прототипом послужило моделирование сущность — связь (Entity-Relationship), где каждый объект моделирует значительный облачный ресурс, для которого ожидаются независимый доступ и манипуляция. Отношения между ресурсами осуществляется с помощью механизма ссылок, основанного на уникальных идентификаторах, который, как ожидается, уже поддерживается окружающей средой реализации и протоколом (например, URI для HTTP).

Модель описывает сама себя и допускает запросы своих собственных метаданных, например для обнаружения поддерживаемых расширений. Также модель может быть расширена различными способами (см. 5.1).

Наряду с этой моделью определена сериализация ее объектов (как в XML, так и в JSON).

Альтернативное представление каждой главной группы ресурсов представлено в виде диаграммы UML.

5.1 Обертки Ресурсов

В данной модели сериализация экземпляров Ресурсов должна соответствовать ряду правил. Например сериализация Ресурса с именем MyResource должна соответствовать следующим правилам:

Сериализация JSON:

Ресурс сериализован как объект, обертывающий все его атрибуты, но без наименования обертки. Ресурс включает в себя resourceURI с URI типа сериализуемого Ресурса, например:

```
{"resourceURI": "gost.e-ecolog.ru                    ",
"attribute": "value"
}
```

Сериализация XML:

Ресурс сериализован как элемент с наименованием, соответствующим наименованию Ресурса; например:

<MyResource xmlns = "gost.e-ecolog.ru
<attribute>value</attribute>
</MyResource>

5.2 Расширяемость

Модель CIMI определяет два типа механизмов расширяемости: один предназначен для использования Потребителями, а второй должен использоваться Поставщиками.

Первый тип механизма расширяемости позволяет Потребителю CIMI добавлять дополнительные данные к Ресурсу. У каждого Ресурса в модели CIMI есть атрибут с наименованием properties. При создании или обновлении Ресурса Потребители могут сохранять любую пару наименование/значение в атрибуте properties. Поставщики CIMI должны сохранять и возвращать эти значения Потребителю. От Поставщика не требуется понимать эти значения или предпринимать какие-либо действия на основании этих значений, так как они существуют только для удобства Потребителя. Поставщики не должны добавлять элементы к атрибуту properties.

Второй тип механизма расширяемости позволяет Поставщику определять свои расширения; в настоящем стандарте для этих целей включен Ресурс ResourceMetadata, который используют, чтобы:

- ввести ограничения на существующие атрибуты ресурса, определенные CIMI, (например, установить максимальное значение для атрибута «cpu» Ресурса MachineConfiguration);
- ввести новые атрибуты для определенных Ресурсов CIMI вместе с любыми ограничениями, связанными с ними (например, новый атрибут «location» для Ресурса Volume, который принимает значение, состоящее из определенной совокупности строк);
- ввести новые операции для любого из Ресурсов, определенных CIMI (например, определить новую операцию «compress» (сжать) для Ресурса Volume);
- указать определенные возможности или особенности любого Поставщика (например, отрезок времени, в течение которого Ресурс Job сохраняется после завершения задания Job).

Поставщикам рекомендуется использовать Ресурс ResourceMetadata для распространения этих атрибутов, операций и возможностей наряду с любыми ограничениями, которые должны быть понятными для Потребителей. Ресурс ResourceMetadata определен в 5.11.

Если Поставщик получает сообщение, содержащее атрибут, который неизвестен или не поддерживается, он должен отклонить запрос. Если Потребитель получает сообщение, содержащее атрибут, который неизвестен или не поддерживается, он должен проигнорировать данный атрибут. Однако Потребители обязаны включать эти атрибуты в сообщения, которые передаются обратно Поставщику.

Примечание — В этих случаях Потребитель не должен понимать или обрабатывать неподдерживаемый атрибут, а должен просто вернуть его обратно Поставщику.

5.3 Идентификаторы

Все идентификаторы (например, наименования Ресурсов, атрибуты, операции, названия параметров), приведенные в настоящем стандарте или определенные расширением, должны соответствовать следующим правилам:

- наименования идентификатора считают чувствительными к регистру;
- наименования идентификатора должны использовать только следующую совокупность символов:
- Прописной ASCII (от U+0041 до U+005A);
- Строчный ASCII (от U+061 до U+007A);
- Цифры (от U+0030 до U+0039);
- Нижнее подчеркивание (U+005F);
- число идентификатора не должны начинаться с Цифры (от U+0030 до U+0039).

Примечание — Эти правила не относятся к общему атрибуту "name", определенному в 5.10.1.

5.4 Ограничения атрибута

Каждый атрибут Ресурсов в модели CIMI дополнен рядом ограничений, которые дополнительно квалифицируют определяемый атрибут. Для каждого атрибута существует совокупность ограничений Поставщика и Потребителя, так как они могут различаться. Возможны следующие ограничения:

необязательная поддержка:

Это ограничение означает, что поддержка для данного атрибута будет опциональной. Если атрибут поддерживается, Поставщики должны распространить информацию о его поддержке через ResourceMetadata. Данные относительно обработки атрибутов, которые неизвестны или не поддерживаются, представлены в 5.2, относительно пустых значений атрибута — в 5.5.14.

Непустые атрибуты, поддерживаемые Потребителем, поддерживающие запись («чтение-запись» и «только для записи»), должны быть частью представления Ресурса, отправленного Потребителем Поставщику, включая запросы на создание.

Непустые атрибуты, поддерживаемые Поставщиками, должны быть частью представления Ресурса, отправленного Поставщиком Потребителю;

обязательная поддержка:

Это ограничение означает, что реализации, совместимые с настоящим стандартом, должны поддерживать атрибут. Если он задан для вложенного атрибута, то поддержка требуется только в том случае, если поддерживается родительский атрибут. Данные о пустых значениях атрибута приведены в 5.5.14.

Непустые обязательные атрибуты, поддерживающие запись («чтение-запись» и «только для записи»), должны быть частью представления Ресурса, отправленного Потребителями Поставщикам, включая запросы «создать».

Непустые обязательные атрибуты Поставщика должны быть частью представления Ресурса, отправленного Поставщиками Потребителю;

неизменяемый:

Это ограничение Поставщика означает, что атрибут, установленный однажды, не должен быть изменен в течение всего срока жизни Ресурса;

изменяемый:

Это ограничение Поставщика означает, что атрибут может быть изменен. Поставщики должны иметь возможность изменять эти атрибуты. Возможность изменения этих атрибутов Потребителями обозначается ограничениями: «только для чтения», «чтение-запись» и «только для записи»;

только для чтения:

Это ограничение Потребителя означает, что атрибут может быть получен, но не может быть обновлен Потребителями. Атрибуты «только для чтения» не требуется включать в сериализации Ресурсов в запросах создания или обновления Ресурсов. Если данные атрибуты указаны в сообщении, они должны быть проигнорированы Поставщиком. Атрибуты вида «только для чтения» должны быть включены в сериализации Ресурсов, отправленных Поставщиками;

чтение-запись:

Это ограничение Потребителя означает, что атрибут может быть получен и/или обновлен Потребителями. Атрибуты «чтение-запись» должны быть включены в сериализации Ресурсов, отправленных Поставщикам и от них. Поставщики могут ограничить возможность обновлять эти атрибуты Потребителями и должны обозначить ограничение в ResourceMetadata;

только для записи:

Это ограничение Потребителя означает, что атрибут может быть обновлен Потребителями, но не передается Потребителям, обычно из соображений безопасности. Атрибуты «только для записи» должны быть включены в сериализации Ресурсов, отправленных Поставщикам, но не должны быть включены в сериализацию Ресурсов, отправленных Поставщиками.

5.5 Типы данных и их сериализация

В случае, если явным образом не указано требование исключить конкретные атрибуты из представления Ресурса, отсутствие необязательного атрибута в представлении означает, что атрибут не имеет значения (т. е. не определен). Настоящий стандарт не распространяется на концепцию необязательного атрибута, имеющего значение по умолчанию.

Примечание — Клиент не может определить (при рассмотрении возвращенного представления), поддерживается ли конкретный отсутствующий атрибут, или он не существует.

Аналогичным образом исключение атрибута из представления Ресурса во входных данных операции обновления означает, что Потребитель запрашивает у Поставщика удаления этого атрибута.

Типы данных и значения, используемые в таблицах, предназначенных для определения CIMI, приведены далее.

5.5.1 boolean (логический тип)

Значение соответствует xs:boolean«XML-схема часть 2» [18], за исключением того, что допустимыми являются только два значения: либо "true", либо "false". Данное значение является чувствительным к регистру.

При сериализации в JSON значения должны иметь тип JSON: *boolean.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:boolean.*

5.5.2 dateTime (дата и время)

Значение соответствует xs:dateTime «XML-схема часть 2» [18], является совместимым с DMTFDSP4004 [4] и ISO 8601 [11]. Метка времени должна содержать информацию о часовом поясе, т. е. содержать компоненты местного времени и смещения от UTC.

Любые ограничения определенных диапазонов, допускаемые для любого конкретного атрибута, установлены в определении данного атрибута либо во время выполнения Поставщиком с помощью механизмов обнаружения метаданных, определенных данной спецификацией.

Например момент времени «понедельник, 25 мая 2012, в 1:30:15 PMEST» (Североамериканский восточный часовой пояс) представлен как:

2012-05-25T13:30:15-05:00

При сериализации в JSON значения должны иметь тип JSON: *string.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:dateTime.*

5.5.3 duration (продолжительность)

Значение соответствуетxs:duration «XML-схема часть 2» [18]. Любые ограничения определенных диапазонов, допускаемые для любого конкретного атрибута, установлены в определении данного атрибута либо во время выполнения Поставщиком с помощью механизмов обнаружения метаданных, определенных данной спецификацией.

При сериализации в JSON значения должны иметь тип JSON: *string.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:duration.*

5.5.4 integer (целое число)

Значение соответствуетxs:integer «XML-схема часть 2» [18]. Любые ограничения определенных диапазонов, допускаемые для любого конкретного атрибута, установлены в определении данного атрибута либо во время выполнения Поставщиком с помощью механизмов обнаружения метаданных, определенных данной спецификацией.

При сериализации в JSON значения должны иметь тип JSON: *number*.

При сериализации в XML значения должны иметь тип схемы XML: *xs:integer.*

5.5.5 string (строка)

Значение соответствуетxs:string «XML-схема часть 2» [18]. Любые ограничения к данному типу для любого конкретного атрибута установлены в определении этого атрибута либо Поставщиком во время выполнения с помощью механизмов обнаружения метаданных, определенных данной спецификацией.

При сериализации в JSON значения должны иметь тип JSON: *string.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:string.*

При сериализации атрибута «строка» необходимо пропустить атрибут, значение которого равно пустой строке.

5.5.6 ref (ссылка)

Ссылка на другой Ресурс.

Ссылки означают для Потребителей возможность перехода к Ресурсам, начиная с Точки входа в облако и следуя по ссылкам, которые появляются в полученных Ресурсах, Потребители могут рекурсивно обнаружить ссылку и перейти ко всем другим Ресурсам.

Как правило, значение ссылки для атрибута хранится во вложенном атрибуте, имеющем наименование "href" (в JSON и в XML).

Сериализация JSON:

В сериализации JSON "href" появляется как тип «строка» — string. Если атрибут будет иметь тип «ссылка», то наименование этого атрибута должно быть ключом, имеющим свойство "href" как вложенное значение. Например атрибут Ресурса "myVolume" типа «ссылка» сериализован как:

"myVolume": {"href": *строка*}

Сериализация XML:

В сериализации XML атрибут href появляется как тип "xs:anyURI". Если атрибут будет иметь тип «ссылка», то наименование этого атрибута должно появиться как наименование элемента XML с XML атрибутом href. Например атрибут Ресурса "myVolume" типа «ссылка» сериализован как:

<myVolumehref = "xs:anyURI"/>

Ссылки в JSON и в XML имеют точку расширения, которая допускает наличие дополнительной информации (например, целевой Ресурс, который будет включен по значению), если данная информация поддерживается. Для удобства представления JSON и XML в соответствии с вышеуказанным, не включают в себя неявные возможные расширения, которые допускают атрибуты целевого Ресурса. Поэтому более точно данные представления могут быть записаны следующим образом:

Для JSON:

"myVolume": {"href": *string,...*}

для XML:

<myVolume href = "xs:anyURI">*xs:any** </myVolume>

Однако для сокращения точки расширения исключены из сериализации Ресурсов.

5.5.7 map (отображение)

Атрибутmap представляет собой «ключ/значение». В рамках атрибута каждый ключ должен использоваться только один раз. Ключи являются чувствительными к регистру.

Если значение атрибута типа «отображение» равно пустому отображению, то этот атрибут не должен быть представлен в сериализации.

5.5.8 structure (структура)

Атрибуты данного типа являются составными атрибутами, состоящими из ряда вложенных атрибутов. Для каждого атрибута данного типа существует дополнительная таблица, определяющая вложенные атрибуты.

Вложенную структуру можно считать определением составного типа. Структуры могут иметь или не иметь наименований. Примеры структуры, имеющей наименование, приведены в таблице 2.

Таблица 2 — Структура, имеющая наименование

Наименование	*summary*	
Атрибут	Тип	Описание
low	*integer*	Число произошедших событий с незначительной степенью важности
medium	*integer*	Число произошедших событий со средней степенью важности
high	*integer*	Число произошедших событий с высокой степенью важности
critical	*integer*	Число произошедших событий с критической степенью важности

Сериализация JSON:

В JSON наименование структуры (т. е. типа, который она представляет) не появляется никогда. Другими словами, имеет ли структура наименование или нет, не имеет значения. Атрибут «SystemIncidents» типа summary (см. выше), сериализован следующим образом:

"systemIncidents": {
"low": *number*,
"medium": *number,*
"high": number,
"critical": number
}

Сериализация XML:

В XML наименование структуры (т.е. Наименование типа, который она представляет) не появляется никогда. Другими словами, имеет ли структура наименование или нет, не имеет значения. Предыдущий пример «SystemIncidents» сериализован так, чтобы атрибуты структуры стали атрибутами элемента-обертки XML<SystemIncidents>:

<systemIncidents low = "xs:integer" medium = "xs:integer" high = "xs:integer" critical = "xs:integer"/>

Примечание — В случае большого количества атрибутов атомарного типа в структуре они могут быть также представлены как дочерние элементы XML для лучшей читаемости. Допускаются оба варианта, однако одна структура должна быть сериализована одним и тем же образом во всех Ресурсах.

5.5.9 byte[] (массив байтов)

Произвольная совокупность байтов для представления блоков двоичных данных. Любые ограничения к данному типу для любого конкретного атрибута установлены в определении этого атрибута

либо Поставщиком во время выполнения с помощью механизмов обнаружения метаданных, определенных данной спецификацией.

При сериализации в JSON значения должны иметь тип JSON: *строка –string.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:hexBinary.*

5.5.10 URI

Формат и синтаксис атрибутов типа «URI» определены в RFC3986 [8].

Настоящий стандарт не устанавливает требований к использованию Поставщиками относительного или абсолютного URI в теле ответа HTTP.

Если URI определены как относительные URI, то они должны быть заданы относительно baseURI.

Алгоритм, используемый для преобразования относительного URI в абсолютный URI, должен соответствовать описанию, приведенному в 5.2 RFC3986 [8]. Преобразование относительного URIв абсолютныйURI приведено в таблице 3.

Таблица 3 — Преобразование относительного URI в абсолютный URI

Базовый URI	Относительный URI	Абсолютный URI
gost.e-ecolog.ru	p1/file	gost.e-ecolog.ru
gost.e-ecolog.ru	p1/file	gost.e-ecolog.ru
gost.e-ecolog.ru	p1/file	gost.e-ecolog.ru

При использовании относительных URIbaseURI должен заканчиваться наклонной чертой, и относительные URI не должны начинаться с наклонной черты. Этот формат совместим с большинством утилит разрешения URI и приводит к тем же результатам, что и простой алгоритм конкатенации строк.

При сериализации в JSON значения должны иметь тип JSON: *строка — string.*

При сериализации в XML значения должны иметь тип схемы XML: *xs:anyURI*.

5.5.11 Массивы

Массив представляет собой упорядоченный перечень элементов одного и того же типа. Массив должен быть атрибутом Ресурса и быть доступен только как таковой (он не является отдельно адресуемым Ресурсом). Если Ресурс будет удален, то элементы его массивов также должны быть удалены. Однаков случае, если эти элементы являются ссылками на другие Ресурсы, то Ресурсы, на которые имеются ссылки, не будут затронуты (см. 5.7).

Атрибуты, являющиеся массивами, определены с помощью нотации itemType [], где itemType — наименование типа для каждого элемента массива. Если тип представляет собой структуру, а не просто тип данных, то рекомендуется в качестве соглашения в модели использовать множественное число в наименовании массива, которое характеризует каждый элемент. Например массив элементов имеет ли структура наименование "Volume" или ссылок на них может иметь наименование "Volumes".

Если атрибут будет иметь тип «массив ссылок» (ref[]) и, в более общем случае, массив атомарного типа, то определение в модели должно включать в себя поле «наименование элемента массива», которое может использоваться в его сериализации.

Сериализация JSON:

В рамках данной спецификации массивы в JSON сериализуются как значение некоторого свойства. Наименование свойства должно быть тем же самым, что и наименование атрибута для массива. Например атрибут "things" типа "thing[]" сериализуется следующим образом:

"things": [

{... }, +] ?

Если элементы в массиве являются структурами, то наименование структуры не должно присутствовать в сериализации JSON.

При наличии массива ссылок, т. е. когда тип "ref" относится к каждому элементу массива, каждый элемент массива должен быть сериализован как свойство href. Например массив "things" типа "ref[]" будет преобразован в последовательную форму следующим образом:

"things": [

{"href": *string*}, +

]?

Примечание — При сериализации массивов совместимые реализации не должны содержать пустых массивов (т. е. массивов, которые не содержат дочерних свойств) в сериализации JSON. При этом дочерний элемент свойства «things» определен с «+», что означает необходимость наличия по меньшей мере одного дочернего элемента. Это требование гарантирует, что сериализация JSON минимизирована и содержит элемент "things" («предметы») только в том случае, если в массиве присутствует "thing".

Сериализация XML:

При сериализации массивов в XML требуется, чтобы каждый элемент массива был представлен как отдельный элемент. Эти элементы должны быть последовательными и непрерывными в сериализации, и наименованием каждого элемента (наименованием тега) должно быть наименование типа элемента (наименование, которое появляется перед «[]» в типе массива). Например атрибут things («предметы») должен быть сериализован как перечень элементов с наименованием thing («предмет»), где "thing" — наименование структуры:

<thing>...

</thing> *

Для массива в XML нет ни одного внешнего оборачивающего элемента.

При наличии массива ссылок, т. е. когда тип "ref" относится к каждому элементу массива, массив сериализуется как перечень элементов XML без внешнего оборачивающего элемента. Наименование каждого элемента совпадает со значением «наименование элемента массива», указанном в определении атрибута. Например массив "things" типа "ref[]", где «наименование элемента массива» равно "thing", сериализуется следующим образом:

<thinghref = "xs:anyURI"/> +

5.5.12 Наборы

Как и массивы, Наборы представляют собой группы Ресурсов одного типа. В отличие от массивов Наборы являются самостоятельными Ресурсами, которые имеют свой собственный URI и к ним можно получить независимый доступ. Наборы также допускают оптимизированный и удобный сценарий взаимодействия, предоставляя специализированную совокупность операций, что позволяет избежать замены большого числа элементов при обновлении совокупности.

Настоящий стандарт распространяется на Наборы, в которых совокупность элементов в списке изменяется часто и потенциально многими Потребителями. С другой стороны, массивы используются, если ожидается, что перечень элементов не изменяется часто или его можно легко изменить путем замены всего перечня, то есть накладные расходы на выполнение служебных операций управления отдельными элементами как отдельными Ресурсы могут оказаться чрезмерными.

Атрибуты, которые являются Наборами, представлены как тип "collection[itemType]". Тип ресурса элементов Набора определен в скобках; например атрибут, который является Набором Ресурсов типа Machine, представлен как "collection[Machine]". Данные атрибуты сериализуются как ссылки на Ресурс Набора. Для краткости слово "ссылка" не используется в таблицах определения модели, несмотря на то, что все данные атрибуты являются ссылками: указывается тип "collection[itemType]".

Каждому из элементов Ресурсов будет соответствовать запись в Наборе. Предполагается, что эти элементы Ресурсов имеют составной тип, они независимо адресуемы и управляемы. Несмотря на то, что различные Наборы содержат записи различных Типов ресурсов, все Наборы соответствуют следующему шаблону:

- Наборы должны содержать атрибут id, который действует как «указатель на себя». Получение данных по этой ссылке должно возвратить Набор. В представлении XML каждый Набор должен быть обернут элементом <Collection>;

- Наборы должны содержать атрибут count, который содержит число Ресурсов в Наборе на момент, когда Набор был запрошен;

- Наборы должны содержать перечень Ресурсов, которые составляют Набор. Как и в случае массивов, при отсутствии Ресурсов в Наборе сериализация перечня должна быть опущена;

- аналогично Ресурсам в модели CIMI, каждый Ресурс в Наборе должен иметь атрибут id, который действует как «указатель на себя». Получение данных в этой ссылке должно возвратить этот единичный Ресурс, а не любой родительский Ресурс, такой как Набор или атрибут-массив;

- добавление новых Ресурсов к Набору производят с помощью операции "add", определенной в Наборе.

Примечание — Отсутствие операции "add" в Наборе указывает, что в это время новые Ресурсы не разрешены;

- удаление Ресурсов из Набора производят с помощью операции "delete" самого Ресурса;

- если не определено иное, при удалении Набора также должны быть удалены все Ресурсы, которые составляют Набор, при этом не должны быть удалены сторонние Ресурсы, на которые ссылаются Ресурсы из Набора, подлежащие удалению;

- Наборы должны быть удалены, если владеющий ими Ресурс удален.

В Наборе присутствуют Ресурсы двух видов:

- Ресурсы инфраструктуры (перечисленные в Точке входа в облако или встроенные в объект, такие как disk в Machine);

- Промежуточный Ресурс, который содержит ссылку на ресурс инфраструктуры, имеющий наименование «целевой Ресурс».

По соглашению, у промежуточных Ресурсов может быть наименование, связывающее наименование Ресурса, владеющего Набором, с наименованием целевого Ресурса, например MachineVolume (наименование промежуточного Ресурса, который используется для связи Machine с Volume).

Наборы промежуточных Ресурсов позволяют отделить жизненный цикл Набора (и его объекта-владельца) от жизненного цикла фактических целевых Ресурсов. Например при удалении Набора должны удаляться его промежуточные Ресурсы, но не целевые Ресурсы. Если ссылка на целевой Ресурс является обязательным атрибутом промежуточного Ресурса, жизненный цикл промежуточного Ресурса не должен быть длиннее, чем у целевого Ресурса.

- Если удален целевой Ресурс, то Поставщик также должен удалить любой промежуточный Ресурс, содержащий ссылку на этот Ресурс как на значение обязательного атрибута.

Сериализация Наборов должна соответствовать следующему шаблону:

Сериализация JSON:

{"resourceURI": *string,*
"id": *string*,
"count": *number*,
"*resourceSpecificGroupingName*": /* наименование атрибута зависит от типа Ресурсов */
[
{"resourceURI": *string*,
"id": *string*,
"name": *string,* ?
"description": *string,* ?
"created": *string,* ?
"updated": *string,* ?
"properties": {*string*: *string*, +}, ?
... данные ресурса...
"operations": [
{"rel": "edit", "href": *string*},?
{"rel": "delete", "href": *string*}?
] ?
...
} +
], ?
"operations": [{"rel": "add", "href": *string*}?]...
}

Сериализация XML:

<Collection resourceURI = "xs:anyURI" xmlns = "gost.e-ecolog.ru ">
<id> xs:anyURI </id>
<count> xs:integer </count>
<!-- наименование атрибута зависит от типа Ресурсов -->
<*ResourceSpecificElementName*>
<id> xs:anyURI </id>
<name> xs:string </name>?
<description> xs:string </description>?
<created> xs:dateTime </created>?
<updated> xs:dateTime </updated>?
<property key= "xs:string"> xs:string </property> *

... данные ресурса...
<operations rel="edit" href = "xs:anyURI"/>?
<operations rel="delete" href = "xs:anyURI"/>?
<xs:any> *
</*ResourceSpecificElementName*> *
< operations rel= "add" href = "xs:anyURI"/>?
<xs:any> *
</Collection>

В данном примере атрибуты resourceURI должны содержать URI Набора или URI, определенный для Ресурса этого типа Набора, а resourceSpecificGroupingName и ResourceSpecificElementName следует заменить на наименование Ресурса, определенного для Набора, например Machine в JSON или Machine в XML.

5.5.12.1 Добавление элементов к Наборам

Вызов операции "add" Набора должен добавить к Набору новый Ресурс. Содержимое тела запроса должно быть либо представлением нового Ресурса, добавляемого к Набору, либо представлением Шаблона, связанного с новым создаваемым Ресурсом. В настоящем стандарте установлено, какие Ресурсы требуют использования Шаблона.

Например для добавления нового Volume к Набору Volume Ресурса Machine, запрос операции "add" должен быть сериализован следующим образом:

Сериализация JSON:

{"resourceURI": "gost.e-ecolog.ru ",
"initialLocation": *string,*
"volume": {"href": *string*}
}

Сериализация XML:

<MachineVolume xmlns = "gost.e-ecolog.ru ">
<initialLocation> xs:string </initialLocation>
<volume href = "xs:string"/>
</MachineVolume>

Примечание — При удалении этого типа Ресурса из Набора Ресурс удаляется и убирается из этого Набора, при этом не должен удаляться сам целевой Ресурс, на который ссылаются, в данном примере —Volume.

При создании нового Ресурса, который требует использования Шаблона, операция «add» должна содержать:

- общие атрибуты в соответствии с 5.10.1;
- данные, определенные для Ресурса, необходимые для его создания. Эти данные должны представлять собой ссылку на Ресурс Шаблона, определенный для Ресурса, либо сам встроенный Ресурс Шаблона конкретного Ресурса;
- в случае XML — элемент обертки (получивший наименование по шаблону <*ResourceName*-Create>).

Например для создания новый экземпляра Machine (что требует использования Шаблона) и добавления его к MachineCollection операция «добавить» MachineCollection должна быть сериализована следующим образом:

Сериализация JSON:

{"resourceURI": "gost.e-ecolog.ru "?
"name": *string,*?
"description": *string,*?
"properties": {*string*: *string*, +},?
"machineTemplate": {"href": *string*?}...
}

Сериализация XML:

<MachineCreate xmlns = "gost.e-ecolog.ru ">
<name> xs:string </name>?
<description> xs:string </ description >?
<property key= "xs:string"> xs:string </property> *
<machineTemplate href = "xs:anyURI"?/>

<xs:any> *
</MachineCreate>
В MachineCollection добавляетсяновыйэкземплярMachine:
Сериализация JSON:
{"resourceURI": "gost.e-ecolog.ru ",
"id": string,
"name": string...
}
Сериализация XML:
<Machine xmlns = "gost.e-ecolog.ru ">
<id> xs:anyURI </id>
<name> xs:string </name...
</Machine>
Обработка операции "add" должна соответствовать семантике, определенной в 4.2.1.1.

Независимо от того, используется ли Шаблон, операция «add» должна создать новый Ресурс и добавить его к Набору, и ссылка (URI) на новую запись будет возвращена в ответном сообщении в заголовке HTTPLocation.

5.5.13 Тип "any"

Некоторые атрибуты являются полиморфными и могут содержать различные типы данных, перечень которых указан в их описании. В таких случаях в модели представления тип атрибута должен быть обозначен "any".

5.5.14 Пустые значения атрибута

Атрибуты следующих типов не приводят в случаях, если имеют пустое значение: строка, карта, массив и Набор. Помимо признаков «Необязательно для Поставщика» или «Необязательно для Потребителя», пустое значение является третьей причиной того, что схема сериализации содержит «?» или «*» для атрибута.

Другие типы атрибутов не имеют пустых значений, поэтому соответствующие атрибуты не должны быть опущены при сериализации.

5.6 Единицы

Некоторые Ресурсы, определенные в данной спецификации, имеют атрибуты, описывающие количество чего-либо, принадлежащего Ресурсу или связанного с ним. Например Ресурс Machine имеет атрибут memory, который описывает «размер памяти, выделенной для этой машины». Допустимые единицы измерения этих атрибутов приведены в таблице 4. Их значения установлены в [6]. Их числовые эквиваленты приведены для удобства Пользователей.

Таблица 4 — Числовые эквиваленты для атрибутов

Единица измерения	Значение	Единица измерения	Значение
Килобайт	10^3	Кибибайт	2^10
Мегабайт	10^6	Мебибайт	2^20
Гигабайт	10^9	Гибибайт	2^30
Терабайт	10^12	Тебибайт	2^40
Петабайт	10^15	Пепибайт	2^50
Эксабайт	10^18	Эксбибайт	2^60
Зеттабайт	10^21	Эебибайт	2^70
Йоттабайт	10^24	Йобибайт	2^80

5.7 Семантика отношений

Ссылка между двумя экземплярами Ресурса имеет семантику простой «ассоциации». Если не указано иное, (a) на экземпляр могут быть ссылки из других экземпляров Ресурсов, т. е. будет «совместно использоваться», и (b) — экземпляр Ресурса, на который указывает ссылка, не будет затронут

при удалении экземпляра Ресурса, который ссылается на него (т. е. операция Delete по умолчанию считается "shallowdelete").

Включение подресурса в другой Ресурс имеет семантику «агрегации» (или отношение целого к части в UML). Если не указано иное, (a) вложенный подресурс не может совместно использоваться несколькими экземплярами Ресурса и (b) — при удалении экземпляра Ресурса вложенные экземпляры подресурсов также удаляются.

5.8 Операции

Все операции Ресурса, определенные в настоящем стандарте, необязательны для их поддержки Поставщиками. Потребители могут определить, поддержка каких операций осуществляется; по метаданным Ресурса — ResourceMetadata. Однако даже для поддерживаемых операций Потребителям необходимо анализировать представление каждого Ресурса, чтобы определить, какие операции поддерживаются в тот или иной момент. Поддержка операции основывается на многих факторах, включая состояние Ресурса и права на управление доступом со стороны Потребителя. Также в соответствии с 4.2 операции и состояния объединены; т. е. при реализации операции, изменяющей состояние Ресурса, определенной в настоящем стандарте, также должно быть реализовано соответствующее(ие) состояние(я). Дополнительная информация приведена в пунктах «Операции» определения каждого Ресурса.

Атрибут «Состояние» тех Ресурсов, которые имеют данный атрибут, должен изменять значение только в следующих случаях:

- операция выполнена на данном Ресурсе, и эта операция требует изменения состояния или
- произошла ошибка, в этом случае атрибут «Состояния» должен получить значение «ОШИБКА».

Например для операции "start" Ресурса Machine требуется, чтобы состояния STARTING и STARTED поддерживались Machine одновременно, поскольку Machine будет отсутствовать в состоянии STARTED только после вызова другой операции, если не произошла ошибка.

Поставщики могут определить дополнительные операции и состояния. Такие расширения должны иметь одну из следующих категорий:

a) Новая операция, которая запускается из состояния, определенного CIMI, или приводит к состоянию, определенному CIMI, либо оба этих варианта. В последнем случае, если операция, определенная CIMI, уже существует для этого перехода между двумя состояниями, определенными CIMI, она также должна поддерживаться Поставщиком в дополнение к новой операции.

b) Новое состояние Ресурса. В этом случае также должна быть создана новая операция, которая приводит к этому состоянию. Другими словами, операция, определенная Поставщиками, должна выполняться прежде, чем может быть достигнуто состояние, определенное Поставщиками.

c) Новая операция, которая обеспечивает переходы между двумя состояниями, определенными Поставщиками.

5.9 Альтернативные форматы модели

В настоящем стандарте определение элементов модели представлено в альтернативных форматах, которые легко можно использовать с помощью инструментов для определенных технологий.

Данная модель также доступна в формате CIM/MOF [CIMI-CIM].

При возникновении несоответствий между различными форматами, требования установленные в настоящем стандарте, имеют приоритетное значение по сравнению со схемами XML и альтернативными форматами, которые, в свою очередь, имеют приоритет над примерами.

5.10 Ресурсы

В настоящем подразделе приведены детали атрибутов Ресурсов, определенных моделью CIMI.

5.10.1 Общие атрибуты

За исключением Ресурсов ResourceMetadata и Ресурсов Набора (см. 5.5.12) у всех Ресурсов, описанных в настоящем стандарте, есть общие атрибуты, приведенные в таблице 5. Для основных и вторичных ресурсов CIMI существуют различные требования. Все Ресурсы, которые могут быть типами элемента для Наборов в CloudEntryPoint, являются основными ресурсами CIMI. Все остальные Ресурсы являются вторичными ресурсами CIMI. Исключением из этого правила является CloudEntryPoint, который считается основным Ресурсом.

Например Machine является основным ресурсом CIMI, поскольку CloudEntryPoint имеет Набор, у которого Machine является типом элементов. Однако, например, MachineVolume является вторичным ресурсом CIMI, поскольку у CloudEntryPoint нет Набора, у которого MachineVolume был бы типом элемента.

Таблица 5 — Общие атрибуты

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>id</td><td>URI</td><td>Уникальный URI, определяющий данный Ресурс, присвоенный после создания Ресурса. Значение атрибута должно быть уникальным в облаке Поставщика.
<u>Ограничения для основных и вторичных Ресурсов:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>name</td><td>string</td><td>Наименование, читаемое Потребителем данного Ресурса, присвоенное создателем как часть входных данных при создании Ресурса.
<u>Ограничения для основных Ресурсов:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись.
<u>Ограничения для вторичных Ресурсов:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>description</td><td>string</td><td>Описание, читаемое Потребителемданного Ресурса, присвоенное создателем как часть входных данных при создании Ресурса.
<u>Ограничения для основных Ресурсов:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись.
<u>Ограничения для вторичных Ресурсов:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>created</td><td>dateTime</td><td>Метка времени, когда данный Ресурс был создан. Формат должен быть однозначным, а значение неизменяемым.
<u>Ограничения для основных и вторичных Ресурсов:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>updated</td><td>dateTime</td><td>Момент времени последнего явного обновления атрибута Ресурса.
Примечание — Несмотря на то, что такие операции как "stop" неявно изменяют атрибут "state", они не изменяют атрибуту "updated"
<u>Ограничения для основных и вторичных Ресурсов:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>properties</td><td>map</td><td>Отображение пар «ключ/значение» (каждая пара составляет отдельное «свойство»), некоторые из которых могут управлять одним или более аспектами этого Ресурса. Свойства могут также служить точкой расширения, позволяя Потребителям сохранить дополнительную информацию о Ресурсе.
Один и тот же ключ не должен использоваться более одного раза в рамках атрибута «properties».
Каждое свойство должно содержать следующие вложенные данные
<table>
<tr><td>Наименование</td><td colspan="2">property</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>ключ</td><td>string</td><td>Наименование свойства.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>значение</td><td>string</td><td>Значение свойства.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения для основных Ресурсов:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись.
<u>Ограничения для вторичных Ресурсов:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>

Следующие псевдосхемы описывают сериализацию данных атрибутов в JSON и в XML:

Сериализация JSON:

"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?

Сериализация XML:

<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *

5.11 Метаданные Ресурса

Потребитель в соответствии с настоящим стандартом должен иметь возможность обнаружения метаданных, связанных с каждым поддерживаемым типом Ресурса. Это позволяет Потребителям узнавать ограничения, наложенные Поставщиками на определенные атрибуты CIMI, а также обнаруживать дополнительные атрибуты или операции, которые могут быть определены Поставщиком. Экземпляр ResourceMetadata содержит метаданные, описывающие определенный тип Ресурса, например Network или Machine, включая любые определенные Поставщиком возможности или особенности. Механизм доступа к метаданным определяется протоколом.

Примечание — Несмотря на то что в настоящем стандарте метаданные ResourceMetadata считаются изменяемыми атрибутами, предполагается, что модифицировать их могут только пользователи, имеющие администраторские привилегии, связанные с Поставщиком. Поэтому для Потребителей они предоставляются в режиме «только для чтения».

Метаданные каждого Ресурса должны содержать сведения, приведенные в таблице 6.

Таблица 6 — Атрибуты ResourceMetadata

Наименование	ResourceMetadata	
Тип URI	gost.e-ecolog.ru /ResourceMetadata	
Атрибут	Тип	Описание
id	*URI*	Уникальный URI, определяющий данный Ресурс, присвоенный после создания Ресурса. Это значение атрибута является **неизменяемым** и в облаке Поставщика должно быть **уникальным**. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
typeURI	*URI*	Уникальный URI, связанный с описываемым типом Ресурса и обозначающий его. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
name	*String*	Наименование описываемого типа Ресурса. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
attributes	*attribute*	Ряд метаданных, определенных Поставщиками, которые могут использоваться Потребителями для обнаружения любых метаданных, связанных с каждым атрибутом описанного типа Ресурса, включая совокупность дополнительных атрибутов, не определенных в этой спецификации.

Продолжение таблицы 6

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>attributes</td><td>attribute</td><td>Каждый атрибут должен содержать следующие вложенные данные
<table>
<tr><td>Наименование</td><td colspan="2">Атрибут</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>name</td><td>string</td><td>Наименование атрибута.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>namespace</td><td>URI</td><td>Пространство имен, в котором определен данный атрибут. Рекомендуется, чтобы при разыменовывании данного URI происходило возвращение информации об атрибуте. Оно не должно присутствовать при описании атрибута, определенного CIMI, но должно присутствовать при описании атрибута, определенного вне CIMI.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>type</td><td>string</td><td>Тип данных атрибута. Не должен присутствовать при описании атрибута, определенного CIMI, но должен присутствовать при описании атрибута, определенного вне CIMI.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>required</td><td>Boolean</td><td>Указывает, требует ли данный Ресурс присутствия данного атрибута. Если он отсутствует, то подразумеваемое значение «false».
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>value constraints</td><td>any</td><td>Данные, определенные типом, которые описывают любые ограничения на значения этого атрибута. Если они отсутствуют, то ограничений нет.
Примечание — Сериализация ограничений на значения должна быть определена типом атрибута (см. 5.11.1).
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>capabilities</td><td>capability</td><td>Ряд метаданных, определенных Поставщиком, которые могут применять функции, предоставленные этим Поставщиком.
Каждая возможность должна содержать следующие вложенные использоваться Потребителем для обнаружения возможности</td></tr>
<tr><td>capabilities</td><td>capability</td><td>
<table>
<tr><td>Наименование</td><td colspan="2">capbility</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>name</td><td>string</td><td>Наименование возможности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>
</td></tr>
</table>

Продолжение таблицы 6

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td></td><td></td><td>
<table>
<tr><td>Наименование</td><td colspan="2">capbility</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>uri</td><td>URI</td><td>URI, который однозначно определяет возможность атрибута на глобальном уровне.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>description</td><td>string</td><td>Описание семантики возможности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>value</td><td>any</td><td>Значение возможности. Конкретный тип зависит атрибута от определения возможности. Если значения не установлены, по умолчанию возможности считаются типа «булевый» со значением "true", указывая, что конкретная возможность поддерживается Поставщиком.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись
</td></tr>
<tr><td>actions</td><td>action []</td><td>Ряд операций, определенных Поставщиком, которые могут использоваться Потребителями для воздействия на Ресурс. Эта совокупность представляет все операции, определенные для данного типа Ресурса, который может быть надмножеством операций, фактически используемых конкретным Потребителем. Подмножество разрешенных операций для конкретного Потребителя должно представлять собой те операции, которые возвращены этому Потребителю при запросе экземпляра описанного типа Ресурса.
Примечание — данный атрибут вызывается "actions", чтобы не конфликтовать с собственным атрибутом "operations" Ресурса ResourceMetadata.
Каждая операция должна содержать следующие вложенные данные
<table>
<tr><td>Наименование</td><td colspan="2"></td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>name</td><td>string</td><td>Наименование операции.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>uri</td><td>URI</td><td>URI, однозначно определяющий операцию на глобальном уровне.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>description</td><td>string</td><td>Описание семантики операции.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>method</td><td>string</td><td>Глагол для выполнения операции, зависящий от протокола
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
</td></tr>
</table>

Окончание таблицы 6

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>actions</td><td>action []</td><td>
<table>
<tr><td>Наименование</td><td colspan="2"></td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>Input Message</td><td>string</td><td>mimeType тела сообщения запроса; может зависеть от формата модели, выбранного Поставщиком.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>output-Message</td><td>string</td><td>mimeType тела сообщения-ответа; может зависеть от формата модели, выбранного Поставщиком.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись
</td></tr>
</table>

При реализации или использовании ResourceMetadata Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов в соответствии с таблицей 6, а также таблицами, описывающими встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Следующие псевдосхемы описывают сериализацию Ресурса и в JSON и в XML (см. 1.3):

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"typeURI": *string*,
"name": *string*,
"attributes" : [
{ "name": *string*,
"namespace": *string*, ?
"type": *string*, ?
"required": *boolean*, ?
...value constraints...? } *
], ?
"capabilities": [
{ "name": *string*, ?
"uri": *string*,
"description": *string*, ?
"value": *any* } *
], ?
"actions" : [
{ "name": *string*,
"uri": *string*,
"description": *string*, ?
"method": *string*,
"inputMessage": *string*, ?
"outputMessage": *string* ? }, *
], ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml**:**
Сериализация XML:
<ResourceMetadata xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name>
<typeURI> xs:anyURI </typeURI>
<attribute name="*xs:string*" namespace="*xs:anyURI*"? type="*xs:string*"
required="*xs:boolean*"? >
...value constraints...?
</attribute> *
<capability name="*xs:string*"? uri="*xs:anyURI*" description="*xs:string*"?>
*xs:any**
</capability> *
<action name="*xs:string*" uri="*xs:anyURI*" description="*xs:string*"?
method="*xs:string*" inputMessage="*xs:string*"?
outputMessage="*xs:string*"? /> *
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<xs:any>*
</ResourceMetadata>
Поставщик может включать дополнительные метаданные Ресурса или атрибутов.

5.11.1 Сериализация ограничений значения атрибута

Следующие примеры описывают значения, синтаксис и сериализацию значения "value constraints" (податрибут "attributes"), имеющего тип "any".

type ="string"
JSON должен иметь форму:
"values": [*string*, +] ?
XML должен иметь форму:
<value>*xs:string*</value> *
type="integer"
JSON должен иметь форму:
"values": [*number*, +], ?
"ranges": [{ "low": *number*, "high": *number*}, +] ?
XML должен иметь форму:
<value>*xs:integer*</value> *
<range low="*xs:integer*" high="*xs:integer*"/> *
Итоговое пространство значений атрибута "integer" – объединение всех значений и диапазонов.
type = "boolean"
JSON должен иметь форму:
"value": *boolean*?
XML должен иметь форму:
<value>*xs:boolean*</value> ?
Допускается только одно значение "value", которое будет указывать на то, какой из атрибутов требуется: "true" или "false".

5.11.1.1 Примеры

В следующем примере представлен образец документа метаданных для Ресурса VolumeConfiguration в XML, перечисляющий допустимые значения для атрибута "format" и содержащий развернутый строковый атрибут "Location":

<ResourceMetadata xmlns="gost.e-ecolog.ru ">
<id> gost.e-ecolog.ru </id>
<typeURI> gost.e-ecolog.ru </typeURI>
<name> VolumeConfiguration </name>
<attribute name="format" type="string" required="false">
<value> ext4 </value>
<value> ntfs </value>
</attribute>

```
<attribute name="Location" namespace="gost.e-ecolog.ru ype="string"/>
</ResourceMetadata>
```

В следующем примере представлен тот же Ресурс VolumeConfiguration, в котором атрибут "Location" ограничен рядом значений и является обязательным:

```
<ResourceMetadata xmlns="gost.e-ecolog.ru ">
<id> gost.e-ecolog.ru </id>
<typeURI> gost.e-ecolog.ru </typeURI>
<name> VolumeConfiguration </name>
<attribute name="format" type="string" required="false">
<value> ext4 </value>
<value> ntfs </value>
</attribute>
<attribute name="Location" namespace="gost.e-ecolog.ru ype="string"
required="true">
<value> NYC </value>
<value> LAX </value>
</attribute>
</ResourceMetadata>
```

В следующем примере указан тот же самый Ресурс VolumeConfiguration, сериализованный в JSON:

```
{ "resourceURI": "gost.e-ecolog.ru ",
"id": "gost.e-ecolog.ru ",
"typeURI": "gost.e-ecolog.ru ",
"name": "VolumeConfiguration",
"attributes": [
{ "name": "format",
"type": "string",
"required": false,
"values": [ "ext4", "ntfs" ]
},
{ "name": "Location",
"namespace": "gost.e-ecolog.ru
"type": "string",
"required": true,
"values": [ "NYC", "LAX" ]
}
]
}
```

В следующем примере представлен Ресурс Volume, сериализованный в JSON, который обеспечивает действие сжатия данных. В данном примере возвращенный метод (POST) предназначен для протокола HTTPCIMI, однако при реализации другого протокола (например, SOAP), метод будет иным:

```
{ "resourceURI": "gost.e-ecolog.ru ",
"id": "gost.e-ecolog.ru ",
"typeURI": "gost.e-ecolog.ru ",
"name": "Volume",
"actions": [
{
   "name": "compress",
   "uri": "gost.e-ecolog.ru "
   "description": "Compress the data stored in the Volume",
   "method": "POST"
}
]
}
```

5.11.2 Возможности

URI возможностей, определенных в настоящем стандарте, приведены в таблице 7. Поставщики могут определить новые URI, причем рекомендуется, чтобы данные URI указывали на документы с опи-

санием, по которому Потребители могут получить сведения о новой возможности. Графа «Наименование Ресурса» содержит наименование Ресурса, в чьем ресурсе ResourceMetadata может содержаться указанная возможность. Графа «Наименование возможности» содержит наименование указанной возможности, которое должно быть уникальным в рамках соответствующего Ресурса. URI каждой возможности создается путем конкатенации «Наименовании Ресурса», наклонной черты (/) и «Наименовании Возможности» к gost.e-ecolog.ru . Например возможность «InitialState» Ресурса Machine должна иметь URI: gost.e-ecolog.ru

Возможности, которые относятся к Поставщику в целом, определены не для одного Ресурса, должны быть связаны с Ресурсом CloudEntryPoint (если возможность применяется только к самому Ресурсу CloudEntryPoint, на это указывает его определение).

Каждая из этих возможностей может быть установлена равной конкретному значению, либо может отсутствовать. Значение отсутствующей возможности определяется следующим образом:

- для возможностей булевского типа: то же, что и значение "false";
- для других возможностей, использующих единичное значение или перечень значений в перечислении: аналогично «нет особых предпочтений или ограничений, осуществляемых для этого значения».

Таблица 7 — URI возможностей

Наименование ресурса	Наименование возможности	Описание
CloudEntryPoint	ExpandParameter	Если "true", Поставщик должен поддерживать параметр запроса $expand
CloudEntryPoint	FilterParameter	Если "true", Поставщик должен поддерживать параметр запроса $filter
CloudEntryPoint	FirstParameter	Если "true", Поставщик должен поддерживать оба параметра запроса: $first и $last
CloudEntryPoint	SelectParameter	Если "true", Поставщик должен поддерживать параметр запроса $select
CloudEntryPoint	FormatParameter	Если "true", Поставщик должен поддерживать параметр запроса $format
CloudEntryPoint	OrderByParameter	Если "true", Поставщик должен поддерживать параметр запроса $orderby
CloudEntryPoint	QueryPathNotation	Если "true", Поставщик должен поддерживать использование нотации путей в параметре запроса $select (см. 4.1.6.3) для разрешения неоднозначности между атрибутами Ресурса Набора и атрибутами отдельных элементов Набора
CloudEntryPoint	MaxPropertyItems	Если установлено, Поставщик должен поддерживать атрибут "properties" с числом элементов, не превосходящим значение, определенное данной возможностью
System	SystemComponent-TemplateByValue	Если "true", Поставщик должен поддерживать спецификацию включения ComponentTemplate по значению в SystemTemplates
Machine	DefaultInitialState	Если данная возможность установлена и не указано иное (например, атрибутом "initialState" ресурса MachineTemplate), Поставщик должен установить состояние нового экземпляра Machine равным значению данной возможности при условии, что значение совместимо с возможностью InitialStates
Machine	InitialStates	Если данная возможность установлена и используется MachineTemplate с атрибутом "initialState", Потребитель должен использовать значение initialState из совокупности значений данной возможности
Machine	MachineConfig-ByValue	Если "true", Поставщик должен поддерживать включение Machine-Configuration по значению, а MachineTemplateByValue также должно иметь значение "true"

Продолжение таблицы 7

Наименование ресурса	Наименование возможности	Описание
Machine	MachineCredential-ByValue	Если "true", Поставщик должен поддерживать включение Credential по значению в операциях create для Machine, а MachineTemplateBy-Value также должна иметь значение "true"
Machine	MachineImageBy-Value	Если "true", Поставщик должен поддерживать включение MachineImage по значению в операциях create для Machine, а MachineTemplate-ByValue также должна иметь значение "true"
Machine	MachineVolume-TemplatesByValue	Если "true", Поставщик должен поддерживать включение VolumeTemplate по значению в операциях create для Machine, а MachineTemplateByValue также должна иметь значение "true"
Machine	MachineTemplateBy Value	Если "true", Поставщик должен поддерживать включение MachineTemplate по значению в операциях create для Machine
Machine	MachineStopForce	Если "true", Поставщик должен поддерживать опцию "force" операций stop и restart для Machine
Machine	MachineStopForce Default	Если "true", Поставщик должен принудительно останавливать экземпляры Machine (имитируя отключение питания), если нет иных указаний. В другом случае Поставщик должен использовать регламентные процедуры остановки
Machine	RestoreFromImage	Если "true", Поставщик должен поддерживать восстановление экземпляров Machine из MachineImage, не являющихся снимком состояния
Machine	UserData	Если данная возможность установлена, это означает, какой метод должен использоваться Поставщиком для внедрения пользовательских данных
Machine	MachineAvailability Level	Если "true", Поставщик поддерживает понятие уровня готовности для Ресурса Machine. Уровень готовности и ограничения к его значениям доступны как атрибут расширения ResourceMetadata для Machine и MachineTemplate
Credential	CredentialTemplate-By-Value	Если "true", Поставщик должен поддерживать включение CredentialTemplate по значению в операциях create для Credential
Volume	SharedVolume-Support	Если "true", Поставщик должен поддерживать возможность совместного использования экземпляра Ресурса Volume несколькими экземплярами Machine
Volume	VolumeConfigBy-Value	Если "true", Поставщик должен поддерживать включение VolumeConfiguration по значению в операциях create для Volume, а VolumeTemplateByValue должна иметь значение "true"
Volume	VolumeImageBy-Value	Если "true", Поставщик должен поддерживать включение VolumeImage по значениию в операциях create для Volume, а VolumeTemplateByValue должна иметь значение "true"
Volume	VolumeSnapshot	Если "true", Поставщик должен поддерживать создание нового VolumeImage, со ссылкой на существующий Volume
Volume	VolumeTemplate-By-Value	Если "true", Поставщик должен поддерживать включение VolumeTemplate по значению в операциях create для Volume
Volume	VolumeAvailability Level	Если "true", Поставщик должен поддерживать понятие уровня готовности для Ресурса Volume. Уровень готовности и ограничения к его значениям доступны как атрибут расширения ResourceMetadata для Volume и VolumeTemplate
Network	NetworkConfigBy-Value	Если "true", Поставщик должен поддерживать включение NetworkConfiguration по значению в операциях create для Network

Окончание таблицы 7

Наименование ресурса	Наименование возможности	Описание
Network	NetworkTemplate-ByValue	Если "true", Поставщик должен поддерживать включение NetworkTemplate по значению в операциях create для Network
Network	DefaultInitialState	Если данная возможность установлена и не указано иное (например, атрибутом "initialState" Ресурса NetworkTemplate), Поставщик должен установить состояние нового экземпляра Network равным значению данной возможности при условии, что значение совместимо с возможностью InitialState
Network	InitialStates	Если данная возможность установлена и используется NetworkTemplate с атрибутом "initialState", то Потребитель должен использовать значение initialState из совокупности значений этой возможности
NetworkPort	NetworkPort-ConfigByValue	Если "true", Поставщик должен поддерживать включение NetworkPortConfiguration по значению в операциях create для NetworkPort
NetworkPort	NetworkPort-TemplateByValue	Если "true", Поставщик должен поддерживать включение NetworkPortTemplate по значению в операции create для NetworkPort
NetworkPort	DefaultInitialState	Если данная возможность установлена и не указано иное (например, атрибутом "initialState" Ресурса NetworkPortTemplate), Поставщик должен установить состояние нового экземпляра NetworkPort равным значению данной возможности при условии, что значение совместимо с возможностью InitialStates
NetworkPort	InitialStates	Если данная возможность установлена и используется NetworkPortTemplate с атрибутом "initialState", Потребитель, должен использовать значение initialState из совокупности значений этой возможности
ForwardingGroup	MixedNetwork	Если "true", Поставщик должен поддерживать ForwardingGroup, которые могут иметь как частные, так и общедоступные соединения одновременно. В противном случае у ForwardingGroup должны быть только либо частные, либо общедоступные соединения в одно и то же время
Job	JobRetention	Если данная возможность установлена, ее значение должно указывать на минимальное число минут, в течение которых задание должно храниться Поставщиком, прежде чем оно будет удалено
Meter	MeterConfigByValue	Если "true", Поставщик должен поддерживать включение MeterConfiguration по значению в операциях create для Meter
Meter	MeterTemplate-ByValue	Если "true", Поставщик должен поддерживать включение MeterTemplate по значению в операциях create для Meter
EventLog	Соединенный	Если "true", Поставщик должен удалить ресурсы EventLog, связанные с Ресурсами, если Ресурс удален

В следующих примерах показаны ResourceMetadata для Machine, предоставляющие информацию о некоторых возможностях:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": "gost.e-ecolog.ru ",
"typeURI": "gost.e-ecolog.ru ",
"name": "Machine",
"capabilities": [
{ "uri":
"gost.e-ecolog.ru ",
"value": true },
{ "uri":

"gost.e-ecolog.ru ",
"value": true },
{ "uri":
"gost.e-ecolog.ru ",
"value": "STARTED" }
}
}

Сериализация XML:

<ResourceMetadata xmlns="gost.e-ecolog.ru ">
<id> gost.e-ecolog.ru </id>
<typeURI> gost.e-ecolog.ru </typeURI>
<name> Machine </name>
<capability
uri="gost.e-ecolog.ru ">
true
</capability>
<capability
uri="gost.e-ecolog.ru ">
true
</capability>
<capability
uri="gost.e-ecolog.ru ">
STARTED
</capability>
</ResourceMetadata>

5.11.3 Ресурс ResourceMetadataCollection

Ресурс ResourceMetadataCollection представляет собой Набор Ресурсов ResourceMetadata Поставщика и соответствует спецификации Набора, определенной в 5.5.12.

Примечание — Изменения Ресурсов в пределах данного Набора могут осуществляться только Потребителями CIMI с администраторским уровнем доступа. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"resourceMetadatas": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
...остальные атрибуты ResourceMetadata...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /ResourceMetadataCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<ResourceMetadata>
<id>*xs:anyURI*</id>
... остальные атрибуты ResourceMetadata...
</ResourceMetadata> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.12 Точка входа в облако

Точка входа в облако (Ресурс CloudEntryPoint) представляет собой точку входа в облако, определенную моделью CIMI. Точка входа в облако реализует каталог Ресурсов, таких как System, SystemTemplate, Machine, MachineTemplate и т.д., которые могут запрашиваться и просматриваться Потребителем.

CloudEntryPoint и связи этого Ресурса с другими Ресурсами приведены на рисунке 1, который представлен в виде схемы отношений Ресурсов, при этом использование UML не является строгим и нормативным.

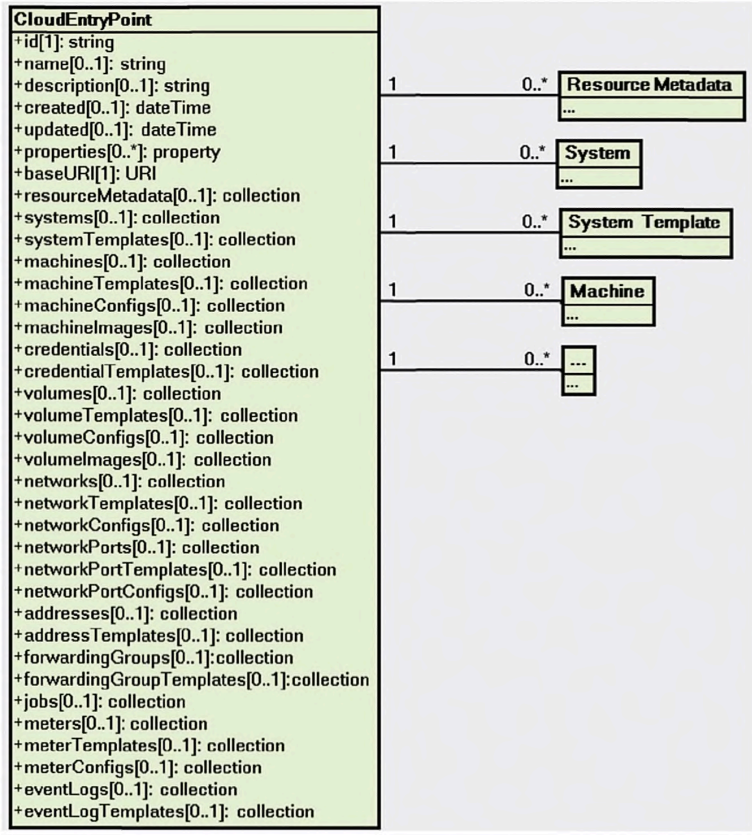

Рисунок 1 — Точка входа в облако

Если Потребитель запрашивает чтение Ресурса CloudEntryPoint, Поставщик должен возвратить Ресурс CloudEntryPoint, который каталогизирует лишь те Ресурсы, с которыми данному Потребителю разрешено выполнять операции. Атрибуты Ресурса CloudEntryPoint приведены в таблице 8.

Таблица 8 — Атрибуты CloudEntryPoint

Наименование	CloudEntryPoint	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
baseURI	*URI*	Абсолютный URI, который указывает на «базовый URI» Поставщика. Этот URI должен использоваться для преобразования относительных URI для Ресурсов в рамках данного Поставщика в абсолютные URI (см. 5.5). **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения

Продолжение таблицы 8

Атрибут	Тип	Описание
Resource Metadata	*collection [Resource Metadata]*	Ссылка на Набор Ресурсов ResourceMetadata данной Точки входа в облако. Набор содержит описание Ресурсов, поддерживаемых Поставщиком. Если Ресурс не имеет метаданных, он не должен появляться в списке, например, он не имеет ограничений помимо определенных спецификацией CIMI, а также не имеет никаких атрибутов расширения. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
systems	*collection [System]*	Ссылка на SystemCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
System Templates	*collection [System Template]*	Ссылка на SystemTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
machines	*collection [Machine]*	Ссылка на MachineCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Machine Templates	*collection [Machine Template]*	Ссылка на MachineTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
machineConfigs	*collection [Machine Configuration]*	Ссылка на MachineConfigurationCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
machineImages	*collection [Machine Image]*	Ссылка на MachineImageCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
credentials	*collection [Credential]*	Ссылка на CredentialCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Credential Templates	*collection [Credential-Template]*	Ссылка на CredentialTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
volumes	*collection [Volume]*	Ссылка на VolumeCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Volume Templates	*collection [VolumeTemplate]*	Ссылка на VolumeTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Volume Configs	*collection [Volume Configuration]*	Ссылка на VolumeConfigurationCollection данной Точки входа в облако **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Продолжение таблицы 8

Атрибут	Тип	Описание
volumeImages	*collection [VolumeImage]*	Ссылка на VolumeImageCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networks	*collection [Network]*	Ссылка на NetworkCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Network Templates	*collection [Network Template]*	Ссылка на NetworkTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Network Configs	*collection [Network Configuration]*	Ссылка на NetworkConfigurationCollection данной Точки входа в облако **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networkPorts	*collection [NetworkPort]*	Ссылка на NetworkPortCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networkPort Templates	collection [NetworkPort *Template]*	Ссылка на NetworkPortTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networkPort Configs	*collection [NetworkPort Configuration]*	Ссылка на NetworkPortConfigurationCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
addresses	*collection [Address]*	Ссылка на AddressCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Address Templates	collection [Address Template]	Ссылка на AddressTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
Forwarding Groups	collection [Forwarding Group]	Ссылка на ForwardingGroupCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
forwardingGroup-Templates	*collection [Forwarding Group Template]*	Ссылка на ForwardingGroupTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
jobs	*collection[Job]*	Ссылка на JobsCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Окончание таблицы 8

Атрибут	Тип	Описание
meters	*collection [Meter]*	Ссылка на MeterCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meterTemplates	*collection [Meter Template]*	Ссылка на MeterTemplateCollectionданной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meterConfigs	*collection [Meter Configuration]*	Ссылка на MeterConfigurationCollectionданной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLogs	*collection [EventLog]*	Ссылка на EventLogCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLogTemplates	*collection [EventLog Template]*	Ссылка на EventLogTemplateCollection данной Точки входа в облако. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Каждый из Наборов, указанных в таблице 8, определен в пунктах, определяющих связанные с ними ресурсы. Например Ресурс MachineCollection определен в 5.14.2 как часть Ресурсов, связанных с Machine.

Реализуя или используя CloudEntryPoint Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов, согласно таблице 8, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии с описанием, приведенным далее. Сериализация Ресурса в JSON и в XMLприведена в псевдосхемах (см. 1.3.):

Тип медиа JSON:application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"baseURI": *string*,
"resourceMetadata": { "href": *string* }, ?
"systems": { "href": *string* }, ?
"systemTemplates": { "href": *string* }, ?
"machines": { "href": *string* }, ?
"machineTemplates": { "href": *string* }, ?
"machineConfigs": { "href": *string* }, ?
"machineImages": { "href": *string* }, ?
"credentials": { "href" string }, ?
"credentialTemplates": { "href" string }, ?
"volumes": { "href": string }, ?
"volumeTemplates": { "href": string }, ?
"volumeConfigs": { "href": string }, ?
"volumeImages": { "href": string }, ?

```
"networks": { "href": string }, ?
"networkTemplates": { "href": string }, ?
"networkConfigs": { "href": string }, ?
"networkPorts": { "href": string }, ?
"networkPortTemplates": { "href": string }, ?
"networkPortConfigs": { "href": string }, ?
"addresses": { "href": string }, ?
"addressTemplates": { "href": string }, ?
"forwardingGroups" { "href": string }, ?
"forwardingGroupTemplates" { "href": string }, ?
"jobs": { "href": string }, ?
"meters": { "href": string }, ?
"meterTemplates": { "href": string }, ?
"meterConfigs": { "href": string }, ?
"eventLogs": { "href": string }, ?
"eventLogTemplates": { "href": string }, ?
"operations": [
{ "rel": "edit", "href": string } ?
] ?
...
}
```

Тип медиа XML: application/xml

Сериализация XML:

```
<CloudEntryPoint xmlns="gost.e-ecolog.ru                    ">
<id>xs:anyURI</id>
<name>xs:string</name> ?
<description>xs:string</description> ?
<created>xs:dateTime</created> ?
<updated>xs:dateTime</updated> ?
<property key="xs:string">xs:string</property> *
<baseURI>xs:anyURI</baseURI>
<resourceMetadata href="xs:anyURI"/> ?
<Systems href="xs:anyURI"/> ?
<SystemTemplates href="xs:anyURI"/> ?
<Machines href="xs:anyURI"/> ?
<MachineTemplates href="xs:anyURI"/> ?
<MachineConfigs href="xs:anyURI"/> ?
<MachineImages href="xs:anyURI"/> ?
<Credentials href="xs:anyURI"/> ?
<CredentialTemplates href="xs:anyURI"/> ?
<Volumes href="xs:anyURI"/> ?
<VolumeTemplates href="xs:anyURI"/> ?
<VolumeConfigs href="xs:anyURI"/> ?
<VolumeImages href="xs:anyURI"/> ?
<networks href="xs:anyURI"/> ?
<networkTemplates href="xs:anyURI"/> ?
<networkConfigs href="xs:anyURI"/> ?
<networkPorts href="xs:anyURI"/> ?
<networkPortTemplates href="xs:anyURI"/> ?
<networkPortConfigs href="xs:anyURI"/> ?
<addresses href="xs:anyURI"/> ?
<addressTemplates href="xs:anyURI"/> ?
<forwardingGroups href="xs:anyURI"/> ?
<forwardingGroupTemplates href="xs:anyURI"/> ?
<jobs href="xs:anyURI"/> ?
```

<meters href="xs:anyURI"/> ?
<meterTemplates href="xs:anyURI"/> ?
<meterConfigs href="xs:anyURI"/> ?
<EventLogs href="xs:anyURI"/> ?
<EventLogTemplates href="xs:anyURI"/> ?
<operation rel="edit" href="xs:anyURI"/> ?
<xs:any>*
</CloudEntryPoint>

5.12.1 Операции

Данный Ресурс поддерживает операции Read и Update.

5.13 Ресурс System и отношения

Ресурсы, составляющие ресурс System, и связи между ними, приведены на рисунке 2, который представлен в виде схемы отношений Ресурсов, при этом использование UML не является обязательным требованием.

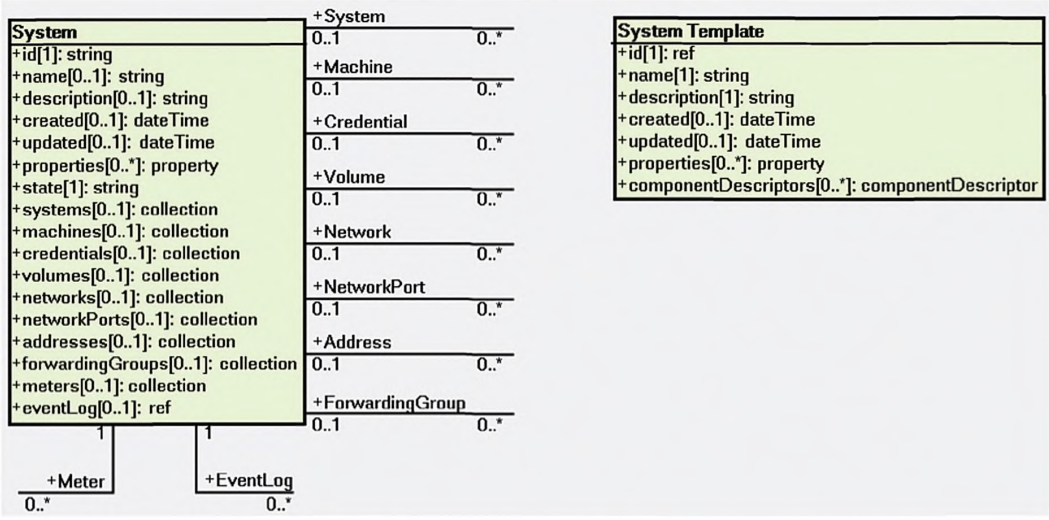

Рисунок 2 — Ресурсы системы

5.13.1 System

System (Система) — реализованный Ресурс, который содержит один или более ресурсов Network (Сеть), Volume (Том), Machine (Машина) и т.д., которые могут связываться и соединяться друг с другом. Ресурс System может создаваться из интерпретации SystemTemplate. Допускается управлять Ресурсом System как единичным Ресурсом; также данный Ресурс формирует стек служб. Например система корзины в интернет-магазине состоит из машин для веб-серверов и базы данных, сетевых адресов для открытого доступа и томов для файлов базы данных. System может непосредственно предоставлять компонент, с которым взаимодействуют пользователи, либо предоставлять инфраструктурный компонент.

System имеет несколько атрибутов «верхнего уровня», которые являются Наборами ссылок на Ресурсы, принадлежащих System. Жизненный цикл Ресурсов, принадлежащих System, связан непосредственно с жизненным циклом System. В частности, если System будет удалена, то все находящиеся в ее ведении Ресурсы должны также быть удалены. Обычно операции cSystem переводятся в операции с ее Ресурсами.

Однако Ресурс, принадлежащий System, в свою очередь, может относиться к некоторым другим Ресурсам, не принадлежащим этой System. Например Machine в System может ссылаться на Volume, который не принадлежит данной System. Эти правила применяются следующим образом:

- по умолчанию все Ресурсы, являющиеся результатом создания System, также принадлежат System. (Это правило может быть переопределено при последующих изменениях в высокоуровневых атрибутах Набора System);

- владение Ресурсом для System выражено таким образом, что ссылка на Ресурс включается в соответствующий Набор в атрибуте верхнего уровня Ресурса System, либо это происходит вследствие того, что данный Ресурс System владеет подсистемой (т. е. транзитивное владение через иерархии ресурсов System);

- если к существующей System добавляется Ресурс, отличающийся от данной System (т. е. принадлежит System после включения его ссылки в соответствующий Набор из атрибута верхнего уровня System), то другие Ресурсы, на которые уже ссылается добавленный Ресурс, по умолчанию не принадлежат System. (Данное правило может быть переопределено при последующих изменениях в Наборах высокоуровневых атрибутов System.)

Ресурс не должен принадлежать более чем одной System ни в один момент времени, (если между этими System нет отношений принадлежности).

Примечание — Ресурсу необязательно принадлежать System (т. е. являться частью любого из атрибутов ее Набора), чтобы на него ссылались или использовали в System. Если Ресурс не включен в любой из Наборов, то он просто не используется в любых действиях, выполняемых в System.

Описание атрибутов Системы приведено в таблице 9.

Таблица 9 — Атрибуты Системы

Наименование	Система (System)	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние Системы. Допустимыми значениями являются: см. 5.14.1. **CREATING**: Система находится в процессе создания. **STARTING/STARTED/STOPPING/STOPPED/PAUSING/PAUSED/SUSPENDING /SUSPENDED:**System должна находиться в одном из данных состояний, если все ресурсы Machine, на которые ссылается System, находятся в этом состоянии. Перечень доступных действий, основанных на состоянии Machinecм. в 5.14.1. Такие транзитные состояния могут просто на то, что все Machine в System подвергаются тем же операциям (например, «start») без необходимости выполнения конкретной операции в System (например, операция «start» не выполняется на уровне System). Фактическая операция в System может быть прослежена при запросе объекта «job». **MIXED:** System должна быть в этом состоянии, если нет никаких ссылок на Machine от данной System, либо Machine, на которые ссылается данная System, находятся в разных состояниях. Разные состояния могут возникнуть, когда в System выполняются операции, приводящие к переходу состояний принадлежащих ей Ресурсов Machine к новому общему состоянию (например, STOPPED, STARTED), но они происходят с разной скоростью или последовательно одна за другой. **DELETING**: System находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в System. Операции, приводящие к возникновению указанных состояний, определены в 5.13.1.2. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
systems	*Collection [System System]*	Ссылка на перечень ссылок на вложенные Ресурсы типа System, принадлежащие данной System. Добавление элемента (типа System) к этому перечню логически эквивалентно присоединению System, на которую ссылаются, к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению System, на которую указывает ссылка, от данной System. Примечание — Тип ресурса SystemSystem представляет собой связь между System и другой System(см. 5.13.1.1.1). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Продолжение таблицы 9

Атрибут	Тип	Описание
machines	*Collection [System Machine]*	Ссылка на перечень ссылок на Machine , принадлежащих данной System. Добавление элемента (типа Machine) к этому перечню логически эквивалентно присоединению Machine к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению Machine от данной System. Примечание — Тип ресурса SystemMachine представляет собой связь между System и Machine (см. 5.13.1.1.2). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
credentials	*Collection [System Credential]*	Ссылка на перечень ссылок на Credential, принадлежащих данной System. Добавление элемента (типаCredential) к данному перечню логически эквивалентно присоединению Credential к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению Credential от данной System. Примечание — Тип ресурса SystemCredential представляет собой связь между System и Credential(см. 5.13.1.1.3). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
volumes	*Collection [System Volume]*	Ссылка на перечень ссылок на Volume, принадлежащих данной System. Добавление элемента (типа Volume) к данному перечню логически эквивалентно присоединению Volume к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению Volumea от данной System. Примечание — Тип ресурса SystemVolume представляет собой связь между System и Volume(см. 5.13.1.1.4). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networks	*Collection [System Network]*	Ссылка на перечень ссылок на Network, принадлежащих данной System. Добавление элемента (типа Network) к данному перечню логически эквивалентно присоединению Network к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению Network от данной System. Примечание — Тип ресурса SystemNetwork представляет собой связь между System и Network (см. 5.13.1.1.5). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
network Ports	*Collection [System Network Port]*	Ссылка на перечень ссылок NetworkPort, принадлежащих данной System. Добавление элемента (типа NetworkPort) к данному перечню логически эквивалентно присоединению NetworkPort к данной System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению NetworkPort от данной System. Примечание — Тип ресурса SystemNetworkPort представляет собой связь между System и NetworkPort(см. 5.13.1.1.6). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Окончание таблицы 9

Атрибут	Тип	Описание
addresses	*Collection [System Address]*	Ссылка на перечень ссылок на Address, принадлежащих данной System. Добавление элемента (типа Address) к этому перечню логически эквивалентно присоединению Addres к этой System в соответствии с семантикой «отношение включения». Удаление элемента из этого перечня логически эквивалентно отсоединению Addres от данной System. Примечание — Тип ресурса SystemAddres представляет собой связь между System и Address(см. 5.13.1.1.7). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
forwarding Groups	*Collection [System Forwarding Groups]*	Ссылка на перечень ссылок на ForwardingGroup, принадлежащих данной System. Добавление элемента (типа ForwardingGroup) к данному перечню логически эквивалентно присоединению ForwardingGroup к данной System в соответствии с семантикой «отношение включения». Удаление элемента из данного перечня логически эквивалентно отсоединению ForwardingGroup от данной System. Примечание — Тип ресурса SystemForwardingGroup представляет собой связь между System и ForwardingGroup (см. 5.13.1.1.8). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meters	*Collection [Meter]*	Ссылка на перечень Ресурсов Meter, которые используются для мониторинга данной System. Примечание — данные Meter предназначены для System, а не для какого-либо отдельного компонента System. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLog	*ref*	Ссылка на EventLog данной System. Примечание — данный EventLog предназначен для System, а не для какого-либо отдельного компонента System. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Реализуя или используя System Поставщики и Потребители должны использовать синтаксис и семантику атрибутов в соответствии с таблицей, приведенной выше, а также таблицами, описывающими встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Следующие псевдосхемы (см. 1.3) описывают сериализация Ресурса и в JSON и в XML.

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,

"systems": { "href": *string* }, ?
"machines": { "href": *string* }, ?
"credentials": { "href": *string* }, ?
"volumes": { "href": *string* }, ?
"networks": { "href": *string* }, ?
"networkPorts": { "href": *string* }, ?
"addresses": { "href": *string* }, ?
"forwardingGroups": { "href": *string* }, ?
"meters": { "href": *string* }, ?
"eventLog": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string },
?
{ "rel": "gost.e-ecolog.ru ", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string },
?
{ "rel": "gost.e-ecolog.ru ", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<System xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<systems href="*xs:anyURI*"/> ?
<machines href="*xs:anyURI*"/> ?
<credentials href="*xs:anyURI*"/> ?
<olumes href="*xs:anyURI*"/> ?
<networks href="*xs:anyURI*"/> ?
<networkPorts href="*xs:anyURI*"/> ?
<addresses href="*xs:anyURI*"/> ?
<forwardingGroups href="*xs:anyURI*"/> ?
<meters href="*xs:anyURI*"/> ?
<eventLog href="*xs:anyURI*"/> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "

href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
*<xs:any>**
</System>

5.13.1.1 Наборы

В настоящем подпункте приведено описание Наборов Ресурсов, принадлежащих System.

5.13.1.1.1 Набор SystemSystem

Типом ресурса для каждого элемента данного Набора является SystemSystem, приведенный в таблице 10:

Таблица 10 — Атрибуты SystemSystem

Наименование	SystemSystem	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
system	*ref*	Ссылка на Ресурс System. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemSystems": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"system": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemSystem>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<system href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</SystemSystem> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.13.1.1.2 Набор SystemMachine

Типом ресурса для каждого элемента данного Набора является SystemMachine, приведенный в таблице 11.

Таблица 11 — Атрибуты SystemMachine

Наименование	SystemMachine	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
machine	*ref*	Ссылка на Ресурс Machine. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemMachines": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?

"machine": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemMachine>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<Machine href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</SystemMachine> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.1.1.3 Набор SystemCredential

Типом Ресурса для каждого элемента данного Набора является SystemCredential, приведенный в таблице 12.

Таблица 12 — Атрибуты SystemCredential

Наименование	SystemCredential	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
credentials	*ref*	Ссылка на РесурсCredential. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,

"systemCredentials": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"credential": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemCredential>
<id> xs:anyURI </id>
<name> xs:string </name> ?
<description> xs:string </description> ?
<created> xs:dateTime </created> ?
<updated> xs:dateTime </updated> ?
<property key="xs:string"> xs:string </property> *
<credential href="xs:anyURI"/>
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
*<xs:any>**
</SystemCredential> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.1.1.4 Набор SystemVolume

Типом ресурса для каждого элемента данного Набора является SystemVolume, приведенный в таблице 13.

Таблица 13 — Атрибуты SystemVolume

Наименование	SystemVolume	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
volume	*ref*	Ссылка на Ресурс Volume. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemVolumes": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"Volume": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<systemVolume>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<volume href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</systemVolume> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.13.1.1.5 Набор SystemNetwork

Типом ресурса для каждого элемента данного Набора является SystemNetwork, приведенный в таблице 14.

Таблица 14 — Атрибуты SystemNetwork

Наименование	SystemNetwork	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
network	*ref*	Ссылка на Ресурс Network. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemNetworks": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"network": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

5.13.1.1.6 Набор SystemNetworkPort

Типом ресурса для каждого элемента данного Набора является SystemNetworkPort, приведенный в таблице15.

Таблица 15 — Атрибуты SystemNetworkPort

Наименование	SystemNetworkPort	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
networkPort	*ref*	Ссылка на Ресурс NetworkPort. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemNetworkPorts": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"networkPort": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemNetworkPort>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<networkPort href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</SystemNetworkPort> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.13.1.1.7 Набор SystemAddress

Типом ресурса для каждого элемента данного Набора является SystemAddress, приведенный в таблице 16.

Таблица 16 — Атрибуты SystemAddress

Наименование	SystemAddress	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
address	*ref*	Ссылка на Ресурс Address. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemAddresses": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"address": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemAddress>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<address href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</SystemAddress> *

<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.1.1.8 Набор SystemForwardingGroup

Типом ресурса для каждого элемента данного Набора является SystemForwardingGroup, приведенный в таблице 17.

Таблица 17 — Атрибуты SystemForwardingGroup

Name	SystemForwardingGroup	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
forwardingGroup	*ref*	Ссылка на Ресурс ForwardingGroup. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count", *number*,
"systemForwardingGroups": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"forwardingGroup": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemForwardingGroup>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<forwardingGroup href="*xs:anyURI*"/>

<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<xs:any>*
</SystemForwardingGroup> *
<operation rel="add" href="xs:anyURI"/> ?
<xs:any>*
</Collection>

5.13.1.1.9 Набор SystemMeter

Типом ресурса для каждого элемента данного Набора является Meter, описание которого приведено в 5.17.3.

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meters": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter ...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter ...
</Meter> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.1.2 Операции

Ресурс System поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс SystemCollection.

Кроме того, определены следующие специальные операции:

start/stop/restart/pause/suspend

/link@rel: gost.e-ecolog.ru

где «xxx» является операцией start, stop, restart, pause или suspend.

Данная операция должна рекурсивно выполнять требуемую операцию для каждого компонента System (Machine или вложенные System).

Примечание — Для этой операции не требуется, чтобы все Machine были в одинаковом состоянии для получения к ним доступа, и воздействие данной операции может быть разным в зависимости от текущего состояния компонента; дополнительная информация о выполнении операций над Ресурсами Machine приведена в 5.14.1.2. Если для некоторого экземпляра Machine операция является невыполнимой, такой экземпляр Machine не следует подвергать данной операции.

export

/link@rel: gost.e-ecolog.ru

Данная операция должна экспортировать экземпляр System. Если по указанному URI уже существует пакет экспорта, то он обновляется в соответствии со значениями System и любых Ресурсов компонентов управления. В противном случае по указанному URI создается новый пакет экспорта с

типом медиа, определенным параметром format. Могут использоваться и другие форматы, если они поддерживаются, но не указаны в настоящем стандарте.

Входные параметры:

1) "format": тип string — необязательный;

2) указывает на тип медиа экспортируемых данных. Если не задан, то значением по умолчанию должно быть "application/ovf";

3) "destination": тип URI — необязательный.

Указывает на местоположение экспортируемых данных. Если данный параметр отсутствует, то в ответе заголовок HTTP Location должен содержать URL экспортированных данных. На основании конкретного протокола, определенного в URI, Потребитель, возможно, должен указать в поле "properties" дополнительную информацию (например, учетные данные). В случае HTTP следует использовать PUT для того, чтобы разместить данные в указанном местоположении.

Выходные параметры: нет.

Протокол HTTP

Чтобы экспортировать System в URI gost.e-ecolog.ru Ресурса System направляется запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "action": "gost.e-ecolog.ru ",
"format": *string, ?*
"destination": *string*, ?
"properties": { *string*: *string*, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<format>*xs:string*</format> ?
<destination> xs:anyURI </destination> ?
<property key="xs:string"> xs:string </property> *
<xs:any>*
</Action>

5.13.2 Ресурс SystemCollection

Ресурс SystemCollection представляет собой Набор Ресурсов System Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count", *number*,
"systems": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты System...
}, +
], ?
"operations": [
{ "rel": "add", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* } ?
]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<System>
<id>*xs:anyURI*</id>
... остальные атрибуты System...
</System> *
<operation rel="add" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.2.1 Операции

Примечание — Операция "add" требует использования SystemTemplate (см. 4.2.1.1).

Ресурсы, созданные во время процесса создания System, должны «принадлежать» созданному Ресурсу System (см. 5.13.1). Например, если componentDescriptor ссылается на шаблон MachineTemplate, который содержит ссылку на VolumeTemplate, то в результате к атрибуту System.machine будет добавлена ссылка на новый экземпляр Machine, а к атрибуту System.volume будет добавлен новый экземпляр Volume. Однако если этот MachineTemplate ссылается на существующий атрибут Volume, то этот Volume не должен добавляться к атрибутам верхнего уровня экземпляра System.

Кроме этого определены следующие специальные операции:

import

/link@rel:gost.e-ecolog.ru

Данная операция должна импортировать System. Помимо создания System также могут быть созданы экземпляры Machine, Volume, Network и, возможно, рекурсивно другие экземпляры System и их компоненты в соответствии с записями дескриптора импорта. Более подробная информация об этом процессе приведена в приложении А.

Входные параметры:

1) "source": тип URI — обязательный.

Указывает местоположение, откуда получают импортированные данные. На основании конкретного протокола, определенного в URI, Потребитель, возможно, должен предоставить дополнительную информацию (например, учетные данные) в поле "properties".

Выходные параметры: нет.

Протокол HTTP

Чтобы импортировать System отправляется запрос POST в URI "gost.e-ecolog.ru .org/cimi/1/action/import" Ресурса SystemCollection, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "action": "gost.e-ecolog.ru ",
"source": *string*, ?
"properties": { *string*: *string*, + } ?
...
}

Типмедиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<source>*xs:anyURI*</source> ?
<property key="*xs:string*">*xs:string*</property> *
*<xs:any>**
</Action>

5.13.3 Ресурс SystemTemplate

Ресурс SystemTemplate содержит совокупность отдельных дескрипторов, необходимых для создания компонентов System. Считают, что каждый дескриптор компонента является сохраненным видом операции create, которая создает экземпляр компонента. На практике Поставщик интерпретирует совокупность дескрипторов компонентов как последовательность операций создания, которые должны выполняться в порядке, удовлетворяющем зависимостям (например, ссылкам между компонентами), выраженными между этими компонентами.

SystemTemplate может содержать в дескрипторах ссылки на компоненты, используемые для выражения связей между компонентами итоговой Системы. Составная ссылка использует атрибут "name" целевого компонента (на который указывает ссылка). Например <Volumehref = "#newVolume"/> может ссылаться на Volume, названный newVolume. Наименование ссылки#newVolume заменяется фактическим URL Ресурса в созданном экземпляре System.

SystemTemplate не должен содержать два дескриптора компонента одинакового типа, которые могут привести к одинаковому непустому значению для атрибута "name" итоговых компонентов. Попытка создать или обновить SystemTemplate, который не удовлетворяет этому правилу, приводит к ошибке. Описание атрибутов SystemTemplate приведено в таблице 18.

Таблица 18 — Атрибуты SystemTemplate

<table>
<tr><td>Наименование</td><td colspan="3">SystemTemplate</td></tr>
<tr><td>Тип URI</td><td colspan="3">gost.e-ecolog.ru</td></tr>
<tr><td>Атрибут</td><td>Тип</td><td colspan="2">Описание</td></tr>
<tr><td>Component Descriptors</td><td>Component Descriptor [ ]</td><td colspan="2">Перечень дескрипторов компонентов, описывающих компоненты экземпляра System, созданного в соответствии с данным SystemTemplate. При создании System для каждого дескриптора компонента создается соответствующий компонент. Каждый дескриптор компонента ссылается на Шаблон (по ссылке или значению) и может также предоставить дополнительные метаданные (наименование, описание, свойства). Порядок создания компонентов в SystemTemplate не определен; в частности, порядок дескрипторов компонентов в данном массиве не задает порядок создания
<table>
<tr><td>Наименование</td><td colspan="2">component Descriptor</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>name</td><td>string</td><td>Значение атрибута "name", связанного с компонентом экземпляра System, созданным из данного дескриптора компонента.
Примечание — Это наименование не следует путать с наименованием, которое может присутствовать в Шаблоне компонента, например, MachineTemplate, от которого создаётся данный компонент.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>descripttion</td><td>string</td><td>Значение атрибута "description", связанного с компонентом экземпляра System, созданным из данного дескриптора компонента.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>properties</td><td>map</td><td>Пары ключ/значение, которые связаны с компонентом экземпляра System, созданным из данного дескриптора компонента.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>
</td></tr>
</table>

Продолжение таблицы 18

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>Component Descriptors</td><td>Component Descriptor []</td><td>
<table>
<tr><td>Наименование</td><td colspan="2"></td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>type</td><td>URI</td><td>TypeURI компонента, который создан из данного дескриптора компонента, например для Machine:
gost.e-ecolog.ru
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>component Template</td><td>any</td><td>Ссылка на Шаблон компонента или непосредственно вложенный Шаблон, (т. е. значение Шаблона).
Следует обратить внимание на то, что точное наименование данного атрибута может варьироваться в зависимости от типа создаваемого Ресурса, создаваемого, например MachineTemplate для Machine.
Данный атрибут должен содержать:
- шаблон, который предоставлен вложением. Такой встроенный Шаблон может содержать ссылки на компоненты, каждая из которых должна разрешаться как URI компонента с тем же наименованием однажды созданного из данного SystemTemplate; либо ссылку на внешний определенный Шаблон. Для некоторых атрибутов внутри элемента componentTemplate могут быть определены пары наименование/значение, чтобы переопределить подобные атрибуты в Шаблоне, на который указывает ссылка (см. 4.2.1.1). В следующем примере показано, как можно добавлять к внешнему Шаблону ссылки на компоненты.
Пример (JSON):
"machineTemplate": {
"href":
"http://example.com/MachineTemplates/72000",
"credential": { "href": "#MyCredential" }
}
Атрибут "credential" предполагает, что есть другой элемент componentDescriptor "MyCredential" типа "Credential" в SystemTemplate. Он должен установить или переопределить подобный атрибут в MachineTemplate, на который указывает ссылка, когда тот будет использоваться для создания компонента Machine.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>quantity</td><td>Integer</td><td>Число экземпляров компонентов, которые будут созданы из данного дескриптора компонентов. По умолчанию это число равно 1. Если значение равно 2 или больше, фактическое наименование, присвоенное каждому экземпляру, получается конкатенацией параметра "name" и порядковым номером (например, если name="myMachine" и quantity=3, то имена будут: myMachine1, myMachine2, myMachine3.)
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись
</td></tr>
</table>

Окончание таблицы 18

Атрибут	Тип	Описание
meter Templates	*Meter Templates[]*	Перечень ссылок на шаблоны MeterTemplate, которые следует использовать для создания и подключения набора новых экземпляров Meter к создаваемой System. Примечание — Могут быть заданы атрибуты MeterTemplate, а не указана ссылка на существующий Ресурс MeterTemplate.
meter Templates	*Meter Templates[]*	**Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
eventLog-Template	*ref*	Ссылка на EventLogTemplate, которая должна использоваться для создания и подключения нового экземпляраEventLog к создаваемой System. Примечание — Могут быть заданы атрибуты EventLogTemplate, а не указана ссылка на существующий Ресурс EventLogTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
import Image	*ref*	Если данный шаблон является результатом импорта, например из пакета OVF, данный атрибут должен использоваться. Если он присутствует, то должен ссылаться на источник импорта (например, пакет OVF) используемый для того, чтобы создать данный Шаблон. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

При реализации или использовании SystemTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов в соответствии с таблицей 18, а также таблицами, описывающими встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Сериализацию Ресурса и в JSON иXML (см. 1.3) осуществляют следующим образом:

JSON тип медиа: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"componentDescriptors": [
{ "name": *string*, ?
"description": *string*, ?
"properties": { *string*: *string*, + }, ?
"type": *string*,
"*componentTemplate*": {
"href": *string, ?*
... атрибуты ComponentTemplate... ?
},
"quantity": *number ?*
}, +
], ?
"meterTemplates": [
{ "href": *string*, ?
... атрибуты MeterTemplate... ?
}, *
], ?

"eventLogTemplate": {
"href": *string*, ?
... *атрибуты EventLogTemplate*... ?
}, ?
"importImage": { "href": string }, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string }?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<SystemTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<componentDescriptor>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<property key="*xs:string*">*xs:string*</property> *
<type>*xs:anyURI*</type>
<*componentTemplate* href="*xs:anyURI*"? >
... *атрибуты ComponentTemplate... ?*
</*componentTemplate*> *
<quantity>*xs:integer*</quantity>
</componentDescriptor> *
<meterTemplate href="*xs:anyURI*"? >
... *атрибуты MeterTemplate*... ?
</meterTemplate> *
<eventLogTemplate href="*xs:anyURI*"? >
... *атрибуты EventLogTemplate*... ?
</eventLogTemplate> ?
<*importImage* href="*xs:anyURI*"? >
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<xs:any>*
</systemTemplate>

5.13.3.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс SystemTemplateCollection.

Кроме того, определены следующие специальные операции:

export

/link@rel: gost.e-ecolog.ru

Данная операция должна экспортировать SystemTemplate. Если в указанном URI уже существует пакет экспорта, то он обновляется в соответствии со значениями SystemTemplate и любых Ресурсов компонентов управления. В противном случае в указанном URI создается новый пакет экспорта с типом медиа, определенным параметром «format». Допускается использовать другие форматы, если они поддерживаются, но не приведены в настоящем стандарте.

Входные параметры:

1) "format": тип string — необязательный;

2) указывает на Тип медиа экспортируемых данных. Если не задан, то значением по умолчанию должно быть "application/ovf";

3) "destination": тип URI — необязательный.

Указывает на местоположение экспортируемых данных. Если данный параметр отсутствует, то в ответе заголовок HTTP Location должен содержать URL экспортированных данных. На основании конкретного протокола, определенного в URI, Потребитель должен указать в поле "properties" дополнительную информацию (например, учетные данные). В случае HTTP следует использовать PUT с целью размещения данных в указанном местоположении.

Выходные параметры: Нет.

Протокол HTTP

Чтобы экспортировать SystemTemplate, в URI gost.e-ecolog.ru Ресурса SystemTemplate направляется запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "action": "gost.e-ecolog.ru ",
"format": *string, ?*
"destination": *string*, ?
"properties": { *string*: *string*, + } ?
…
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<format>*xs:string*</format> ?
<destination> xs:anyURI </destination> ?
<property key="xs:string"> xs:string </property> *
<xs:any>*
</Action>

5.13.4 Ресурс SystemTemplateCollection

Ресурс SystemTemplateCollection представляет собой Набор Ресурсов SystemTemplateCollection Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"systemTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты SystemTemplate...
}, +
], ?
"operations": [
{ "rel": "add", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* } ?
]
…
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<SystemTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты System...
</SystemTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.13.4.1 Операции

Определены следующие специальные операции:

import

/link@rel: gost.e-ecolog.ru

Данная операция должна импортировать SystemTemplate. Помимо создания SystemTemplate также могут быть созданы экземпляры MachineTemplate, VolumeTemplate и NetworkTemplate, и, возможно, рекурсивно другие экземпляры SystemTemplate и их компоненты в соответствии с записями дескриптора импорта. Дополнительная информация об этом процессе приведена в приложенииА.

Входные параметры:

1) "source": тип URI — обязательный

Указывает на местоположение, откуда поступают импортированные данные. На основании конкретного протокола, определенного в URI, Потребитель должен предоставить дополнительную информацию (например, учетные данные) в поле "properties".

Выходные параметры: нет.

Протокол HTTP

Чтобы импортировать SystemTemplate, в URIgost.e-ecolog.ru .org/cimi/1/action/import Ресурса SystemTemplateCollection передают запросPOST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "action": "gost.e-ecolog.ru ",
"source": *string*, ?
"properties": { *string*: *string*, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<source>*xs:anyURI*</source> ?
<property key="*xs:string*">*xs:string*</property> *
*<xs:any>**
</Action>

5.14 Ресурс Machine и отношения

Ресурсы, используемые при конструировании Machine, и отношения между ними приведены на рисунке 3, который представлен в виде схемы отношений Ресурсов, при этом использование UML не является строгим и нормативным.

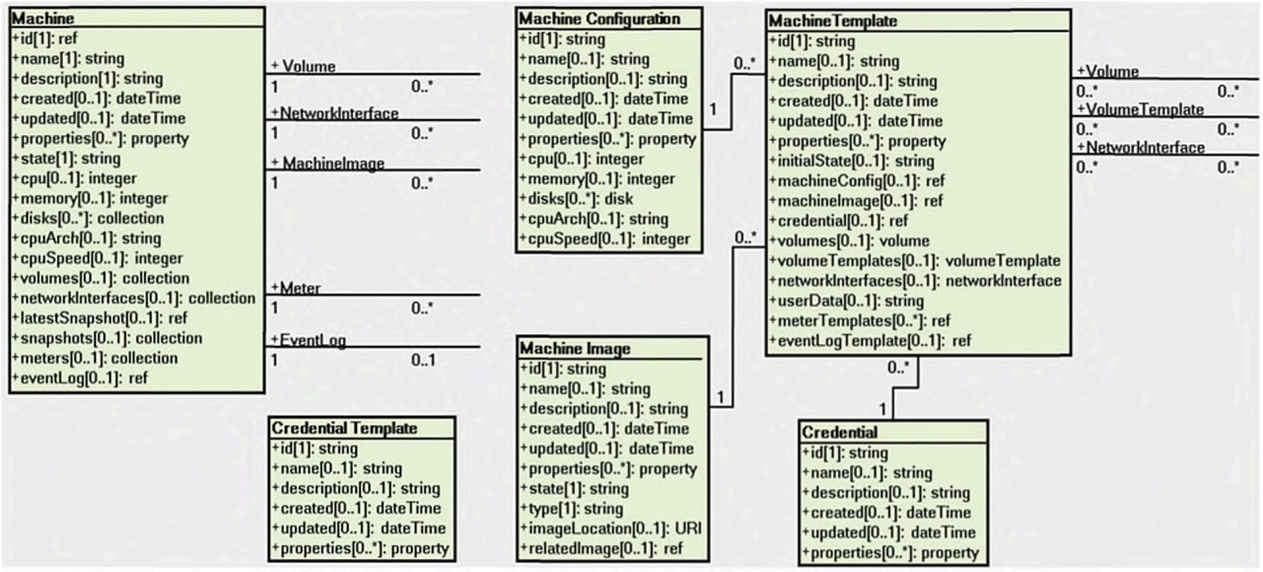

Рисунок 3 — Ресурсы Machine

5.14.1 Machine (Машина)

Machine является обособленным вычислительным Ресурсом, инкапсулирующим CPU, и память. Атрибуты Machine приведены в таблице 19.

Таблица 19 — Атрибуты Machine

Наименование	Machine	
Тип URI	gost.e-ecolog.ru /Machine	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние Machine. Допустимые значения включают в себя: **CREATING**: Machine находится в процессе создания. **STARTING**: Machine находится в процессе запуска. **STARTED**: Machine доступна и готова к использованию. **STOPPING**: Machine находится в процессе остановки. **STOPPED**: Это значение является виртуальный эквивалентом выключения физической Machine. Какого-либо сохраненного состояния ЦП или памяти не существует. В 5.14.2.1 указано первоначальное состояние Machine. **PAUSING**: Machine в процессе приостановки и перехода в состояние PAUSED. **PAUSED**: В этом состоянии Machineи ее виртуальные ресурсы остаются инстанцированными, ресурсы остаются выделенными аналогично состоянию STARTED, но Machine и ее виртуальным ресурсам не разрешено выполнение задач **SUSPENDING**: Machine находится в процессе приостановки и сохранения ресурсов в энергонезависимую память. **SUSPENDED**: В данном состоянии Machine и ее виртуальные ресурсы сохраняются в энергонезависимой памяти. Machine и ее ресурсам не разрешено выполнение задач. **DELETING**: Machine находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в Machine. Операции, которые приводят к переходам к указанным состояниям, приведены в 5.14.1.2. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
cpu	*integer*	Число ЦП, выделенное данному экземпляру Machine. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

Продолжение таблицы 19

Атрибут	Тип	Описание
memory	*integer*	Размер памяти (RAM) в кибибайтах, выделенных данному экземпляру Machine. Если это значение увеличено, это подразумевает, что для Machine выделено больше RAM, и наоборот — если значение уменьшено. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
discs	*Collection [Disk]*	Ссылка на перечень дисков (локальное хранилище), которые являются частью Machine. Добавление элемента к этому перечню создает диск. Примечание — Тип Ресурса Disk определен в 5.14.1.1.1. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
cpuArch	*string*	Архитектура ЦП, которая поддерживается Machine, созданной с помощью данной конфигурации. Допустимые значения включают в себя: **68000**, **Alpha**, **ARM**, **Itanium**, **MIPS**, **PA_RISC**, **POWER**, **PowerPC**, **x86**, **x86_64**, **z/Architecture**, **SPARC**. Поставщики могут установить дополнительные значения. **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения
cpuSpeed	*integer*	Приблизительная скорость ЦП данного экземпляра Machine в мегагерцах. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
volumes	*collection [Machine Volume]*	Ссылка на перечень ссылок на экземпляры Volume, которые подключены к данной Machine. Добавление Volume к этому перечню означает, что у Machine есть некоторый доступ к данным, хранящимся в Volume. Удаление Volume из этого перечня означает, что у Machine больше нет доступа к данным в Volume. Примечание — Тип Ресурса MachineVolume представляет связь между Machine и Volume (см. 5.14.1.1.2). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
networkInterfaces	*collection [Machine Network Interface]*	Ссылка на перечень MachineNetworkInterfa седанного экземпляра Machine. Примечание —Тип Ресурса MachineNetworkInterface представляет связь между Machine и NetworkInterface (см. 5.14.1.1.3). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
latestSnapshot	*ref*	Ссылка на снимок состояния, представляющий последнее состояние, захваченное для данной Machine (последний по времени снимок состояния, либо снимок, из которого было восстановлено состояние). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Окончание таблицы 19

Атрибут	Тип	Описание
snapshots	*collection [Machine Snapshot]*	Ссылка на перечень ссылок на образы снимков состояния данной Machine. Примечание — Тип Ресурса MachineSnapshot представляет связь между Machine и Snapshot (см. 5.14.1.1.5). **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meters	*collection [Meter]*	Ссылка на перечень экземпляров Meter, которые используются для мониторинга данной Machine. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLog	*ref*	Ссылка на EventLog данной Machine. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

При реализации или использовании Machine Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов в соответствии с таблицей 19, а также таблицами, описывающими встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Сериализацию Ресурса и в JSON и в XML (см. 1.3) осуществляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"cpu": *number*,
"memory": *number*,
"disks" : { "href": *string* }, ?
"cpuArch": *string*, ?
"cpuSpeed": *number*, ?
"volumes": { "href": *string* }, ?
"networkInterfaces": { "href": *string* }, ?
"latestSnapshot": { "href": *string* }, ?
"snapshots": { "href": *string* }, ?
"meters": { "href": *string* }, ?
"eventLog": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* },?
{ "rel": "gost.e-ecolog.ru ", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string }
?

{ "rel": "gost.e-ecolog.ru ", "href": string }
?
{ "rel": "gost.e-ecolog.ru ", "href": string }
?
]
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Machine xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<cpu>*xs:integer*</cpu>
<memory>*xs:integer*</memory>
<disks href="*xs:anyURI*"/> ?
<cpuArch>*xs:string*</cpuArch> ?
<cpuSpeed>*xs:integer*</cpuSpeed> ?
<volumes href="*xs:anyURI*"/> ?
<networkInterfaces href="*xs:anyURI*"/> ?
<latestSnapshot href="*xs:anyURI*"/> ?
<snapshots href="*xs:anyURI*"/> ?
<meters href="*xs:anyURI*"/> ?
<eventLog href="*xs:anyURI*"/> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<*xs:any*>*
</Machine>

5.14.1.1 Наборы

В настоящем подпункте приведены Ресурсы Наборов, принадлежащих Machine.

5.14.1.1.1 Набор DiskCollection

Типом ресурса для каждого элемента данного Набора является Disk, приведенный в таблице 20.

Таблица 20 — Атрибуты Disk

Наименование	Disk	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
capacity	*integer*	Начальная емкость диска в килобайтах. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
initialLocation	*string*	Путь к точке первого появления диска в пространстве наименований операционной системы после развертывания. Потребители могут переместить диск. Поддержка данного атрибута указывает, что Поставщик может сообщить эту информацию Потребителю. **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения

Сериализация JSON:

```
{ "resourceURI": "gost.e-ecolog.ru                                    ",
"id": string,
"count": number,
"disks": [
        { "resourceURI": "gost.e-ecolog.ru                                    ",
                "id": string,
                "name": string, ?
                "description": string, ?
                "created": string, ?
                "updated": string, ?
                "properties": { string: string, + }, ?
                "capacity": number,
                "initialLocation": string, ?
                "operations": [
                        { "rel": "edit", "href": string }, ?
                        { "rel": "delete", "href": string } ?
        ] ?
    ...
        }, +
        ], ?
        "operations": [ { "rel": "add", "href": string } ? ]
        ...
}
```

Сериализация XML:

```
<Collection resourceURI="gost.e-ecolog.ru                                    "
xmlns="gost.e-ecolog.ru                                    ">
<id>xs:anyURI</id>
<count>xs:integer</count>
<Disk>
<id>xs:anyURI</id>
<name>xs:string</name> ?
<description>xs:string</description> ?
<created>xs:dateTime</created> ?
<updated>xs:dateTime</updated> ?
<property key="xs:string">xs:string</property> *
<capacity>xs:integer</capacity>
<initialLocation>xs:string</initialLocation> ?
```

<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</Disk> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.1.1.2 Набор MachineVolumeCollection

Типом Ресурса для каждого элемента данного Набора является MachineVolume, приведенный в таблице 21.

Таблица 21 — Атрибуты MachineVolume

Наименование	MachineVolume	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
initialLocation	*string*	Путь к точке первого появления тома в пространстве наименований операционной системы Примечание — После развертывания Потребители могут переместить том. Поддержка данного атрибута указывает, что Поставщик может сообщить эту информацию Потребителю **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения
volume	*ref*	Ссылка на связанный Ресурс Volume. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализация JSON:

```
{ "resourceURI": "gost.e-ecolog.ru                                    ",
   "id": string,
   "count": number,
   "machineVolumes": [
            { "resourceURI": "gost.e-ecolog.ru                                    ",
                     "id": string,
                     "name": string, ?
                     "description": string, ?
                     "created": string, ?
                     "updated": string, ?
                     "properties": { string: string, + }, ?
                     "initialLocation": string, ?
                     "volume": { "href": string },
                     "operations": [
                              { "rel": "edit", "href": string }, ?
                              { "rel": "delete", "href": string } ?
                     ] ?
                     ...
            }, +
   ], ?
   "operations": [ { "rel": "add", "href": string } ? ]
   ...
}
```

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MachineVolume>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description> xs:string </description> ?
<created> xs:dateTime </created> ?
<updated> xs:dateTime </updated> ?
<property key="xs:string"> xs:string </property> *
<initialLocation> xs:string </initialLocation> ?
<volume href="xs:anyURI"/>
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
*<xs:any>**
</MachineVolume> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.1.1.3 Набор MachineNetworkInterfaceCollection

Типом Ресурса для каждого элемента данного Набора является MachineNetworkInterface, приведенный в таблице 22.

Таблица 22 — Атрибуты MachineNetworkInterface

Наименование	MachineNetworkInterface	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
addresses	*collection [Machine Network Interface Address]*	Ссылка на перечень ссылок на экземпляры Addres для данного сетевого интерфейса. Примечание — Тип ресурса MachineNetworkInterfaceAddres представляет связь между MachineNetworkInterface и Address (см. 5.14.1.1.4). **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
network	*ref*	Ссылка на Network для данного сетевого интерфейса. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
networkPort	*ref*	Ссылка на NetworkPort для данного сетевого интерфейса. Если будет предоставлен данный атрибут, то атрибут «network» в NetworkPort, на который указывает ссылка, должен иметь то же самое значение, что и атрибут "networks" в данном сетевом интерфейсе. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
state	*string*	Состояние MachineNetworkInterface. Допустимые значения включают в себя: **ACTIVE**: активный интерфейс — основной интерфейс, который в состоянии передавать трафик. **PASSIVE**: пассивный интерфейс находится в режиме ожидания, готовый к передаче трафика, если возникает ошибка в основном интерфейсе. **DISABLED:** отключенный интерфейс — такой интерфейс не может передавать трафик.

Окончание таблицы 22

Атрибут	Тип	Описание
		Ограничения: **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
macAddress	*string*	Адрес, присвоенный гипервизором при создании машины, либо уникальный адрес, присвоенный вручную. Несмотря на то, что данный атрибут может быть задан Пользователем, в большинстве случаев он предоставляется Поставщиком. Значение задается обычно только в том случае, если Шаблон используется только для одной конкретной Машины. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
mtu	*integer*	Определяет максимальный размер передаваемого пакета (MTU — MaximumTransmissionUnit) **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number,*
"machineNetworkInterfaces": [
{
"resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"addresses": { "href": *string* },
"network": { "href": *string* },
"networkPort": { "href": *string* }, ?
"state": *string*, ?
"macAddress": *string*, ?
"mtu": *number*, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>

<MachineNetworkInterface>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<addresses href="*xs:anyURI*"/>
<network href="*xs:anyURI*"/>
<networkPort href="*xs:anyURI*"/> ?
<state>*xs:string*</state> ?
<macAddress>*xs:string*</macAddress> ?
<mtu>*xs:integer*</mtu> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineNetworkInterface> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.1.1.4 Ресурс MachineNetworkInterfaceAddressCollection

Типом Ресурса для каждого элемента этого Набора является MachineNetworkInterfaceAddress, приведенный в таблице 23.

Таблица 23 — Атрибуты MachineNetworkInterfaceAddress

Наименование	MachineNetworkInterfaceAddress	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
address	*ref*	Ссылка на Ресурс Address. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"machineNetworkInterfaceAddresses": [
{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"address": { "href": *string* },
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...

}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru
Collection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MachineNetworkInterfaceAddress>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<address href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineNetworkInterfaceAddress> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.1.1.5 Ресурс MachineSnapshotCollection

Типом ресурса для каждого элемента этого Набора является **MachineSnapshot**, приведенный в таблице 24.

Таблица 24 — Атрибуты MachineSnapshot

Наименование	MachineSnapshot	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
spapshot	*ref*	Ссылка на снимок состояния, Ресурс MachineImage. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
"count": number,
"machineSnapshots": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"snapshot": { "href": string },
"operations": [

{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
] ?
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id> xs:anyURI </id>
<count> xs:integer </count>
<MachineSnapshot>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<snapshot href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineSnapshot> *
*<xs:any>**
</Collection>

Примечание — Предыдущие версии данной спецификации включали в себя операцию «add» для данного Ресурса. В настоящее время вместо этого рекомендуется создание нового экземпляра MachineImage с атрибутом imageLocation, указывающим на Machine, чей снимок состояния хранится в созданном экземпляре.

5.14.1.1.6 Ресурс MachineMeterCollection

Типом ресурса для каждого элемента этого Набора является Meter, приведенный в 5.17.3.

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meters": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter ...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter ...

</Meter> *
<operation rel="add" href="xs:anyURI"/> ?
<xs:any>*
</Collection>

5.14.1.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MachineCollection.

Кроме этого определены следующие специальные операции:

start

/link@rel: gost.e-ecolog.ru

Данная операция должна запустить Machine.

Входные параметры: Нет.

Выходные параметры: Нет.

Во время обработки этой операции Machine должна быть в состоянии STARTING.

После успешного завершения этой операции Machine должна быть в состоянии STARTED.

Если Machine будет в состоянии STOPPED, то ее запуск должен быть виртуальным эквивалентом включения на физической машине. Какого-либо восстановленного состояния ЦП или Памяти не существует, поэтому гость ОС обычно выполняет задачи установки или начальной загрузки.

Если Machine была в состоянии SUSPENDED или PAUSED, ее запуск должен быть равносилен ее возобновлению.

Протокол HTTP

Для запуска Machine в URI "gost.e-ecolog.ru .org/cimi/1/action/start" Ресурса Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="xs:string">*xs:string*</property> *
<*xs:any*>*
</Action>

После успешной обработки запроса ответ HTTP может быть пустым.

stop

/link@rel: gost.e-ecolog.ru

Данная операция должна остановить экземпляр Machine

Входные параметры:

1) "force": тип булевая переменная — необязательный.

Флажок, указывающий, что Поставщик должен моделировать отключение питания (force=true), либо должен моделировать операцию завершения работы, которая позволит приложениям и файловой системе корректно сохранить состояние (force=false). Включение данного параметра Потребителями является необязательным условием и если он не указан, Поставщик может выбрать любой механизм. Поставщикам предлагается информировать об этой опции через MachineStopForceDefault.

Выходные параметры: Нет.

Во время обработки данной операции Machine должна быть в состоянии STOPPING.

После успешного завершения данной операции Machine должна быть в состоянии STOPPED. Остановка Machine с force=true должна быть виртуальным эквивалентом выключения компьютера. Какого-либо сохраненного состояния ЦП или памяти не существует. Остановка Machine с force=false должна привести к машине с корректными файловыми системами.

Потребитель может повторно вызвать операцию stop, если текущее состояние STOPPING возможно с force=true, но Поставщики не должны вызывать операцию stop с force=true самостоятельно.

Протокол HTTP

Для остановки Machine в URI "gost.e-ecolog.ru :.org/cimi/1/action/stop" экземпляра Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"force": *boolean*, ?
"properties": { *string*: *string*, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<force>*xs:boolean*</force> ?
<property key="*xs:string*">*xs:string*</property> *
*<xs:any>**
</Action>

После успешной обработки запроса ответ HTTP может быть пустым.

restart

/link@rel: gost.e-ecolog.ru

Данная операция должна перезапустить Machine. Если Machine будет в состоянии STARTED, то эта операция должна иметь эффект выполнения операций "stop", а затем "start". Если Machine будет в состоянии STOPPED, то результат данной операции должен быть равнозначен результату выполнения операции "start".

Входные параметры:

1) "force": тип булевая переменная — необязательный.

Флажок, указывающий, что Поставщик должен моделировать отключение питания (force=true), либо должен моделировать операцию завершения работы, которая позволит приложениям и файловой системе корректно сохранить свое состояние (force=false). Включение данного параметра Потребителями является необязательным условием и, если он не указан, Поставщик может выбрать любой механизм. Поставщикам предлагается информировать об этой опции через возможность MachineStopForceDefault.

Выходные параметры: Нет.

Во время обработки данной операции Machine должна быть в состоянии STOPPING и/или STARTING, в зависимости от своего начального состояния.

После успешного завершения данной операции Machine должна быть в состоянии STARTED. Перезапуск Machine должен быть виртуальным эквивалентом выключения, а затем включения компьютера.

При невосстановлении ЦП или Памяти гостевая Операционная система обычно выполняет задачи загрузки или установки.

Протокол HTTP

Для перезапуска Machine в URI "gost.e-ecolog.ru " Ресурса Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"force": *boolean*, ?
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<force>*xs:boolean*</force> ?

<property key="*xs:string*">*xs:string*</property> *

<*xs:any*>*

</Action>

После успешной обработки запроса тело ответа HTTP может быть пустым.

pause

/link@rel: gost.e-ecolog.ru

Данная операция должна приостановить Machine.

Входные параметры: Нет.

Выходные параметры: Нет.

Во время обработки данной операции Machine должна быть в состоянии PAUSING.

После успешного завершения данной операции Machine должна быть в состоянии PAUSED. Приостановка Machine должна сохранить Machine и ее ресурсы, но Machine не должна быть доступной для выполнения любых задач. Текущее состояние ЦП и памяти должно быть сохранено в энергозависимой памяти.

Протокол HTTP

Для приостановки Machine, в URI "gost.e-ecolog.ru .org/cimi/1/action.pause" Ресурса Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",

"action": "gost.e-ecolog.ru ",

"properties": { *string*: *string*, + } ?

...

}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<property key="*xs:string*">*xs:string*</property> *

<*xs:any*>*

</Action>

После успешной обработки запроса тело ответа HTTP может быть пустым.

suspend

/link@rel: gost.e-ecolog.ru

Данная операция должна приостановить Machine.

Входные параметры: Нет.

Выходные параметры: Нет.

Во время обработки данной операции Machine должна быть в состоянии SUSPENDING.

После успешного завершения данной операции Machine должна быть в состоянии SUSPENDED. Приостановка Machine должна сохранить Machine и ее ресурсы инстанцированными, но Machine должна быть недоступной для выполнения любых задач. Текущее состояние ЦП и Памяти должно быть сохранено в энергонезависимой памяти.

Протокол HTTP

Для приостановки Machine, в URI "gost.e-ecolog.ru .org/cimi/1/action/suspend" Ресурса Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",

"action": "gost.e-ecolog.ru ",

"properties": { *string*: *string*, + } ?

...

}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<property key="*xs:string*">*xs:string*</property> *

<*xs:any*>*

</Action>

После успешной обработки запроса тело ответа HTTP может быть пустым.

capture

/link@rel: gost.e-ecolog.ru

Данная операция должна создавать новый Ресурс MachineImage из существующей Machine. Данная операция определена в рамках Ресурса MachineImage; дополнительная информация приведена в 5.14.7.1.

Примечание — Несмотря на то, что данная операция выполняется над экземпляром MachineImage, она включена в сериализацию Machine с целью информирования о поддержке данной операции.

Создание снимка состояния Machine

/link@rel: gost.e-ecolog.ru

Данная операция должна создать новый снимок состояния типа MachineImage из существующей Machine. Данная операция определена в рамках Ресурса MachineImage; дополнительная информация приведена в 5.14.7.1.

Примечание — Несмотря на то, что данная операция выполняется над экземпляром MachineImage, она включена в сериализацию Machine с целью информирования о поддержке данной операции.

Восстановление состояния Machine

/link@rel: gost.e-ecolog.ru

Данная операция должна восстановить Machine из ранее созданного MachineImage.

Входные параметры:

1) "image": тип URI — обязательный.

Ссылка на MachineImage.

Выходные параметры: Нет.

Во время обработки данной операции Machine должна быть в состоянии RESTORING.

После успешного завершения данной операции состояние Machine должно быть аналогичным состоянию, определенному в MachineImage, если оно определено. Дополнительная информация приведена в 5.14.2.1.

Примечание — Поставщики могут обозначить наличие поддержки восстановления из MachineImage, не являющимся снимком состояния, в возможности «RestoreFromImage» Ресурса Machine. Если возможность RestoreFromImage не поддерживается, а операция restore поддерживается, то операция restore может восстановить только из MachineImage, являющегося снимком состояния.

Протокол HTTP

Для восстановленияMachine в URI "gost.e-ecolog.ru " Ресурса Machine должен быть отправлен запрос POST, где запрос HTTP должен быть представлен следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",

"action": "gost.e-ecolog.ru ",

"image": *string*,

"properties": { *string*: *string*, + } ?

...

}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<image href="*xs:anyURI*"/>
<property key="*xs:string*">*xs:string*</property> *
*<xs:any>**
</Action>

В этом URI "image" является ссылкой на MachineImage, который будет использоваться.

После успешной обработки запроса ответ HTTP может быть пустым.

5.14.2 MachineCollection

Ресурс MachineCollection представляет Набор Ресурсов Machine Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"machines": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Machine...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Machine>
<id>*xs:anyURI*</id>
... остальные атрибуты Machine...
</Machine> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.2.1 Операции

Примечание — Операция «add» требует, чтобы использовался MachineTemplate (см. 4.2.1.1).

В Ресурсах NetworkInterface, принадлежащих MachineTemplate, может быть ссылка на Ресурс Address. Если данный ресурс не предоставлен при создании, Поставщик должен создать его от имени Потребителя. В таких случаях, если не приняты определенные меры, чтобы изменить это состояние, Addres связывается с создаваемой Machine и должен быть удален Поставщиком в процессе удаления Machine. Кроме того, если созданные Поставщиками Ресурсы Addres будут отсоединены от Machine, Поставщик должен их удалить. Если Ресурс Addres предоставляет Потребитель, то данный Ресурс не должен быть удален при удалении Machine, и Потребитель сам решает, удалять ли этот Ресурс с помощью какой-то другой операции.

После успешной обработки операции "add", если иное не определено в атрибуте «initialState» ресурса MachineTemplate, состояние новой Machine должно быть значением возможности DefaultInitialState, если данная возможность определена. Если ни одна возможность DefaultInitialState не определена, то значением по умолчанию должно быть STOPPED.

Семантика "initialState" должна быть эквивалентна выполнению Поставщиком надлежащих действий над новой Machine для перевода ее в заданное состояние.

Примечание — Указанная семантика управляет действиями гипервизора; также на состояние ресурсов, принадлежащих Machine (например, операционной системы), влияют данные, содержащиеся в MachineImage, использованном для создания новой Machine. Например, если бы initialState новой Machine было STARTED и для создания новой Machine использовался MachineImage со снимком состояния, то Machine не была бы «загружена», а продолжала бы выполнение операции из сохраненного состояния в MachineImage.

Если Поставщик не способен установить начальное состояние Machine (указанное в шаблоне MachineTemplate либо подразумеваемое правилами), то создание Machine должно завершиться с ошибкой.

Если Поставщик неспособен создать новую Machine вследствие недействительных или некорректных учетных данных в MachineTemplate, процесс создания Machine должен потерпеть неудачу. Если какие-либо учетные данные будут включены в MachineTemplate, то они должны быть частью новой Machine, независимо от типа используемого MachineImage.

5.14.3 MachineTemplate

Ресурс MachineTemplate представляет совокупность метаданных и инструкций, используемых при создании Machine. Атрибуты MachineTemplate приведены в таблице 25.

Таблица 25 — Атрибуты MachineTemplate

<table>
<tr><td>Наименование</td><td colspan="2">MachineTemplate</td></tr>
<tr><td>Тип URI</td><td colspan="2">gost.e-ecolog.ru</td></tr>
<tr><td>Атрибут</td><td>Тип</td><td>Описание</td></tr>
<tr><td>initialState</td><td>string</td><td>Начальное состояние новой Machine.
Возможные значения включают в себя такие состояния, определенные для атрибута "state" Machine, которые не являются промежуточными (например, STARTED, STOPPED) и определяются действиями, которые поддерживаются Поставщиком. Поставщики должны распространить перечень доступных значений через возможность "initialStates" Ресурса Machine.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>Machine Config</td><td>ref</td><td>Ссылка на Ресурс MachineConfiguration, который используется для создания Machine из данного MachineTemplate.
Примечание — Могут определяться атрибуты MachineConfiguration, а не ссылка на существующий Ресурс MachineConfiguration.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>machineImage</td><td>ref</td><td>Ссылка на MachineImage, который используется для создания Machine из данного MachineTemplate.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>credential</td><td>ref</td><td>Ссылка на Credential, которые используются для создания начальных учетных данных входа в систему на новой Machine.
Примечание — Могут определяться атрибуты Credential, а не ссылка на существующий Ресурс Credential.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>volumes</td><td>Volume []</td><td>Перечень структур, каждая из которых содержит ссылку на существующий Volume и потенциально описывает аспекты способа подключения данного Volume к Machine во время ее создания от данного шаблона MachineTemplate. У каждой структуры Volume есть следующие атрибуты:
<table>
<tr><td>Наименование</td><td colspan="2">Volume</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>initialLocation</td><td>string</td><td>Путь к точке первого появления тома в пространстве наименований операционной системы.
Поддержка данного атрибута указывает, что Поставщик разрешает Потребителю выбирать местонахождение тома.</td></tr>
</table>
</td></tr>
</table>

Продолжение таблицы 25

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>volumes</td><td>Volume []</td><td>
<table>
<tr><td>Наименование</td><td colspan="2">Volume</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>initialLocation</td><td>string</td><td><u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>volume</td><td>ref</td><td>Ссылка на подключаемый Volume.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения</u>:
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись
</td></tr>
<tr><td>Volume Templates</td><td>Volume Template []</td><td>Перечень структур, каждая из которых содержит ссылку на VolumeTemplate, из которого создается Volume и подключается к Machine, создаваемой из данного шаблона MachineTemplate. Каждая структура может потенциально также включать в себя аспекты, определяющие способ подключения созданного Volume к новой Machine. Если Machine создана в рамках создания System, то Ресурсы Volume, созданные из этих Шаблонов считаются частью этой System, при этом не требуется, чтобы данные шаблоны VolumeTemplate перечислялись в атрибуте volumeTemplate соответствующего SystemTemplate. Если одна и та же ссылка VolumeTemplate указана как в атрибуте VolumeTemplate шаблона SystemTemplate, так и в атрибуте volumeTemplate шаблона MachineTemplate, который содержится в этом шаблоне SystemTemplate, это означает, что создаются несколько различных экземпляров Volume как часть полного создания System. У каждой структуры volumeTemplate есть следующие атрибуты
<table>
<tr><td>Наименование</td><td colspan="2">VolumeTemplate</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>initialLocation</td><td>string</td><td>Путь к точке первого появления тома в пространстве имен операционной системы.
Поддержка данного атрибута указывает, что Поставщик разрешает Потребителю выбирать местонахождение тома.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>volume-Template</td><td>ref</td><td>Ссылка на VolumeTemplate, который используется для создания нового ресурса Volume.
Примечание — Могут определяться атрибуты VolumeTemplate, а не ссылка на существующий Ресурс VolumeTemplate.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>network-Interfaces</td><td>network-Interface []</td><td>Перечень структур, каждая из которых содержит ссылки на Ресурсы и атрибуты, определяющие сетевой интерфейс, который должен быть создан для Machine, создаваемой из данного шаблона MachineTemplate. Ресурсами, на которые ссылается каждая структура networkInterface, является Network, NetworkPort и перечень Addresses</td></tr>
</table>

Продолжение таблицы 25

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>network-Interfaces</td><td>network-Interface []</td><td>
<table>
<tr><td>Наименование</td><td colspan="2">networkInterface</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>addresses</td><td>ref []</td><td>Перечень ссылок на Addres для данного сетевого интерфейса.
Название элемента массива: address.
<u>Ограничения</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>network</td><td>ref</td><td>Ссылка на ресурс Network для этого сетевого интерфейса. Предполагается, что ресурсы NetworkPort и Network определены отдельно, и раньше Ресурсов Machine, которые подключаются к ним.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>NetworkPort</td><td>ref</td><td>Ссылка на NetworkPort для этого сетевого интерфейса.
Примечание — Это ссылка на NetworkPort а не на NetworkPortTemplate. Предполагается, что Ресурсы NetworkPort и Network определены отдельно и раньше Ресурсов Machine, которые подключаются к ним.
Если будет предоставлен данный атрибут, то атрибут "network" в NetworkPort, на который указывает ссылка, должен иметь то же самое значение, что и атрибут "networks" в данном сетевом интерфейсе.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>state</td><td>string</td><td>Состояние сетевого интерфейса. Допустимые значения включают в себя:
ACTIVE: активным интерфейсом является основной интерфейс, который обеспечивает передачу трафика.
PASSIVE: пассивный интерфейс находится в дежурном режиме, готовый к передаче трафика при возникновении ошибки в основном интерфейсе.
DISABLED: отключенным интерфейсом является интерфейс не обеспечивающий трафика.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>mtu</td><td>integer</td><td>Определяет максимальный размер передаваемого пакета (MTU — Maximum Transmission Unit)
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>userData</td><td>string</td><td>Строка, закодированная в Base64. Строка, полученная декодированием userData, должна быть внедрена в Ресурсы Machine, созданные из данного Шаблона. Описание внедрения данных, определенных пользователем, приведено далее.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>

Окончание таблицы 25

Атрибут	Тип	Описание
meter-Templates	*Meter Templates []*	Перечень ссылок на шаблоны MeterTemplate, используемых для создания и подключения новых экземпляров Meter к создаваемой Machine. Примечание — Могут определяться атрибуты MeterTemplate, а не ссылка на существующий Ресурс MeterTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
eventLog-Template	*ref*	Ссылка на EventLogTemplate, используемых для создания и подключения нового экземпляра EventLog к создаваемой Machine. Примечание — Могут определяться атрибуты EventLogTemplate, а не ссылка на существующий Ресурс EventLogTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

При реализации или использовании MachineTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов в соответствии с таблицей 25, а также таблицами, описывающими встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии с приведенным далее описанием. Сериализацию Ресурса и в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"initialState": *string*, ?
"machineConfig": {
"href": *string* |... *атрибуты MachineConfiguration...*
}, ?
"machineImage": {
"href": *string* |... *атрибуты MachineImage...*
}, ?
"credential": {
"href": *string* |... *атрибуты CredentialTemplate...*
}, ?
"volumes": [
{ "initialLocation": *string*?, "href": *string* }, +
], ?
"volumeTemplates": [
{ "initialLocation": string?,
"href": string, ?
... *атрибуты VolumeTemplate*... ?
}, +
], ?
"networkInterfaces": [
{ "addresses": [
{"href": string}, +
],
"network": {"href": string},

"networkPort": {"href": string}, ?
"state": string,
"mtu": number ?
}, +
], ?
"userData": string, ?
"meterTemplates": [
{ "href": string, ?
... *амрибуты MeterTemplate*... ?
}, *
], ?
"eventLogTemplate": {
"href": string, ?
... *амрибуты EventLogTemplate*... ?
}, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<MachineTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<initialState>*xs:string*</initialState> ?
<machineConfig href="*xs:anyURI*"?>
... *амрибуты MachineConfiguration*... ?
</machineConfig> ?
<machineImage href="*xs:anyURI*"?>
... *амрибуты MachineImage*... ?
</machineImage> ?
<credential href="*xs:anyURI*"?>
... *амрибуты CredentialTemplate*... ?
</credential> ?
<volume initialLocation="*xs:string*"? href="*xs:anyURI*" /> *
<volumeTemplate initialLocation="*xs:string*"? href="*xs:anyURI*"? >
... *амрибуты VolumeTemplate... ?*
</volumeTemplate> *
<networkInterface>
<address href="*xs:anyURI*"/> *
<network href="*xs:anyURI*"/>
<networkPort href="*xs:anyURI*"/> ?
<state>*xs:string*</state>
<mtu>*xs:integer*</mtu> ?
</networkInterface> *
<userData>*xs:string*</userData> ?
<meterTemplate href="*xs:anyURI*"? >
... *амрибуты MeterTemplate*... ?
</meterTemplate> *
<eventLogTemplate href="*xs:anyURI*"? >

... *атрибуты EventLogTemplate...* ?
</eventLogTemplate> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineTemplate>

Внедрение данных, определенных пользователем

Для упрощения настройки отдельных Ресурсов Machine допускается передавать произвольные данные в новую Machine с помощью параметра userData. Значение этого параметра должно быть полезной нагрузкой, закодированной с помощью Base64. Поставщик должен организовать доступ к этим данным из Machine одним из следующих методов:

1) Сервер метаданных: данные могут быть получены путем запроса GETHTTP к gost.e-ecolog.ru .

2) Диск: Machine имеет доступ к диску с файловой системой ISO 9660 на нем. Данные могут быть найдены в файле в *<путь к диску>*/cimi/user-data.

3) Изменение образа виртуальной машины: Поставщик меняет корневую файловую систему в образе ресурса Machine до запуска этого экземпляра Machine. В операционных системах, подобных UNIX, данные приведены в файле /var/lib/cimi/user-data.

Поставщику следует реализовать сервер метаданных либо внедрением с помощью диска, поскольку изменение образа потенциально опасно и может не работать для некоторых операционных систем. Поставщик должен указать, какой из этих трех методов поддерживается в возможности «UserData» в ResourceMetadata для Machine. Значением для этой характеристики должно быть одним из «metadata», «disk» или «imgmod» в соответствии с тремя вышеуказанными методами.

Поставщик должен сохранять эти данные при перезапусках Machine. Эти данные представляют собой результат декодирования из Base64 данных, которые были переданы в запрос MachineCreate.

5.14.3.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MachineTemplateCollection.

5.14.4 Ресурс MachineTemplateCollection

Ресурс MachineTemplateCollection представляет собой Набор Ресурсов MachineTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"machineTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты MachineTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MachineTemplate>
<id> xs:anyURI </id>
... остальные атрибуты MachineTemplate...
</MachineTemplate> *

<operation rel="add" href="xs:anyURI"/> ?
<xs:any>*
</Collection>

5.14.4.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов MachineTemplate поддерживается путем отправки запроса POST на адрес URI операции «add» в соответствии с 4.2.1.1.

5.14.5 Ресурс MachineConfiguration

Ресурс MachineConfiguration содержит совокупность конфигурационных параметров, которые определяют (виртуальные) аппаратные ресурсы создаваемого Экземпляра Machine. Ресурсы MachineConfiguration создаются Поставщиками и, по их решению, могут создаваться Потребителями. Атрибуты MachineConfiguration приведены в таблице 26.

Таблица 26 — Атрибуты MachineConfiguration

<table>
<tr><td>Наименование</td><td colspan="2">MachineConfiguration</td></tr>
<tr><td>Тип URI</td><td colspan="2">gost.e-ecolog.ru</td></tr>
<tr><td>Атрибут</td><td>Тип</td><td>Описание</td></tr>
<tr><td>cpu</td><td>integer</td><td>Число CPU, которое имеет Machine, созданная с использованием данной конфигурации.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
<tr><td>memory</td><td>integer</td><td>Размер памяти (RAM) в кибибайтах, выделенных для Machine, созданной с использованием данной конфигурации.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>discs</td><td>Disc []</td><td>Перечень структур, каждая из которых содержит атрибуты, определяющие диски, которые должны быть созданы для Machine, созданной с помощью данного Ресурса MachineConfiguration. Диски представляют собой локальное хранилище для Machine.
У каждого атрибута disc есть следующие вложенные атрибуты:
<table>
<tr><td>Наименование</td><td colspan="2">Диск</td></tr>
<tr><td>Данные</td><td>Тип</td><td>Описание</td></tr>
<tr><td>capacity</td><td>integer</td><td>Начальная емкость диска, описанного данным атрибутом, в килобайтах.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>format</td><td>string</td><td>Формат/тип этого диска (например, ext4, NTFS).
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>initial-Location</td><td>string</td><td>Путь к точке первого появления диска в пространстве наименований операционной системы. После создания экземпляра Machine, Потребители могут переместить диск.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый.
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: необязательная поддержка; изменяемый
Потребитель: необязательная поддержка; чтение-запись</td></tr>
</table>

Окончание таблицы 26

Атрибут	Тип	Описание
cpuArch	string	Архитектура ЦП, которая поддерживается Machine, созданной с использованием данной конфигурации. Допустимыезначениявключают в себя: **68000**, **Alpha**, **ARM**, **Itanium**, **MIPS**, **PA_RISC**, **POWER**, **PowerPC**, **x86**, **x86_64**, **z/Architecture**, **SPARC**. Поставщики могут установить дополнительные значения. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
cpuSpeed	*integer*	Приблизительная скорость ЦП на данной Machine в мегагерцах. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

Примечание — Атрибуты дисков «format» не сохраняются в Ресурсах Machine, так как после создания Machine пользователь Machine получает возможность изменить данный атрибут диска даже без ведома Поставщика. Поэтому данные атрибуты не могут быть аспектом Machine, которым может достоверно управлять Поставщик.

Тип медиа JSON: application/json

Сериализация JSON:

```
{ "resourceURI": "gost.e-ecolog.ru                                    ",
   "id": string,
   "name": string, ?
   "description": string, ?
   "created": string, ?
   "updated": string, ?
   "properties": { string: string, + }, ?
   "cpu": number,
   "memory": number,
   "disks" : [
            { "capacity": number,
            "format": string,
            "initialLocation": string?
            }, +
   ], ?
   "cpuArch": string, ?
   "cpuSpeed": number, ?
   "operations": [
            { "rel": "edit", "href": string }, ?
            { "rel": "delete", "href": string } ?
   ] ?
   ...
}
```

Тип медиа XML: application/xml

Сериализация XML:

```
<MachineConfiguration xmlns="gost.e-ecolog.ru                    ">
<id>xs:anyURI</id>
<name>xs:string</name> ?
<description>xs:string</description> ?
<created>xs:dateTime</created> ?
<updated>xs:dateTime</updated> ?
<property key="xs:string">xs:string</property> *
<cpu>xs:integer</cpu>
<memory>xs:integer</memory>
<disk>
<capacity>xs:integer</capacity>
```

<format>*xs:string*</format>
<initialLocation>*xs:string*</initialLocation> ?
</disk> *
<cpuArch>*xs:string*</cpuArch> ?
<cpuSpeed>*xs:integer*</cpuSpeed> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineConfiguration>

5.14.5.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MachineConfigurationCollection.

5.14.6 РесурсMachineConfigurationCollection

Ресурс MachineConfigurationCollection представляет собой Набор Ресурсов MachineConfiguration Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"machineConfigurations": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты MachineConfiguration...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
…
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /MachineConfigurationCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MachineConfiguration>
<id>*xs:anyURI*</id>
... остальные атрибуты MachineConfiguration...
</MachineConfiguration> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.6.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов MachineConfiguration поддерживается путем отправки запроса POST на адрес URI операции "add"в соответствии с 4.2.1.1.

5.14.7 Ресурс MachineImage

Данный Ресурс представляет информацию о виртуализированных Ресурсах аппаратных средств, необходимую для создания экземпляра Machine; он содержит такие параметры конфигурации, как инструкции запуска, включая возможные комбинации следующих элементов, в зависимости от атрибута «type» созданного MachineImage:

- образ программного обеспечения (т. е. копия установленной Machine), которое необходимо установить на диске и других виртуальных ресурсах. Образ может быть снимком, состоящим из образов дисков, оперативной памяти и другой информации о состоянии ресурса;

- инсталляторы программного обеспечения, которое при выполнении на аппаратных (виртуальных) ресурсах создает реализацию Machine;

- образ диска и набора программного обеспечения и параметров для установки новых компонентов, не включенных в образ исходного диска. Атрибуты MachineImage приведены в таблице 27.

Таблица 27 — Атрибуты MachineImage

Наименование	MachineImage	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние MachineImage. Допустимые значения включают в себя: **CREATING**: MachineImage находится в процессе создания. **AVAILABLE**: MachineImage доступен и готов к использованию. **Если не определено иное, после успешного создания MachineImage должен быть в данном состоянии.** **DELETING**: MachineImage находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в MachineImage. Операции, которые приводят к переходам к указанным состояниям, приведены в 5.14.7.1. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
type	*string*	Тип MachineImage, который представлен данным Ресурсом. В настоящем стандарте установлены следующие значения: **IMAGE**: Этот тип представляет сохраненные данные остановленной Machine. В отличие от "snapshot", он не содержит информацию о времени выполнения. Если используется это значение, то атрибут «relatedImage» не должен присутствовать. **SNAPSHOT**: Этот тип представляет сохраненные данные Machine. Если Machine не была в остановленном состоянии в момент создания данного образа, он также содержит информацию о времени выполнения. Если используется это значение, то атрибут «relatedImage» должен ссылаться на последний созданный образ Machine или образ, из которого была восстановлена Machine, что позволяет легко находить «предыдущий» снимок состояния. Атрибут «relatedImage» не должен устанавливаться Потребителями. **PARTIAL_SNAPSHOT**: Данный тип использует ту же семантику, что и SNAPSHOT MachineImage за исключением того, что он просто содержит изменения (дельты) состояния Machine по отношению к экземпляру MachineImage, на который ссылается атрибут «relatedImage», а не полное представление Machine. Если MachineImage будет удален, то следует использовать следующую семантику: - любые экземпляры MachineImage типа «SNAPSHOT», ссылающиеся на удаляемый экземпляр в атрибуте «relatedImage», должны установить этот атрибут равным значению «relatedImage» удаляемого экземпляра MachineImage; - любые экземпляры MachineImage типа «PARTIAL_SNAPSHOT», ссылающиеся на удаляемый экземпляр в атрибуте «relatedImage», также должны быть удалены. Это требование применяется рекурсивно к любому последующему экземпляру MachineImage типа «PARTIAL_SNAPSHOT». **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
imageLocation	*URI*	Ссылка на местоположение двоичных данных, которые составляют данный образ. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
relatedImage	*ref*	Ссылка на другой Ресурс MachineImage, связанный с данным. Фактическое значение этого атрибута варьируется в зависимости от типа MachineImage. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"type": *string*,
"imageLocation": string,
"relatedImage": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<MachineImage xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<type>*xs:string*</type>
<imageLocation>*xs:anyURI*</imageLocation>
<relatedImage href="*xs:anyURI*"/> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MachineImage>

5.14.7.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MachineImageCollection.

При создании нового Ресурса MachineImage представление нового MachineImage может содержать ссылку в атрибуте «imageLocation». Поставщики должны проверить эту ссылку (например, способом HTTPHEAD), чтобы определить необходимость какой-либо специальной обработки. В настоящем стандарте установлены следующие дополнительные шаги, которые должны предпринять Поставщики в зависимости от типа Ресурса, на который ссылаются:

gost.e-ecolog.ru

Если «imageLocation» является ссылкой на Machine, то Поставщик должен создать новый экземпляр MachineImage типа SNAPSHOT, основанный на Machine, на которую указывает ссылка. После завершения операции create атрибут «imageLocation»MachineImage не должен ссылаться на Machine (так как по прошествии времени Machine могла измениться), однако вместо этого он должен ссылаться на статическое представление Machine (или содержать данные о нем). Кроме того, у Machine, на которую идет ссылка, должен быть обновлен атрибут MachineSnapshotCollection, чтобы включать ссылку на этот недавно созданный снимок состояния в Ресурсе MachineImage.

5.14.8 Ресурс MachineImageCollection

Ресурс MachineImageCollection представляет собой Набор Ресурсов MachineImage Поставщика и соответствует спецификации Набора, определенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"machineImages": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты MachineImage...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MachineImage>
<id>*xs:anyURI*</id>
... остальные атрибуты MachineImage...
</MachineImage> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.8.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов MachineImage поддерживается путем отправки запроса POST на адрес URI операции "add"в соответствии с 4.2.1.1. Описание запроса и способ обработки приведены в 5.14.7.1.

5.14.9 Ресурс Credential

Ресурс Credential содержит информацию, необходимую для создания начального административного суперпользователя недавно созданной Machine или для представления учетных данных, необходимых для выполнения определенной операции. Из-за различий между операционными системами и Поставщиками настоящий стандарт не выделяет одну конкретную совокупность атрибутов для поддержки всех реализаций. Однако предполагается, что Поставщики будут расширять данный Ресурс дополнительными атрибутами, чтобы обеспечить соответствие их требованиям.

Например Поставщик может расширить данный Ресурс атрибутами "username" и "password", которые в этом случае будут информацией о входе в систему для новых экземпляров Machine. Эти дополнительные атрибуты могут появиться на одном уровне с общими атрибутами, такими, как "name" и "description". Атрибуты Credential приведены в таблице 28.

Таблица 28 — Атрибуты Credential

Наименование	Учетные данные	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
TBD		Точная совокупность атрибутов определена Поставщиком

Некоторые общие атрибуты расширения, которые могут использоваться Поставщиками, включают в себя указанные в таблицах 29 и 30:

Таблица 29 — Атрибуты наименования пользователя/пароля

Атрибут	Тип	Описание
userName	*string*	Наименование пользователя начального суперпользователя. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
password	*string*	Пароль начального суперпользователя. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для записи

Таблица 30 — Атрибуты открытого ключа

Атрибут	Тип	Описание
key	*byte []*	Открытый ключ для начального суперпользователя. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании Credential Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 30, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии с приведенным далее. При этом сериализацию представляют следующим образом (см. 1.3):

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Credential xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</Credential>

5.14.9.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс CredentialCollection.

5.14.10 Ресурс CredentialCollection

Ресурс CredentialCollection представляет собой Набор Ресурсов Credential Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"credential": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Credential...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Credential>
<id>*xs:anyURI*</id>
... остальные атрибуты Credential...
</Credentials> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.14.10.1 Операции

Примечание — Операция «add» требует, чтобы использовался CredentialTemplate (см. 4.2.1.1).

5.14.11 Ресурс CredentialTemplate

Данный Ресурс содержит конфигурационные параметры для создания Ресурса Credential. Один экземпляр CredentialTemplate может использоваться для создания нескольких экземпляров Credential. Атрибуты CredentialTemplate приведены в таблице 31.

Таблица 31 — Атрибуты CredentialTemplate

Наименование	CredentialTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
TBD		Точная совокупность атрибутов определена поставщиком.

При реализации или использовании CredentialTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 31, а также таблицам, описывающим связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии с приведенным далее. Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,

"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<CredentialTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</CredentialTemplate>

5.14.11.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс CredentialTemplateCollection.

5.14.12 Ресурс CredentialTemplateCollection

Ресурс CredentialTemplateCollection представляет собой Набор Ресурсов CredentialTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализациация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"credentialTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты CredentialTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<CredentialTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты CredentialTemplate...
</CredentialTemplate> *

<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.14.12.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса CredentialTemplate поддерживается путем отправки запроса POST на адрес URI операции «add» в соответствии с 4.2.1.1.

5.15 Ресурсы Volume и отношения

Ресурсы, используемые при создании Volume, и связи между ними приведены на рисунке 4. Данный рисунок представляет собой схему отношений Ресурсов, при этом использование UML не является обязательным требованием.

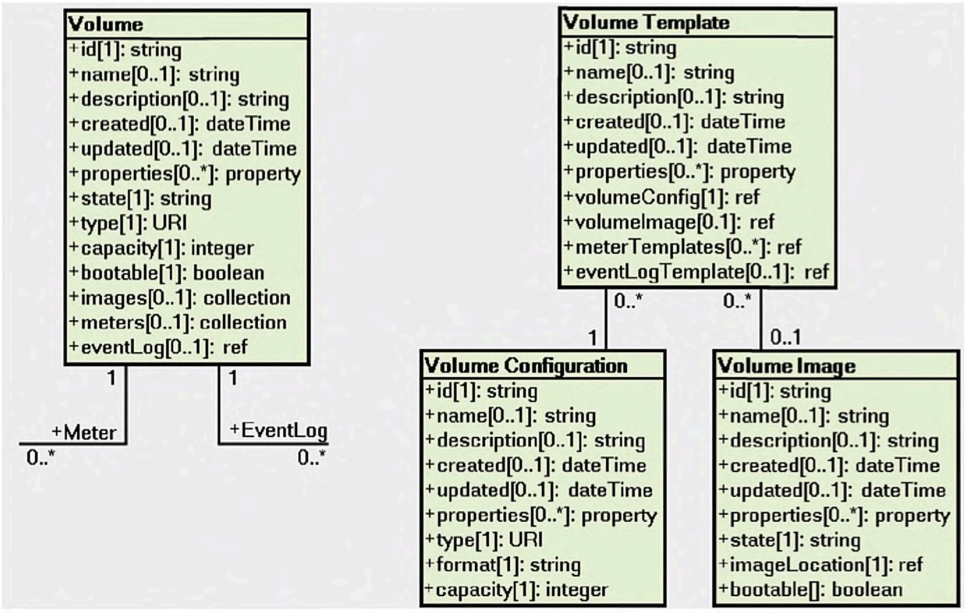

Рисунок 4 — Ресурсы Volume

5.15.1 Volume

Ресурс Volume представляет собой хранилище для System на уровне устройства в виде блока или файловой системы. Ресурсы Volume могут быть подключены к экземплярам Machine. После подключения получить доступ к Volume можно с помощью процессов на такой Machine. Атрибуты Volume приведены в таблице 32.

Таблица 32 — Атрибуты Volume

Наименование	Volume	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние Volume. Допустимые значения включают в себя: **CREATING**: Volume находится в процессе создания. **AVAILABLE**: Volume доступен и готов к использованию.Если не определено иное, после успешного создания Volume должен быть в этом состоянии. **CAPTURING**: Volume находится в процессе создания снимка для нового VolumeImage. Допустимым действием в этом состоянии является **delete**. **DELETING**: Volume находится в процессе удаления.

Окончание таблицы 32

Атрибут	Тип	Описание
state	*string*	**ERROR**: Поставщик обнаружил ошибку в Volume. Операции, которые приводят к переходам к указанным состояниям, указаны в 5.15.1.2 **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
type	*URI*	URI, который указывает на тип создаваемого Ресурса Volume. В настоящем стандарте установлен следующий URI: gost.e-ecolog.ru : Указывает, что создаваемыйVolume может совместно использоваться несколькими экземплярами Machine без явной операции монтирования в гостевой операционной системе. Могут определяться дополнительные значения. Если некоторые типы Volume требуют дополнительных данных, предполагается, что данный Ресурс расширен. Например для типа "sharedFileSystem" может потребоваться, чтобы были определены дополнительные сведения о сети и учетных данных. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
capacity	*integer*	Максимальный размер Volume, если он ограничен, в килобайтах. Если это значение увеличивается, Volume может содержать больше данных. При уменьшении этого значения может потребоваться проведение дополнительных оценок. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
bootable	*boolean*	Свойство указывает на то, является ли данный Volume загрузочным. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
images	*collection [VolumeVolume-Image]*	Ссылка на перечень ссылок на VolumeImage, которые представляют снимки состояния данного экземпляра Volume. Примечание — Тип Ресурса VolumeVolumeImage представляет собой связь между Volume и VolumeImage согласно 5.15.1.1.1. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meters	*collection [Meter]*	Ссылка на перечень Meter, используемых для мониторинга данного Volume. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLog	*ref*	Ссылка на EventLog данного Volume. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

При реализации или использовании Volume Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 32, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?

"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"type": *string*,
"capacity": *number*,
"bootable": *boolean*,
"images": { "href": *string* }, ?
"meters": { "href": *string* }, ?
"eventLog": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...

Тип медиа XML: application/xml

Сериализация XML:

<Volume xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="xs:string"> xs:string </property> *
<state> xs:string </state>
<type> xs:anyURI </type>
<capacity> xs:integer </capacity>
<bootable> xs:boolean </bootable>
<images href="xs:anyURI"/> ?
<meters href="xs:anyURI"/> ?
<eventLog href="xs:anyURI"/> ?
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<*xs:any*>*
</Volume>

5.15.1.1 Наборы

В настоящем подпункте приведены Наборы Ресурсов, принадлежащих Volumes.

5.15.1.1.1 Набор VolumeVolumeImageCollection

Типом ресурса для каждого элемента данного Набора является VolumeVolumeImage, приведенный в таблице 33.

Таблица 33 — Атрибуты VolumeVolumeImage

Наименование	VolumeVolumeImage	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
volumeImage	*ref*	Ссылка на Ресурс VolumeImage, который представляет снимок состояния данного Volume. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Сериализация JSON:

```
{ "resourceURI":
"gost.e-ecolog.ru                                                    ",
"id": string,
"count": number,
"volumeVolumeImages": [
{ "resourceURI":
"gost.e-ecolog.ru                                                    ",
"id": string,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"volumeImage": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
] ?
...
}
```

Сериализация XML:

```
<Collection
resourceURI="gost.e-ecolog.ru                                VolumeImageCollection"
xmlns="gost.e-ecolog.ru                                      ">
<id>xs:anyURI</id>
<count>xs:integer</count>
<VolumeVolumeImage>
<id>xs:anyURI</id>
<name>xs:string</name> ?
<description>xs:string</description> ?
<created>xs:dateTime</created> ?
<updated>xs:dateTime</updated> ?
<property key="xs:string">xs:string</property> *
<volumeImage href="xs:anyURI"/>
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<xs:any>*
</VolumeVolumeImage> *
<xs:any>*
</Collection>
```

Примечание — Предыдущие версии данной спецификации включали в себя операцию "add" для данного Ресурса. В настоящее время вместо этого рекомендуется создание нового экземпляра VolumeImage с атрибутом imageLocation, указывающим на Volume, чей снимок состояния хранится в созданном экземпляре.

5.15.1.1.2 Набор VolumeMeterCollection

Типом ресурса для каждого элемента этого Набора является Meter (см. 5.17.3).

Сериализация JSON:

```
{ "resourceURI":
"gost.e-ecolog.ru                                                    ",
"id": string,
"count": number,
"meters": [
```

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter...
*},` +
], ?
"operations*": [{ "rel": "add", "href": string } ?]*
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter...
</Meter> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.15.1.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс VolumeCollection.

5.15.2 Ресурс VolumeCollection

Ресурс VolumeCollection представляет собой Набор Ресурсов Volume Поставщика и соответствует спецификации Набора, приведенный в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"volumes": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Volume...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Volume>
<id>*xs:anyURI*</id>
... остальные атрибуты Volume...
</Volume> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.15.2.1 Операции

Примечание — Операция «add» требует, чтобы использовался VolumeTemplate (см. 4.2.1.1).

5.15.3 Ресурс VolumeTemplate

Данный Ресурс содержит конфигурационные параметры для создания Volume. Один экземпляр VolumeTemplate может использоваться для создания нескольких экземпляров Volume. Атрибуты VolumeTemplate приведены в таблице 34.

Таблица 34 — Атрибуты VolumeTemplate

Наименование	VolumeTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
volumeConfig	*ref*	Ссылка на VolumeConfiguration, который используется для создания Volume из данного шаблона VolumeTemplate. Примечание — Могут определяться атрибуты VolumeConfiguration, а не ссылка на существующий Ресурс VolumeConfiguration. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
volumeImage	*ref*	Ссылка на VolumeImage, который используется для создания Volume из данного шаблона VolumeTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
meterTemplates	*meter Templates []*	Перечень ссылок на шаблоны MeterTemplate, которые должны использоваться для создания и подключения набора новых Meter к создаваемому Volume. Примечание — Могут определяться атрибуты MeterTemplate, а не ссылка на существующий Ресурс MeterTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
eventLogTemplate	*ref*	Ссылка на EventLogTemplate, который должен использоваться для создания и подключения нового EventLog к создаваемому Volume. Примечание — Могут определяться атрибуты EventLogTemplate, а не ссылка на существующий Ресурс EventLogTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

При реализации или использовании VolumeTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 34, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"volumeConfig": {
"href": *string* |... *VolumeConfiguration...*
},
"volumeImage": { "href": *string* }, ?
"meterTemplates": [

{ "href": *string*, ?
... *MeterTemplate...* ?
}, *
], ?
"EventLogTemplate": {
"href": *string,* ?
... *EventLogTemplate...* ?
}, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<VolumeTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated> xs:dateTime </updated> ?
<property key="xs:string"> xs:string </property> *
<volumeConfig href="xs:anyURI"?>
... VolumeConfiguration... ?
</volumeConfig>
<volumeImage href="xs:anyURI"/> ?
<meterTemplate href="xs:anyURI"? >
... MeterTemplate... ?
</meterTemplate> *
<eventLogTemplate href="xs:anyURI"? >
... EventLogTemplate... ?
</eventLogTemplate> ?
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<xs:any>*
</VolumeTemplate>

5.15.3.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс VolumeTemplateCollection.

5.15.4 Ресурс VolumeTemplateCollection

Ресурс VolumeTemplateCollection представляет собой Набор Ресурсов VolumeTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"volumeTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты VolumeTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<VolumeTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты Volume Templates...
</VolumeTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.15.4.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса VolumeTemplate поддерживается путем отправки запроса POST на адрес URI операции "add" в соответствии с 4.2.1.1.

5.15.5 Ресурс VolumeConfiguration

Ресурс VolumeConfiguration содержит набор параметров конфигурации для создания Volume с заданными характеристиками. VolumeConfiguration создаются Поставщиками, но по их усмотрению, могут быть созданы Потребителями.

Атрибуты VolumeConfiguration приведены в таблице 35.

Таблица 35 — Атрибуты VolumeConfiguration

Наименование	VolumeConfiguration	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
type	*URI*	URI, указывающий на тип создаваемого Ресурса Volume. Настоящий стандарт определяет следующий URI: gost.e-ecolog.ru : Указывает, что создаваемый Volume может совместно использоваться несколькими экземплярами Machine без явной операции монтирования в гостевой операционной системе. Могут определяться дополнительные значения. Если некоторые типы Volume требуют дополнительных данных, предполагается, что данный Ресурс расширен. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
format	*string*	Формат файловой системы, которая будет размещена на Ресурсах Volume, создаваемых из данной конфигурации. Данный атрибут является значимым только для тех VolumeConfiguration, которые описывают блочные устройства. Данный атрибут является необязательным; отсутствие этого атрибута указывает, что Ресурсы Volume, созданные из данной конфигурации, не отформатированы, на них нет файловой системы. Значения в качестве примера: "ext4", "ntfs". **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
capacity	*integer*	Размер по умолчанию в килобайтах, если ограничен РесурсомVolume, созданным из данного VolumeConfiguration. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,

"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"type": string,
"format": string,
"capacity": number,
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<VolumeConfiguration xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<type>*xs:anyURI*</type>
<format>*xs:string*</format>
<capacity>*xs:integer*</capacity>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</VolumeConfiguration>

5.15.5.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс VolumeConfigurationCollection.

5.15.6 Ресурс VolumeConfigurationCollection

Ресурс VolumeConfigurationCollection представляет собой Набор Ресурсов VolumeConfiguration Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"volumeConfigurations": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты VolumeConfiguration...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>

<count>*xs:integer*</count>
<VolumeConfiguration>
<id>*xs:anyURI*</id>
... остальные атрибуты VolumeConfiguration...
</VolumeConfiguration> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.15.6.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов VolumeImage поддерживается путем отправки запроса POST на адрес URI операции «add» в соответствии с 4.2.1.1.

5.15.7 Ресурс VolumeImage

Данный Ресурс представляет образ, который может быть помещен в предварительно загруженный Volume. Атрибуты VolumeImage приведены в таблице 36.

Таблица 36 — Атрибуты VolumeImage

Наименование	VolumeImage	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние VolumeImage. Допустимые значения включают в себя: **CREATING**: VolumeImage находится в процессе создания. **AVAILABLE**: VolumeImage доступен и готов к использованию. Если не определено иное, то после успешного создания РесурсVolumeImage первоначально должен находиться в этом состоянии. **DELETING**: VolumeImage находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в VolumeImage. Операции, которые приводящие к переходам к указанным состояниям, приведены в 5.15.7.1. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
imageLocation	*ref*	Ссылка на местоположение двоичных данных, составляющих данный образ. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
bootable	*boolean*	Данное свойство указывает, являются ли Ресурсы Volume, созданные из данного VolumeImage, загрузочными. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализацию Ресурса и в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"imageLocation": { "href": *string* },

"bootable": *boolean*,
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<VolumeImage xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<imageLocation href="*xs:anyURI*"/>
<bootable>*xs:boolean*</bootable>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</VolumeImage>

5.15.7.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурсы VolumeImageCollection.

5.15.8 Ресурс VolumeImageCollection

Ресурс VolumeImageCollection представляет собой Набор Ресурсов VolumeImage Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"volumeImages": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты VolumeImage...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<VolumeImage>
<id>*xs:anyURI*</id>
... остальные атрибуты VolumeImage...
</VolumeImage> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.15.8.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов VolumeImage поддерживается путем отправки запроса POST на адрес URI операции "add"в соответствии с 4.2.1.1.

Если во время создания нового Ресурса VolumeImage атрибут "imageLocation" относится к существующему Volume, то эта операция должна интерпретироваться как запрос на создание снимка состояния этого ресурса Volume. После ее завершения атрибут "imageLocation" нового Ресурса VolumeImage не должен указывать на исходный Volume; вместо этого он должен ссылаться на статическую копию Volume. Кроме этого VolumeVolumeImageCollection ресурса Volume, на который указывала ссылка, должен быть обновлен: в него должна быть добавлена ссылка на созданный Ресурс VolumeImage, содержащий снимок состояния. При необходимости во время этого процесса Поставщик может перевести Volume в состояние CAPTURING.

5.16 Сетевые ресурсы и отношения

Ресурсы, вовлеченные в создание ресурсов Network и связанных с ними NetworkPort, а также отношения между ними представлены на рисунке 5. Данный рисунок представляет собой схему отношений Ресурсов, при этом использование UML не является обязательным требованием.

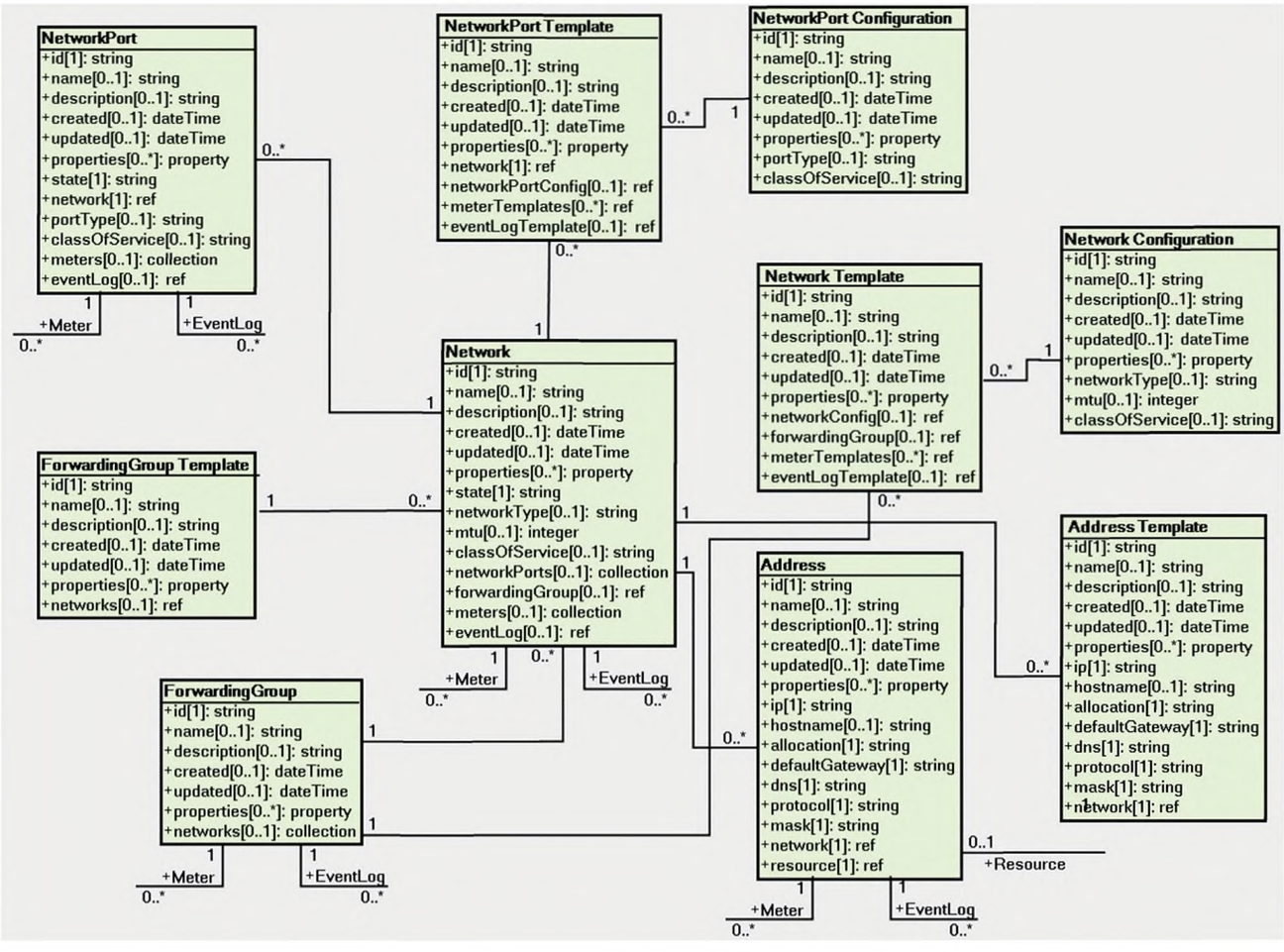

Рисунок 5 — Сетевые ресурсы

5.16.1 Ресурс Network

Ресурс Network представляет собой Набор связанных между собой логических служб, предназначенных для передачи трафика данных между оконечными узлами.

Ресурсы Network в ForwardingGroup должны иметь один и тот же атрибут «networkType», чтобы предотвратить утечку данных для сетей с ограниченным доступом, если они будут в одной группе ForwardingGroup с общедоступными сетями. Атрибуты Network приведены в таблице 37.

Таблица 37 — Атрибуты Network

Наименование	Сеть	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние Network. Допустимые значения включают в себя: **CREATING**: Network находится в процессе создания. **STARTING**: Network находится в процессе запуска. **STARTED**: Network доступен и готов к использованию. **STOPPING**: Network находится в процессе остановки. **STOPPED**: Network остановлен и не доступен для использования. **DELETING**: Network находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в Network. Операции, приводящие к переходам в указанные состояния, приведены в 5.16.1.2.Начальное состояние Сети приведено в 5.16.2.1. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
networkType	*string*	Индикатор того, есть ли у Ресурса Machine доступ к сети общего пользования или частной сети. Допустимые значения включают в себя: **PUBLIC**: представляет открытую сеть, трафик из которой может передаваться в Интернет. **PRIVATE**: определяет локальную закрытую сеть. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
mtu	*integer*	(Maximum Transmission Unit — максимальный размер передаваемого пакета) Максимальный размер пакета, поддерживаемый в данной сети. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
classOfService	*string*	Поддерживаемая Поставщиком категория, объединяющая Набор атрибутов, характеризующих уровень качества обслуживания. Следующие значения приведены в качестве примера: **GOLD**: высокая пропускная возможность, низкая задержка, низкие частотные отклонения (джиттер). **SILVER**: улучшенное, по сравнению с бронзовым классом, качество передачи голосового и видео трафика. **BRONZE**: отсутствие гарантий доставки. Спецификация возможных значений и подразумеваемое качество их обслуживания не входит в область применения настоящего стандарта. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
networkPorts	*Collection* [Network Network Port]	Ссылка на перечень NetworkPort, которые связаны с данной Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
forwardingGroup	*ref*	Ссылка на ForwardingGroup, в который входит данный Ресурс Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
meters	*Collection [Meter]*	Ссылка на перечень Ресурсов Meter, которые используются для мониторинга данного экземпляра Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

Окончание таблицы 37

Атрибут	Тип	Описание
eventLog	*ref*	Ссылка на EventLog данного экземпляра Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

При реализации или использовании Network Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 37, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"networkType": *string*, ?
"mtu": *number*, ?
"classOfService": *string*, ?
"networkPorts": { "href": *string* }, ?
"forwardingGroup": { "href": *string* }, ?
"meters": { "href": *string* }, ?
"eventLog": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru , "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Network xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<networkType>*xs:string*</networkType> ?
<mtu> xs:integer </mtu> ?
<classOfService> xs:string </classOfService> ?
<networkPorts href="xs:anyURI"/> ?
<forwardingGroup href="xs:anyURI"/> ?
<meters href="xs:anyURI"/> ?
<eventLog" href="xs:anyURI"/> ?
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?

<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<xs:any>*
</Network>

5.16.1.1 Наборы

Следующие пункты описывают Наборы Ресурсов, принадлежащие Network.

5.16.1.1.1 Ресурс NetworkNetworkPortCollection

Если Ресурсы NetworkPort созданы с помощью операции "add" ресурса NetworkPortCollection, принадлежащего некоторому ресурсу Network, они также должны быть добавлены к глобальному Набору, принадлежащему Точке входа в облако, NetworkPortCollection.

При удалении Ресурса Network все ее Наборы, а также Ресурсы этих Наборов, должны быть удалены согласно 5.5.12. Это означает, что все Ресурсы NetworkPort, связанные с удаляемой Network, также должны быть удалены.

Типом ресурса для каждого элемента этого Набора является NetworkPort, приведенный в 5.16.7.

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": string,
"count": number,
"networkNetworkPorts": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"networkPort": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
] ?
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /NetworkNetworkPortCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkNetworkPort>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description> xs:string </description> ?
<created> xs:dateTime </created> ?
<updated> xs:dateTime </updated> ?
<property key="*xs:string*"> xs:string </property> *
<networkPort href="xs:anyURI"/>
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?

<*xs:any*>*
</NetworkNetworkPort> *
<*xs:any*>*
</Collection>

5.16.1.1.2 PecypcNetworkMeterCollection

Типом ресурса для каждого элемента этого Набора является Meter, приведенный в 5.17.3.

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meters": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter...
</Meter> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.1.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkCollection.

Кроме этого выполняют следующие специальные операции:

start

/link@rel: gost.e-ecolog.ru

Данная операция должна запустить Ресурс Network.

Параметры входа: Нет.

Параметры выхода: Нет.

Во время обработки этой операции ресурс Network должен быть в состоянии STARTING.

После успешного завершения этой операции Network должен быть в состоянии STARTED.

Протокол HTTP

Для запуска Network по адресу URI «gost.e-ecolog.ru Ресурса Network передают запрос POST, где запрос HTTP должен быть следующим.

Тип медиа JSON:application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<property key="*xs:string*">*xs:string*</property> *

<*xs:any*>*

</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

stop

/link@rel: gost.e-ecolog.ru

Данная операция должна остановить Network. Если Network остановлена, она не должна передавать данные.

Параметры входа: Нет.

Параметры выхода: Нет.

Во время обработки данной операции Network должна быть в состоянии «STOPPING».

После успешного завершения данной операции Network должна быть в состоянии «STOPPED».

Протокол HTTP

Для остановки Network в URI «gost.e-ecolog.ru Ресурса Network передают запрос POST, где запрос HTTP должен включать в себя следующее.

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",

"action": "gost.e-ecolog.ru ",

"properties": { *string*: string, + } ?

...

}

Тип медиа XML: application/xml

Сериализация XML :

<Action xmlns="gost.e-ecolog.ru ">

<action> gost.e-ecolog.ru </action>

<property key="*xs:string*">*xs:string*</property> *

<*xs:any*>*

</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

5.16.2 Ресурс NetworkCollection

Ресурс NetworkCollection представляет собой Набор Ресурсов Network Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",

"id": *string*,

"count": *number*,

"networks": [

{ "resourceURI": "gost.e-ecolog.ru ",

"id": *string*,

... остальные атрибуты Network...

}, +

], ?

"operations": [{ "rel": "add", "href": *string* } ?]

...

}

Сериализация XML:

<Collection

resourceURI="gost.e-ecolog.ru /NetworkCollection"

xmlns="gost.e-ecolog.ru ">

<id>*xs:anyURI*</id>

<count>*xs:integer*</count>
<Network>
<id>*xs:anyURI*</id>
... остальные атрибуты Network...
</Network> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.2.1 Операции

Примечание — Операция «add» требует, чтобы использовался NetworkTemplate (см. 4.2.1.1).

После успешной обработки операции "add", если иное не определено в атрибуте "initialState" шаблона NetworkTemplate, состояние нового Network должно быть значением возможности DefaultInitialState Ресурса ResourceMetadata Ресурса Network, если он определен. Если возможность DefaultInitialState не определена, значением по умолчанию должно быть STOPPED. Семантика "initialState" должна быть эквивалентна выполнению Поставщиком надлежащих действий с новым Network для перевода его в заданное состояние.

Если Поставщик не способен установить начальное состояние Network (указанное в шаблоне NetworkTemplate, либо подразумеваемое правилами), то создание Network должно завершиться с ошибкой.

5.16.3 Ресурс NetworkTemplate

Ресурс NetworkTemplate содержит набор конфигурационных параметров, необходимых для создания Network. Один экземпляр NetworkTemplate может использоваться для создания нескольких Ресурсов Network. Атрибуты NetworkTemplate приведены в таблице 38.

Таблица 38 — Атрибуты NetworkTemplate

Наименование	NetworkTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
initialState	*string*	Начальное состояние новой Network. Возможные значения включают в себя состояния, установленные для атрибута «state»Network, которые не являются промежуточными (например, STARTED, STOPPED), и определяемые действиями, которые поддерживаются Поставщиком. Поставщики должны предоставитьперечень доступных значений через возможность «initialStates» Ресурса Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
networkConfig	*ref*	Ссылка на NetworkConfiguration, используемый для создания Network из этого шаблона NetworkTemplate. Примечание — Могут определяться атрибуты NetworkConfiguration, а не ссылка на существующий Ресурс NetworkConfiguration. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
forwardingGroup	*ref*	Ссылка на ForwardingGroup, в которую входит данная Network. Примечание — Ресурсы Network всегда передают трафик сами себе; поэтому данный атрибут появляется только в тех случаях, где Network, созданная из этого Шаблона, переадресует трафик одной или нескольким дополнительным сетям Network. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

Окончание таблицы 38

Атрибут	Тип	Описание
meterTemplates	*Meter Templates []*	Перечень ссылок на MeterTemplate, который должен использоваться для создания и подключения набора экземпляров Meter к создаваемой Network. Примечание — Могут определяться атрибуты MeterTemplate, а не ссылка на существующий Ресурс MeterTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
eventLogTemplate	*ref*	Ссылка на EventLogTemplate, который должен использоваться для создания и подключения нового EventLog к создаваемой Network. Примечание — Могут определяться атрибуты EventLogTemplate, а не ссылка на существующий Ресурс EventLogTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

При реализации или использовании NetworkTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 38, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

JSON тип медиа: application/json

Сериализация JSON:

{"resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"initialState": *string*, ?
"networkConfig": {
"href": *string* |... *NetworkingConfiguration...*
}, ?
"forwardingGroup": { "href": *string* }, ?
"meterTemplates": [
{ "href": *string*, ?
... *MeterTemplate...* ?
}, *
], ?
"eventLogTemplate": {
"href": *string*, ?
... *EventLogTemplate...* ?
}, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<NetworkTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?

<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<initialState>*xs:string*</initialState> ?
<networkConfig href="*xs:anyURI*"?>
... *NetworkConfiguration...* ?
</networkConfig> ?
<forwardingGroup href="xs:anyURI"/> ?
<meterTemplate href="xs:anyURI"? >
... MeterTemplate... ?
</meterTemplate> *
<eventLogTemplate href="xs:anyURI"? >
... EventLogTemplate... ?
</eventLogTemplate> ?
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<xs:any>*
</NetworkTemplate>

5.16.3.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkTemplateCollection.

5.16.4 Ресурс NetworkTemplateCollection

Ресурс NetworkTemplateCollection представляет собой Набор Ресурсов NetworkTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"networkTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты NetworkTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты NetworkTemplate...
</NetworkTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.4.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса NetworkTemplate поддерживается путем отправки запроса POST к URI операции «add» в соответствии с 4.2.1.1.

5.16.5 Ресурс NetworkConfiguration

Совокупность параметров конфигурации, приведенная в таблице 39, представляет информацию, необходимую для создания Network с заданными характеристиками.

Таблица 39 — Атрибуты NetworkConfiguration

Наименование	NetworkConfiguration	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
networkType	*string*	Индикатор, указывающий, является ли Network сетью общего пользования или частной сетью. Допустимые значения включают в себя: **PUBLIC**: представляет открытую сеть, трафик из которой может передаваться в Интернет. **PRIVATE**: определяет локальную закрытую сеть. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
mtu	*integer*	Maximum Transmission Unit является максимальным размером передаваемого пакета. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
classOfService	*string*	Поддерживаемая Поставщиком категория, объединяющая Набор атрибутов, характеризующих уровень качества обслуживания. Следующие значения приведены в качестве примера: **GOLD**: Высокая пропускная возможность, низкая задержка, низкие частотные отклонения (джиттер). **SILVER**: улучшенное по сравнению с бронзовым классом качество передачи голосового и видео трафика. **BRONZE**: отсутствие гарантий доставки. Спецификация возможных значений и подразумеваемое качество их обслуживания не входит в область применения настоящего стандарта. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись

Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": string, ?
"properties": { string: string, + }, ?
"networkType": string, ?
"mtu": number, ?
"classOfService": string, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<NetworkConfiguration xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<networkType>*xs:string*</networkType> ?
<mtu>*xs:integer*<mtu> ?
<classOfService>*xs:string*</classOfService> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</NetworkConfiguration>

5.16.5.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkConfigurationCollection.

5.16.6 Ресурс NetworkConfigurationCollection

Ресурс NetworkConfigurationCollection представляет собой Набор Ресурсов NetworkConfiguration Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": number,
"networkConfigurations": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
... остальные атрибуты NetworkConfiguration...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkConfiguration>
<id>*xs:anyURI*</id>
... остальные атрибуты NetworkConfiguration...
</NetworkConfiguration> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.6.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса NetworkConfiguration поддерживается путем отправки запроса POST на адрес URI операции «add» в соответствии с 4.2.1.1.

5.16.7 Ресурс NetworkPort

Ресурс NetworkPort реализует точку подключения Ресурса (например,Machine) к Network. Атрибуты NetworkPort приведены в таблице 40.

Таблица 40 — Атрибуты NetworkPort

Наименование	NetworkPort	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
state	*string*	Рабочее состояние NetworkPort. Допустимые значения включают в себя следующее: **CREATING**: NetworkPort находится в процессе создания. **STARTED**: NetworkPort доступен (включен) и готов к использованию. **STOPPED**: NetworkPort остановлен (отключен) и не доступен для использования. **DELETING**: NetworkPort находится в процессе удаления. **ERROR**: Поставщик обнаружил ошибку в NetworkPort. Операции, приводящие к переходам к указанным состояниям, приведены в 5.16.7.2. Начальное состояние NetworkPort указано в 5.16.8.1. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
network	*ref*	Ссылка на Network, связанный с данным NetworkPort. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
portType	*string*	Порт используется либо в качестве порта доступа (элемента сети), либо как магистральный порт, который служит транспортом для нескольких сетей. Допустимые значения включают в себя: **ACCESS**: элемент сети. **TRUNK**: транспорт для нескольких сетей. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
classOfService	*string*	Поддерживаемая Поставщиком категория, объединяющая Набор атрибутов, характеризующих уровень качества обслуживания. Следующие значения приведены в качестве примера: **GOLD**: высокая пропускная возможность, низкая задержка, низкие частотные отклонения (джиттер). **SILVER**: улучшенное по сравнению с бронзовым классом качество передачи голосового и видео трафика. **BRONZE**: отсутствие гарантий доставки. Спецификация возможных значений и подразумеваемое качество их обслуживания не входит в область применения настоящего стандарта. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
meters	*collection [Meter]*	Ссылка на перечень Ресурсов Meter, используемых для мониторинга данного экземпляра NetworkPort. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения
eventLog	*ref*	Ссылка на EventLog данного NetworkPort. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; только для чтения

При реализации или использовании NetworkPort Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 40, а также таблицам, описывающим встро-

енные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"network": { "href": string },
"portType": string, ?
"classOfService": string, ?
"meters": { "href": *string* }, ?
"eventLog": { "href": *string* }, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<NetworkPort xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<network href="*xs:anyURI*"/>
<portType>*xs:string*</portType> ?
<classOfService>*xs:string*</classOfService> ?
<meters href="*xs:anyURI*"/> ?
<eventLog" href="xs:anyURI"/> ?
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<xs:any>*
</NetworkPort>

5.16.7.1 Наборы

В настоящем подпункте приведены Наборы Ресурсов, принадлежащие NetworkPorts.

5.16.7.1.1 Ресурс NetworkPortMeterCollection

Типом ресурса для каждого элемента этого Набора является Meter, приведенный в 5.17.3.

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,

"count": *number*,
"meters": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter...
</Meter> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.16.7.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkPortCollection.

При удалении NetworkPort данный Ресурс должен быть удален из глобального Набора (принадлежащего Точке входа в облако) NetworkPortCollection, а также из NetworkPortsCollection соответствующей сети Network.

Кроме этого выполняют следующие специальные операции:

start

/link@rel: gost.e-ecolog.ru

Эта операция должна запустить NetworkPort.

Параметры входа: Нет.

Параметры выхода: Нет.

После успешного завершения этой операции NetworkPort должен быть в состоянии STARTED.

Протокол HTTP

Для запуска NetworkPort передают запрос POST в URI «gost.e-ecolog.ru Ресурса NetworkPort, где запрос HTTP должен быть следующим.

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="*xs:string*">*xs:string*</property> *
*<xs:any>**
</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

stop

/link@rel: gost.e-ecolog.ru

Эта операция должна остановить NetworkPort. После остановки NetworkPort он не должен быть доступен для использования, и через него не должен проходить сетевой трафик.

Параметры входа: Нет.

Параметры выхода: Нет.

После успешного завершения этой операции NetworkPort должен быть в состоянии STOPPED.

Протокол HTTP

Для остановки NetworkPort передают запрос POST в URI «gost.e-ecolog.ru
stop» Ресурса NetworkPort, где запрос HTTP должен быть следующим.

Тип медиа JSON:application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="*xs:string*">*xs:string*</property> *
<*xs:any*>*
</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

5.16.8 Ресурс NetworkPortCollection

Ресурс NetworkPortCollection представляет собой Набор Ресурсов NetworkPort Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"networkPorts": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты NetworkPort...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /NetworkPortCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkPort>
<id>*xs:anyURI*</id>
... остальные атрибуты NetworkPort...
</NetworkPort> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.8.1 Операции

Примечание — Операция "add" требует, чтобы использовался шаблон NetworkPortTemplate (см. 4.2.1.1).

Если Ресурсы NetworkPort созданы с помощью операции "add" глобального, т. е. принадлежащего Точке входа в облако, Набора NetworkPortCollection, они также автоматически добавляются к Ресурсу NetworkPortCollection соответствующей сети Network.

После успешного выполнения операции "add", если иное не определено в атрибуте "initialState" шаблона NetworkPortTemplate, состояние нового NetworkPort должно быть значением возможности DefaultInitialState в Ресурсе ResourceMetadata Ресурса NetworkPort, если таковой определен. Если возможность DefaultInitialState не определена, то значением по умолчанию должно быть STOPPED. Семантика "initialState" должна быть эквивалентна выполнению Поставщиком надлежащих действий с новым NetworkPort для перевода его в заданное состояние.

Если Поставщик неспособен установить начальное состояние NetworkPort, указанное в шаблоне NetworkPortTemplate либо подразумеваемое правилами, то создание NetworkPort должно завершиться с ошибкой.

5.16.9 Ресурс NetworkPortTemplate

Ресурс NetworkPortTemplate содержит набор конфигурационных параметров для создания Ресурса NetworkPort. Ресурс NetworkPortTemplate может использоваться для создания нескольких Ресурсов NetworkPort. Атрибуты NetworkPortTemplate приведены в таблице 41.

Таблица 41 — Атрибуты NetworkPortTemplate

Наименование	NetworkPortTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
initialState	*string*	Начальное состояние нового NetworkPort. Возможные значения включают в себя такие состояния, определенные для атрибута "state" Ресурса
initialState	*string*	NetworkPort, которые не являются промежуточными (например, STARTED, STOPPED), и определяются действиями, которые поддерживаются Поставщиком. Поставщики должны распространить перечень доступных значений с помощью возможности "initialStates" Ресурса ResourceMetadata, принадлежащего РесурсуNetworkPort. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
network	*ref*	Ссылка на сеть, которая будет связана с создаваемым NetworkPort. Если данный Шаблон используется для создания нового NetworkPort через глобальный Набор, принадлежащий Точке входа в облако NetworkPortCollection, то данный атрибут должен быть указан. Если данный Шаблон используется для создания нового NetworkPort через Набор NetworkPortsCollection, принадлежащий Ресурсу Network, данный атрибут должен отсутствовать или иметь то же значение, что и "id" Ресурса Network, к которому добавляется данный NetworkPort. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
networkPortConfig	*ref*	Ссылка на NetworkPortConfiguration, который используется для создания NetworkPort из данного NetworkPortTemplate. Примечание — Могут определяться атрибуты NetworkPortConfiguration, а не ссылка на существующий Ресурс NetworkPortConfiguration. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Окончание таблицы 41

Атрибут	Тип	Описание
meterTemplates	*meterTemplates []*	Перечень ссылок на MeterTemplate, которые должны использоваться для создания и подключения набора новых Ресурсов Meter к создаваемому NetworkPort. Примечание — Могут определяться атрибуты MeterTemplate, а не ссылка на существующий Ресурс MeterTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
eventLogTemplate	*ref*	Ссылка на EventLogTemplate, который должен использоваться для создания и подключения нового EventLog к создаваемому NetworkPort. Примечание — Могут определяться атрибуты EventLogTemplate, а не ссылка на существующий Ресурс EventLogTemplate. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый **Потребитель:** необязательная поддержка; чтение-запись

При реализации или использовании NetworkPortTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 41, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"initialState": *string*, ?
"network": { "href": string }, ?
"networkPortConfig": {
"href": string |... NetworkPortConfiguration...
},
"meterTemplates": [
{ "href": string, ?
... MeterTemplate... ?
}, *
], ?
"eventLogTemplate": {
"href": string, ?
... EventLogTemplate... ?
}, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<NetworkPortTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>

<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<initialState>*xs:string*</initialState> ?
<network href="*xs:anyURI*"/> ?
<networkPortConfig href="*xs:anyURI*"?>
... *NetworkPortConfiguration...* ?
</networkPortConfig>
<meterTemplate href="*xs:anyURI*"? >
... *MeterTemplate...* ?
</meterTemplate> *
<eventLogTemplate href="*xs:anyURI*"? >
... *EventLogTemplate...* ?
</eventLogTemplate> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</NetworkPortTemplate>

5.16.9.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkPortTemplateCollection.

5.16.10 Ресурс NetworkPortTemplateCollection

Ресурс NetworkPortTemplateCollection представляет собой Набор Ресурсов NetworkPortTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"networkPortTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты NetworkPortTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkPortTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты NetworkPortTemplate...
</NetworkPortTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.10.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса NetworkPortTemplate поддерживается путем отправки запроса POST к URI операции «add» в соответствии с 4.2.1.1.

5.16.11 Ресурс NetworkPortConfiguration

Совокупность параметров конфигурации для создания NetworkPort с заданными характеристиками. Атрибуты NetworkPortConfiguration приведены в таблице 42.

Таблица 42 — Атрибуты NetworkPortConfiguration

Наименование	NetworkPortConfiguration	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
portType	*string*	Порт используется либо в качестве порта доступа (элемента сети), либо как магистральный порт, который служит транспортом для нескольких сетей. Допустимые значения включают в себя следующее: **ACCESS**: элемент сети. **TRUNK**: транспорт для нескольких сетей. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
classOfService	*string*	Поддерживаемая Поставщиком категория, объединяющая Набор атрибутов, характеризующих уровень качества обслуживания. Следующие значения приведены в качестве примера: **GOLD**: высокая пропускная возможность, низкая задержка, низкие частотные отклонения (джиттер). **SILVER**: улучшенное по сравнению с бронзовым классом качество передачи голосового и видео трафика. **BRONZE**: отсутствие гарантий доставки. Спецификация возможных значений и подразумеваемое качество их обслуживания не входит в область применения настоящего стандарта. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"portType": *string*, ?
"classOfService": *string*, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<NetworkPortConfiguration xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<portType>*xs:string*</portType> ?
<classOfService>*xs:string*</classOfService> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</NetworkPortConfiguration>

5.16.11.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс NetworkPortConfigurationCollection.

5.16.12 Ресурс NetworkPortConfigurationCollection

Ресурс NetworkPortConfigurationCollection представляет собой Набор Ресурсов NetworkPortConfiguration Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"networkPortConfigurations": [
{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
... остальные *атрибуты NetworkPortConfiguration...*
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru :imi/1/NetworkPortConfigurationCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<NetworkPortConfiguration>
<id>*xs:anyURI*</id>
... остальные атрибуты NetworkPortConfiguration...
</NetworkPortConfiguration> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.16.12.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов NetworkPortConfiguration поддерживается путем отправки запроса POST к URI операции "add" в соответствии с 4.2.1.1.

5.16.13 Ресурс Address

Ресурс Addres представляет собой IP-адрес и метаданные, связанные с ним, для конкретной сети. Если Потребитель создает Ресурс Address, это семантически эквивалентно запросу статического IP-адреса, который позже может быть связан с Ресурсами. Ресурсы Addresse, созданные Потребителями вручную, не должны удаляться автоматически при удалении Ресурса, использующего данный Address (например, Machine), так как предполагается, что срок жизни Ресурсов Address, созданных вручную, отличается от срока жизни Ресурсов, которые их используют. Ресурсы Address, созданные Поставщиками от имени Потребителя, должны удаляться по усмотрению Поставщика. В частности, Поставщик должен удалять Ресурсы Address, которые он создал от имени Потребителя, если удален Ресурс, использующий данный Address, или если Addres отсоединяется от Ресурса.

Ресурсы Address, созданные Поставщиками, могут быть преобразованы в адреса, находящиеся под контролем Потребителя (т. е. не удаляются до явного требования Потребителя), путем изменения атрибута "allocation" с "dynamic" на "static", если такая функция поддерживается Поставщиками. Атрибуты Address приведены в таблице 43.

Таблица 43 — Атрибуты Address

Наименование	Address	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
ip	*string*	IP-адрес, присвоенный виртуальному интерфейсу. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
hostname	*string*	Разрешимое наименованиеDNS, связанное с этим сетевым интерфейсом. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
allocation	*string*	Значение «dynamic» либо «static». Указывает на того, кто управляет данным адресом: Поставщик или Потребитель. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
defaultGateway	*string*	IP-адрес маршрутизатора (шлюза), который обслуживает другие сети. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
dns	*string []*	IP-адрес сервера службы доменных наименований для разрешения IP адресов. **Ограничения:**
dns	*string []*	**Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
protocol	*string*	Выбранный сетевой протокол, напримерIPv4 или IPv6. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
mask	*string*	Сетевая маска, связанная с данным Address. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
network	*ref*	Ссылка на Network, с которой связан данный Address. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Окончание таблицы 43

Атрибут	Тип	Описание
resource	*ref*	Ссылка на Ресурс, который использует данный Address. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

При реализации или использовании Addres Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 43, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"ip": *string*,
"hostname": *string*, ?
"allocation": *string*,
"defaultGateway": *string*, ?
"dns": [*string*, +], ?
"protocol": string,
"mask": string, ?
"network": { "href": string },
"resource": { "href": string }, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Address xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<ip>*xs:string*</ip>
<hostname>*xs:string*</hostname> ?
<allocation>*xs:string*</allocation>
<defaultGateway>*xs:string*</defaultGateway> ?
<dns>*xs:string*</dns> *
<protocol>*xs:string*</protocol>
<mask>*xs:string*</mask> ?
<network href="*xs:anyURI*"/>
<resource href="*xs:anyURI*"/> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</Address>

5.16.13.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс AddressCollection.

5.16.14 Ресурс AddressCollection

Ресурс AddressCollection представляет собой Набор Ресурсов Addres Поставщика, которыми владеет/управляет Потребитель или Поставщик, и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"addresses": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Address...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
…
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /AddressCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Address>
<id>*xs:anyURI*</id>
... остальные атрибуты Address...
</Address> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.16.14.1 Операции

Примечание — Операция «add» требует использования шаблона AddressTemplate (см. 4.2.1.1).

5.16.15 Ресурс AddressTemplate

Данный Ресурс содержит конфигурационные параметры для создания Ресурса Address. Один экземпляр AddressTemplate может использоваться для создания нескольких экземпляров Address Атрибуты AddressTemplate, приведенных в таблице 44.

Таблица 44 — Атрибуты AddressTemplate

Наименование	AddressTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
ip	*string*	IP-адрес, присвоенный виртуальному интерфейсу. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
hostname	*string*	Допустимое наименование DNS, связанное с этим сетевым интерфейсом. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый **Потребитель:** необязательная поддержка; чтение-запись

Окончание таблицы 44

Атрибут	Тип	Описание
allocation	*string*	Значение «dynamic» либо «static». Указывает на того, кто управляет данным адресом: Поставщик или Потребитель. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
defaultGateway	*string*	IP-адрес маршрутизатора (шлюза), который обслуживает другие сети. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
dns	*string []*	IP-адрес сервера службы доменных наименований для разрешения IP адресов. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
protocol	*string*	Выбранный сетевой протокол, например IPv4 или IPv6. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
mask	*string*	Сетевая маска, связанная с данным Address. **Ограничения:** **Поставщик:** необязательная поддержка; изменяемый. **Потребитель:** необязательная поддержка; чтение-запись
network	*ref*	Ссылка на Network, с которой связан данный Address. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании AddressTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 44, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"ip": *string*,
"hostname": *string*, ?
"allocation": *string*,
"defaultGateway": *string*, ?
"dns": [*string*, +], ?
"protocol": string,
"mask": string, ?
"network": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<AddressTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<ip>*xs:string*</ip>
<hostname>*xs:string*</hostname> ?
<allocation>*xs:string*</allocation>
<defaultGateway>*xs:string*</defaultGateway>
<dns>*xs:string*</dns> +
<protocol>*xs:string*</protocol>
<mask>*xs:string*</mask>
<network href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</AddressTemplate>

5.16.15.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс AddressTemplateCollection.

5.16.16 Ресурс AddressTemplateCollection

Ресурс AddressTemplateCollection представляет собой Набор Ресурсов AddressTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"addressTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты AddressTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<AddressTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты AddressTemplate...
</AddressTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.16.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов AddressTemplate поддерживается путем отправки запроса POST к URI "addLink" в соответствии с 4.2.1.1.

5.16.17 Ресурс ForwardingGroup

ForwardingGroup представляет собой совокупность Ресурсов Network, между которыми настроена маршрутизация.

Все сети в ForwardingGroup должны иметь одинаковые атрибуты "networkType", чтобы предотвратить утечку данных для сетей с ограниченным доступом, если они будут в одной группе ForwardingGroup с общедоступными сетями.

Поставщики не должны позволять маршрутизацию между двумя экземплярами Network, если они не соединены явным образом, будучи частью общей ForwardingGroup. Атрибуты ForwardingGroup приведены в таблице 45.

Таблица 45 — Атрибуты ForwardingGroup

Наименование	ForwardingGroup	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
networks	*collection [ForwardingGroup Network]*	Ссылка на перечень ссылок на Network в данной ForwardingGroup. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

При реализации или использовании ForwardingGroup Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 45, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"networks": [
{ "href": *string* }, +
], ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<ForwardingGroup xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<network href="*xs:anyURI*"> *

<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
*<xs:any>**
</ForwardingGroup>

5.16.17.1 Наборы

Следующие подпункты описывают Наборы Ресурсов, принадлежащие ForwardingGroups.

5.16.17.1.1 Набор ForwardingGroupNetworkCollection

Типом ресурса для каждого элемента этого Набора является ForwardingGroupNetwork, приведенный в таблице 46.

Таблица 46 — Атрибуты ForwardingGroupNetwork

Наименование	ForwardingGroupNetwork	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
network	*ref*	Ссылка на Network в ForwardingGroup. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": string,
"count": number,
"forwardingGroupNetworks": [
{ "resourceURI":
"gost.e-ecolog.ru ",
"id": string,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"network": { "href": string },
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<ForwardingGroupNetwork>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?

<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<network href="*xs:anyURI*"/>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</ForwardingGroupNetwork> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.17.2 Операции

Данный Ресурс поддерживает Операции Read, Update и Delete. Операция Create поддерживается через Ресурс ForwardingGroupCollection.

5.16.18 Ресурс ForwardingGroupCollection

Ресурс ForwardingGroupCollection представляет собой Набор Ресурсов ForwardingGroup Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"*count*": *number*,
"forwardingGroups": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты ForwardingGroup...
}, +
], ?
"*operations*": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<ForwardingGroup>
<id>*xs:anyURI*</id>
... остальные атрибуты ForwardingGroup...
</ForwardingGroup> *
<operation rel="add" href="*xs:anyURI*"/> ?
<*xs:any*>*
</Collection>

5.16.18.1 Операции

Примечание — Операция "add" требует использования ForwardingGroupTemplate (см. 4.2.1.1).

5.16.19 Ресурс ForwardingGroupTemplate

Данный Ресурс содержит конфигурационные параметры для создания Ресурсов ForwardingGroup. Один экземпляр ForwardingGroupTemplate может использоваться для создания нескольких экземпляров ForwardingGroups. Атрибуты ForwardingGroupTemplate приведены в таблице 47.

Таблица 47 — Атрибуты ForwardingGroupTemplate

Наименование	ForwardingGroupTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
networks	*ref []*	Массив ссылок на Network в данной группе ForwardingGroup. Название единицы массива: сеть **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании ForwardingGroupTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 47, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Потребитель и Поставщик должны сериализовать данный Ресурс в соответствии со следующим описанием. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"networks": [
{ "href": *string* }, +
], ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
…
}

Тип медиа XML: application/xml

Сериализация XML:

<ForwardingGroupTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<network href="*xs:anyURI*"> *
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</ForwardingGroupTemplate>

5.16.19.1 Операции

Данный Ресурс поддерживает Операции Read, Update и Delete. Операция Create поддерживается через Ресурс ForwardingGroupTemplateCollection.

5.16.20 Ресурс ForwardingGroupTemplateCollection

Ресурс ForwardingGroupTemplateCollection представляет собой Набор Ресурсов ForwardingGroupTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"forwardingGroupTemplates": [
{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты ForwardingGroupTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<ForwardingGroupTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты ForwardingGroupTemplate...
</ForwardingGroupTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.16.20.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов ForwardingGroupTemplate поддерживается путем отправки запроса POST к URI операции «add»в соответствии с 4.2.1.1.

5.17 Ресурсы мониторинга и отношения

Ресурсы, используемые при отслеживании выполнения операций, а также при измерении и мониторинге состояния других Ресурсов, приведены на рисунке 6, который представлен в виде схемы отношений Ресурсов, при этом использование UML не является обязательным требованием.

5.17.1 Ресурс Job

Данный Ресурс представляет собой процесс (т. е. последовательность одной или нескольких операций, направленных на достижение определенной цели), который выполняется Поставщиком.

Если Поставщик поддерживает предоставление Ресурсов Job Потребителям, то каждый запрос от Потребителя, который вызывает изменения в окружении, может привести к созданию Ресурса Job; в этом случае Потребителям, инициировавшим запросы, должна быть доступна абсолютная ссылка URI к созданному Ресурсу Job. Поставщики могут создать дополнительные Ресурсы Job для операций, инициируемых Поставщиками, если Поставщик предоставляет Ресурсы Job Потребителям.

Если задание Job завершено неуспешно (например, находится в состоянии FAILED или STOPPED), в настоящем стандарте не установлены какие-либо требования к Поставщику, обеспечивающие гарантированное сохранение задействованных Ресурсов в определенных состояниях. На основании внешних условий в этот момент времени Поставщик может принять решение отменить эффекты операции; приостановить процесс; провести некоторые действия по очистке; либо предпринять какие-либо другие действия. Однако Поставщики должны перечислить все Ресурсы, на которые повлияло выполнение задания Job в атрибуте "affectedResources", таким образом давая Потребителям возможность определить состояние Ресурсов самостоятельно. В случае, когда Ресурс удален, ссылки на такой Ресурс не должны появляться в атрибуте «affectedResources».

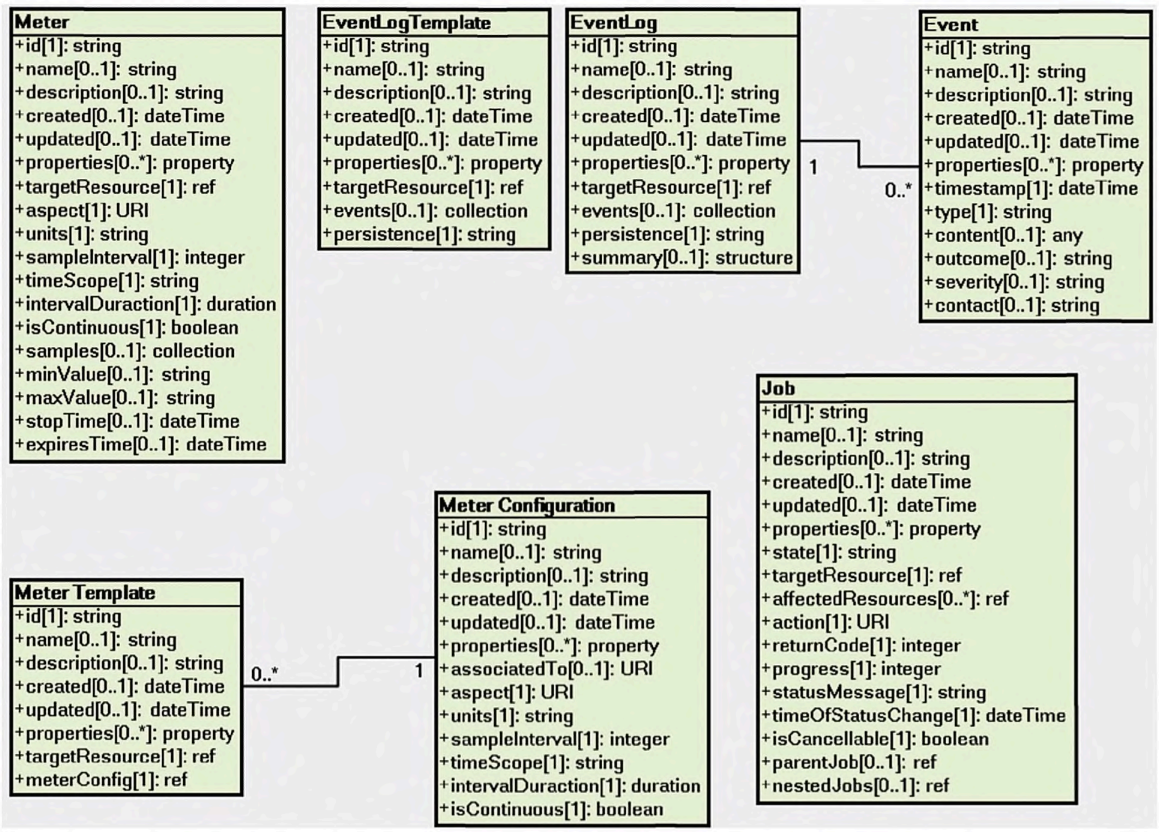

Рисунок 6 — Ресурсы мониторинга

Ресурс Job допускает вложение Ресурсов Job. Настоящий стандарт не распространяется на определение преобразования единичной в несколько вложенных Ресурсов Job. Однако если существуют вложенные задания, то самый верхний ресурс Job должен предоставить общую информацию о состоянии всех заданий и находиться в состоянии SUCCESS, только если все вложенные задания также находится в состоянии SUCCESS. Если созданы вложенные Ресурсы Job, для самого верхнего Ресурса Job не требуется ссылаться на все затронутые Ресурсы в его атрибуте "affectedResources". Потребитель должен проверить все вложенные ресурсы Job, чтобы определить полный перечень Ресурсов, на которые повлияли задания. Атрибуты Job приведены в таблице 48.

Таблица 48 — Атрибуты Job

Наименование	Job	
Тип URI	gost.e-ecolog.ru cimi/1/Job	
Атрибут	Тип	Описание
state	*string*	Состояние процесса, связанного с данной операцией. Допустимые значения включают в себя: **QUEUED**: Указывает, что операция еще не начала обрабатываться. **RUNNING**: Указывает, что операция все еще выполняется. **FAILED**: Указывает, что операция была завершена неуспешно. **SUCCESS**: Указывает, что операция завершена успешно. **STOPPING**: Указывает, что операция находится в процессе остановки. **STOPPED**: Указывает, что операция была прервана. Операции, приводящие к переходам к указанным состояниям, приведены в 5.17.1.1.

Продолжение таблицы 48

Атрибут	Тип	Описание
state	*string*	**Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
targetResource	*ref*	Ссылка на Ресурс верхнего уровня, с которым выполняется операция. Как правило этим Ресурсом является Ресурс, для которого была вызвана операция. Примечание — Если задание "add" выполняется на Наборе (например, MachineCollection), атрибут targetResource должен ссылаться на Ресурс Набора, поскольку именно в этом Ресурсе выполняется данная операция. Кроме того, созданный Ресурс должен появиться в атрибуте "affectedResources". **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
affectedResources	*ref []*	Перечень ссылок на Ресурсы, на которые повлияло данное задание. Примечание — Переченьв обязательном порядке должен содержать ссылку "targetResource". Наименование элемента массива: affectedResource **Ограничения**: **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
action	*URI*	URI, указывающий на тип выполняемой операции. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
returnCode	*integer*	Код возврата операции. Конкретное значение, определенное для реализации. В настоящем стандарте зарезервированы для использования значения в диапазоне от 0 до 9999. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
progress	*integer*	Целое число в диапазоне от 0 до 100, указывающее прогресс выполнения задания. Если задание больше не выполняется, то вне зависимости от результата значение должно быть равно 100. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
statusMessage	*string*	Строка, читаемая человеком, которая предоставляет информацию об операции и используется для того, чтобы затем квалифицировать или предоставить дополнительную информацию о текущем статусе операции. Например данный атрибут может указать на причину того, почему операция была неуспешной либо была отменена Потребителем или Поставщиком. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
timeOfStatusChange	*dateTime*	Метка времени, определяющая момент времени, когда изменился статус операции. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

Окончание таблицы 48

Атрибут	Тип	Описание
parentJob	*ref*	Ссылка на Ресурс Job, для которого данный Ресурс является подчиненным. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
nestedJobs	*ref []*	Массив ссылок на совокупность вложенных заданий. Наименование объекта массива: nestedJob **Ограничения**: **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения

При реализации или использовании Job Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 48, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"state": *string*,
"targetResource": { "href": *string* },
"affectedResources": [{ "href": *string* }, +],
"action": *string*,
"returnCode": *number*,
"progress": *number*,
"statusMessage": *string*,
"timeOfStatusChange": *date*,
"parentJob": { "href": *string* }, ?
"nestedJobs": [
{ "href": *string* }, +
], ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* }, ?
{ "rel": "gost.e-ecolog.ru ", "href": *string* } ?
] ?
…
}

Тип медиа XML: application/xml

Сериализация XML:

<Job xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:datelime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<state>*xs:string*</state>
<targetResource href="*xs:anyURI*"/>

<affectedResource href="*xs:anyURI*"/> +
<action>*xs:anyURI*</action>
<returnCode>*xs:integer*</returnCode>
<progress>*xs:integer*<progress>
<statusMessage>*xs:string*</statusMessage>
<timeOfStatusChange>*xs:dateTime*</timeOfStatusChange>
<parentJob href="*xs:anyURI*"/> ?
<nestedJob href="*xs:anyURI*"/> *
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<operation rel="gost.e-ecolog.ru "
href="*xs:anyURI*"/> ?
<*xs:any*>*
</Job>

5.17.1.1 Операции Ресурса

Данный Ресурс поддерживает операции Read, Update и Delete. Удаление Ресурса Job в состоянии RUNNING должно быть эквивалентно остановке Job и последующему удалению. Ответом на запрос на удаление выполняющегося Ресурса Job, который не поддерживает действие "stop", должен быть отказ.

Кроме этого определены следующие специальные операции:

stop

/link@rel: gost.e-ecolog.ru

Данная операция должна остановить задание Job.

Параметры входа: Нет.

Параметры выхода: Нет.

Во время обработки этой операции Job должен быть в состоянии STOPPING.

После успешного завершения этой операции Job должен быть в состоянии "STOPPED".

Протокол HTTP

Чтобы остановить Job, необходимо отправить запрос POST к URI "gost.e-ecolog.ru action/stop" Ресурса Job, где запрос HTTP должен быть следующим.

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="xs:string"> xs:string </property> *
<xs:any>*
</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

5.17.2 Ресурс JobCollection

Ресурс JobCollection представляет собой Набор Ресурсов Job Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *integer*,
"jobs": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,

... остальные атрибуты Job...
}, +
] ?
...
}

Сериализация XML:

<Collection resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Job>
<id>*xs:anyURI*</id>
... остальные атрибуты Job...
</Job> *
*<xs:any>**
</Collection>

5.17.3 Ресурс Meter

Данный Ресурс представляет доступ к некоторому измеримому свойству, связанному с заданным Ресурсом.

Если Ресурс, на который ссылается «targetResource» Ресурса Meter, удален, то все Ресурсы Meter, связанные с этим Ресурсом, также должны быть удалены. Другими словами, удаление набора MetersCollection, определенного для Ресурса (например, MetersCollection Ресурса Machine), также должно привести к удалению всех Ресурсов Meter, на которые ссылается этот Набор. Атрибуты Meter приведены в таблице 49.

Таблица 49 — Атрибуты Meter

Наименование	Meter	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
targetResource	*ref*	Ссылка на Ресурс, с которым связан Meter. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
aspect	*URI*	Уникальный идентификатор, представляющий аспект измеряемого Ресурса. **Ограничения:**
aspect	*URI*	**Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
units	*string*	Наименование используемых единиц измерения, например килобиты в секунду, процент использования ЦП и т.д. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
sampleInterval	*integer*	Время между последовательными измерениями в секундах. **Ограничения;** **Поставщик:** обязательная поддержка; изменяемый **Потребитель:** обязательная поддержка; чтение-запись
timeScope	*string*	Временной аспект применения данного Ресурса Meter. Имеются два возможных значения: "Point" указывает, что Meter относится к моменту времени, "Interval" — что Meter относится к временному интервалу. Например можно определить Ресурс Meter, цель которого заключается в том, чтобы предоставить среднее использование ЦПУ за день. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения

Окончание таблицы 49

Атрибут	Тип	Описание
intervalDuration	*duration*	Продолжительность интервала в случае, когда timeScope установлен равным «Interval». Возможные значения: hourly, daily, weekly, monthly, yearly. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
isContinuous	*boolean*	Данное значение указывает, является ли показатель РесурсаMeter непрерывным или скалярным. PerformanceMeter служит примером линейной метрики. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
samples	*collection [Sample]*	Ссылка на перечень измерений. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
minValue	*string*	Ожидаемое минимальное значение показателя. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
maxValue	*string*	Ожидаемое максимальное значение показателя. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
stopTime	*dateTime*	Момент времени, с которого Ресурс Meter прекращает проводить измерения. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
expiresTime	*dateTime*	Момент времени, с которого Ресурс Meter больше не опрашивается. Что означает удаление Ресурса Meter по окончании этого момента времени. Примечание — Meter может быть удален до этого времени, если удален измеряемый Ресурс. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании Ресурса Meter Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 49, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"targetResource": { "href": *string* },
"aspect": *string*,
"units": *string*,

"sampleInterval": number,
"timeScope": string,
"intervalDuration": string,
"isContinuous": boolean,
"samples": { "href": string }, ?
"minValue": string, ?
"maxValue": string, ?
"stopTime": string, ?
"expiresTime": string, ?
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string }, ?
{ "rel": "gost.e-ecolog.ru ", "href": string },?
{ "rel": "gost.e-ecolog.ru ", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Meter xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<targetResource href="*xs:anyURI*"/>
<aspect>*xs:anyURI*</aspect>
<units>*xs:string*</units>
<sampleInterval>*xs:integer*</sampleInterval>
<timeScope>*xs:string*<timeScope>
<intervalDuration *xs:duration*</intervalDuration>
<isContinuous>*xs:boolean*</isContinuous>
<samples href="*xs:anyURI*"/> ?
<minValue>*xs:string*</minValue> ?
<maxValue>*xs:string*</maxValue> ?
<stopTime>*xs:dateTime*</stopTime> ?
<expiresTime>*xs:dateTime*</expiresTime> ?
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<operation rel="gost.e-ecolog.ru "
href="xs:anyURI"/> ?
<xs:any>*
</Meter>

5.17.3.1 Наборы

В настоящем подпункте приведены Наборы Ресурсов, принадлежащие Ресурсам Meter.

5.17.3.1.1 Ресурс SampleCollection

Типом ресурса для каждого элемента этого Набора является Sample, приведенный в таблице 50.

Таблица 50 — Атрибуты Sample

Наименование	Sample	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
timestamp	*dateTime*	Указывает на временной интервал, когда был измерен показатель timeScope = "Point". Если timeScope —"Interval", он указывает на момпент окончания временного интервала. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
value	*string*	Обозначает измеренное значение показателя. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения

При реализации или использовании Sample Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 50, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Набора Sample как в JSON, так и в XML представляют следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"samples": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"timestamp": string,
"value": string
...
}, +
], ?
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Sample>
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<sample timestamp="xs:dateTime" value="xs:string"/>
*<xs:any>**
</Sample> *

*<xs:any>**
</Collection>

5.17.3.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MeterCollection. Удаление Ресурса Meter должно приводить к исключению из атрибута Meter атрибута targetResource.

Кроме того, определены следующие специальные операции:

start

/link@rel: gost.e-ecolog.ru

Эта операция должна запускать Meter.

Параметры входа: Нет.

Параметры выхода: Нет.

После успешного завершения этой операции Meter должен начать записывать измерения показателя, связанного с его целевым Ресурсом.

Протокол HTTP

Для запуска Meter необходимо отправить запрос POST к URI "gost.e-ecolog.ru start" Ресурса Meter, где запрос HTTP должен быть следующим.

Тип медиа JSON:application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="xs:string"> xs:string </property> *
<xs:any>*
</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

stop

/link@rel: gost.e-ecolog.ru

Эта операция должна останавливать Meter.

Параметры входа: Нет.

Параметры выхода: Нет.

После успешного завершения этой операции Ресурс Meter не записывает измерения показателя, связанного с его целевым Ресурсом.

Протокол HTTP

Для остановки Ресурса Meter необходимо отправить запрос POST к URI "gost.e-ecolog.ru cimi/1/action/stop" Ресурса Meter, где запрос HTTP должен быть следующим.

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"action": "gost.e-ecolog.ru ",
"properties": { *string*: string, + } ?
...
}

Тип медиа XML: application/xml

Сериализация XML

<Action xmlns="gost.e-ecolog.ru ">
<action> gost.e-ecolog.ru </action>
<property key="xs:string"> xs:string </property> *
<xs:any>*
</Action>

После успешной обработки запроса текст ответа HTTP может быть пустым.

5.17.4 Ресурс MeterCollection

Ресурс MeterCollection представляет собой Набор Ресурсов Meter Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meters": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты Meter...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru org/cimi/1">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<Meter>
<id>*xs:anyURI*</id>
... остальные атрибуты Meter...
</Meter> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.4.1 Операции

Примечание — Операция "add" требует использования MeterTemplate (см. 4.2.1.1).

Если Ресурсы Meter создаются через операцию "add" глобального набора MeterCollection (принадлежащего Точке входа в облако), они также должны автоматически добавляться к Набору Meters соответствующего targetResource.

5.17.5 Ресурс MeterTemplate

Ресурс MeterTemplate представляет информацию, необходимую для создания нового экземпляра Meter. Атрибуты MeterTemplate приведены в таблице 51.

Таблица 51 — Атрибуты MeterTemplate

Наименование	MeterTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
targetResource	*ref*	Ссылка на измеряемый Ресурс. Тип Ресурса должен быть одним из типов "associatedTo", перечисленных в MeterConfiguration, на который ссылается данный шаблон. Если данный Шаблон используется для создания нового экземпляра Meter через глобальный Набор MetersCollection (принадлежащий Точке входа в облако), то данный атрибут должен присутствовать. Если данный Шаблон используется для создания нового экземпляра Meter через MetersCollection целевого Ресурса, данный атрибут должен отсутствовать или иметь такое же значение, что и "id" Ресурса, к которому добавляется создаваемый экземпляр Meter. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Окончание таблицы 51

Атрибут	Тип	Описание
meterConfig	*ref*	Ссылка на MeterConfiguration, который используется для создания Meter от данного MeterTemplate. Примечание — Могут определяться атрибуты MeterConfiguration, а не ссылка на существующий Ресурс MeterConfiguration. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании MeterTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 51, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"targetResource": { *string* },
"meterConfig": {
"href": *string* |... *MeterConfiguration...*
},
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<MeterTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<targetResource href="*xs:anyURI*"/>
<meterConfig href="*xs:anyURI*"?>
... *MeterConfiguration...* ?
</meterConfig>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
<*xs:any*>*
</MeterTemplate>

5.17.6 Ресурс MeterTemplateCollection

Ресурс MeterTemplateCollection представляет собой Набор Ресурсов MeterTemplate Поставщика и соответствует спецификации Набора, приведенный в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meterTemplates": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты MeterTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MeterTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты MeterTemplate...
</MeterTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.6.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание новых Ресурсов MeterTemplate поддерживается с помощью запроса POST к URI операции "add" в соответствии с 4.2.1.1.

5.17.7 Ресурс MeterConfiguration

Ресурс MeterConfiguration представляет определение Meter. Атрибуты MeterConfiguration приведены в таблице 52.

Таблица 52 — Атрибуты MeterConfiguration

Наименование	MeterConfiguration	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
associatedTo	*URI []*	Массив вида URI, указывающий на типы Ресурсов, к которым может быть применен Meter, созданный из этой конфигурации. Пространство значений этих URI идентично находящимся в ResourceMetadata.typeURI, который является URI, однозначно определяющим тип Ресурса. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
aspect	*URI*	Уникальный идентификатор, представляющий аспект измеряемого Ресурса (см. URI, приведенные в CIM, в таблице 53). **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
units	*string*	Наименование используемых единиц измерения, например, килобиты в секунду, процент использования ЦП и т.д. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Окончание таблицы 52

Атрибут	Тип	Описание
sampleInterval	*integer*	Интервал времени между последовательными измерениями в секундах. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
timeScope	*string*	Временной аспект применения создаваемого Ресурса Meter. Имеются два возможных значения: "Point" указывает, что Meter относится к моменту времени, "Interval" — что Meter относится к временному интервалу. Например, можно определить MeterConfiguration, цель которого заключается в определении среднего использования ЦПУ за день. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
intervalDuration	*duration*	Продолжительность временного интервала в случае, когда timeScope установлен равным "Interval". Возможные значения: hourly, daily, weekly, monthly, yearly. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись
isContinuous	*boolean*	Данное значение указывает, является ли показатель Meter непрерывным или скалярным. PerformanceMeter служит примером линейной метрики. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Сериализацию Ресурса в JSON и в XML представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
"name": string, ?
"description": string, ?
"created": string, ?
"updated": string, ?
"properties": { string: string, + }, ?
"associatedTo": [
{ "href": string }, +
], ?
"aspect": string,
"units": string,
"sampleInterval": number,
"timeScope": string,
"intervalDuration": string,
"isContinuous": boolean,
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
...
}

Тип медиа XML: application/xml

сериализация XML:

<MeterConfiguration xmlns="gost.e-ecolog.ru ">
<id> xs:anyURI </id>
<name> xs:string </name> ?

<description> xs:string </description> ?
<created> xs:dateTime </created> ?
<updated> xs:dateTime </updated> ?
<property key="xs:string"> xs:string </property> *
<associatedTo href="xs:anyURI"/> *
<aspect> xs:anyURI </aspect>
<units> xs:string </units>
<sampleInterval> xs:integer </sampleInterval>
<timeScope> xs:string </timeScope>
<intervalDuration> xs:duration </intervalDuration>
<isContinuous> xs:boolean </isContinuous>
<operation rel="edit" href="xs:anyURI"/> ?
<operation rel="delete" href="xs:anyURI"/> ?
<xs:any>*
</MeterConfiguration>

В таблице 53 приведены URI для атрибута «aspect», рассматриваемые в настоящем стандарте. Поставщики могут определять дополнительные URI аспектов, при этом рекомендуется, чтобы эти URI указывали на документы с описанием, из которого Потребители смогут получить сведения о дополнительных аспектах. Для краткости графа «Аспект» в таблице показывает только завершающую часть URI. Она должна быть добавлена к gost.e-ecolog.ru .

Таблица 53 — URI аспектов

Аспект	Описание
cpu	Процент использования ЦПУ Ресурсом. Обычно связан с Ресурсами System и Machine в CloudEntryPoint. Для Ресурсов, которые группируют другие Ресурсы (например, CloudEntryPoint или Ресурсы System), данный аспект представляет агрегированное значение процента использования ЦПУ
memory	Объем памяти, используемой Ресурсом. Обычно связан с Ресурсами System и Machine в CloudEntryPoint. Для Ресурсов, которые группируют другие Ресурсы (например, CloudEntryPoint или Ресурсы System), данный аспект представляет агрегированное значение использования памяти
disc	Объем дискового пространства, используемого Ресурсом. Обычно связан с Ресурсами System, Machine и Volume в CloudEntryPoint. Для Ресурсов, которые группируют другие Ресурсы (например, CloudEntryPoint или Ресурсы System), данный аспект представляет агрегированное значение использования дискового пространства
bandwidth	Объем сетевого трафика. Обычно связан с Ресурсами System и Network в CloudEntryPoint. Для Ресурсов CloudEntryPoint и System данный аспект представляет общую пропускную способность всех подчиненных сетей
inputBandwidth	Использование Ресурсом входящей пропускной способности. Обычно связано с Ресурсами Machine, NetworkPort и Volume. Для Ресурсов Machine данный аспект представляет использование совокупной входящей пропускной способности всех сетевых интерфейсов
outputBandwidth	Использование Ресурсом исходящей пропускной способности. Обычно связано с Ресурсами Machine, NetworkPort и Volume. Для Ресурсов Machine данный аспект обеспечивает использование совокупной исходящей пропускной способности всех сетевых интерфейсов

5.17.7.1 Операции

Данный Ресурс поддерживает операции Read, Update и Delete. Операция Create поддерживается через Ресурс MeterConfigurationCollection.

5.17.8 Ресурс MeterConfigurationCollection

Ресурс MeterConfigurationCollection представляет собой Набор Ресурсов MeterConfiguration Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"meterConfigurations": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты MeterConfiguration...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<MeterConfiguration>
<id>*xs:anyURI*</id>
... остальные атрибуты MeterConfiguration...
</MeterConfiguration> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.8.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса MeterConfiguration поддерживается путем отправки запроса POST к URI операции "add"в соответствии с 4.2.1.1.

5.17.9 Ресурс EventLog

Ресурс представляет реестр событий.

Если Ресурс, на который указывает атрибут "targetResource" в EventLog, удален, то EventLog, связанный с этим Ресурсом, также может быть удален. Другими словами, удаление Ресурса (например, Machine) может привести к удалению EventLog, на который имеется ссылка из этого Ресурса. Такое поведение определяется возможностью EventLog.Linked.

Если EventLog будет удален, то все его события также должны быть удалены. Атрибуты EventLog приведены таблице 54.

Таблица 54 — Атрибуты EventLog

Наименование	EventLog	
Тип URI	gost.e-ecolog.ru /EventLog	
Атрибут	Тип	Описание
Target-Resource	*ref*	Ссылка на Ресурс, с которым связаны события. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
events	*collection [Events]*	Ссылка на перечень произошедших событий. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; только для чтения
persistence	*string*	Значение, которое указывает на длительность хранения событий в EventLog. Например, daily (день), weekly (неделя), monthly (месяц) или yearly (год). События, которые хранятся дольше указанного срока, могут быть удалены.

Окончание таблицы 54

<table>
<tr><th>Атрибут</th><th>Тип</th><th>Описание</th></tr>
<tr><td>persistence</td><td>string</td><td><u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; чтение-запись</td></tr>
<tr><td>summary</td><td><unnamed-structure> (структура без наименования)</td><td>Сводка всех событий, существующих в EventLog, при выполнении операции чтения, сгруппированная по степени важности.
Каждый атрибут сводки — структура (без имени) со следующими атрибутами:
<table>
<tr><th>Наименование</th><th>Тип</th><th>Описание</th></tr>
<tr><td>low</td><td>integer</td><td>Число произошедших событий с низкой степенью важности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>medium</td><td>integer</td><td>Число произошедших событий со средней степенью важности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>high</td><td>integer</td><td>Число произошедших событий с высокой степенью важности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый
Потребитель: обязательная поддержка; только для чтения</td></tr>
<tr><td>critical</td><td>integer</td><td>Число произошедших событий с критической степенью важности.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый
Потребитель: обязательная поддержка; только для чтения</td></tr>
</table>
<u>Ограничения:</u>
Поставщик: обязательная поддержка; изменяемый.
Потребитель: обязательная поддержка; только для чтения</td></tr>
</table>

При реализации или использовании EventLog Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 54, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"targetResource": { "href": string },
"events": { "href": *string* },
"persistence": *string*,
"summary": {

"low": *number*,
"medium": *number*,
"high": *number*,
"critical": *number*
}, ?
"operations": [
{ "rel": "edit", "href": *string* }, ?
{ "rel": "delete", "href": *string* } ?
] ?
...
}

Тип медиа XML: application/xml

Сериализация XML :

<EventLog xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<targetResource href="*xs:anyURI*"/>
<events href="*xs:anyURI*"/>
<persistence>*xs:string*</persistence>
<summary>
<low>*xs:integer*</low>
<medium>*xs:integer*</medium>
<high>*xs:integer*<high>
<critical>*xs:integer*</critical>
</summary>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</EventLog>

5.17.9.1 Наборы

В настоящем подпункте приведены Наборы Ресурсов, принадлежащие Ресурсам EventLog.

5.17.9.1.1 Ресурс EventCollection

Типом ресурса для каждого элемента этого Набора является Event, приведенный в 5.17.13.

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"events": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": string,
... остальные атрибуты Event...
}, +
], ?
"operations": [{ "rel": "add", "href": string } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /EventCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>

<count>*xs:integer*</count>
<Event>
<id>*xs:anyURI*</id>
... остальные атрибуты Event...
</Event> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.9.2 Операции

Данный Ресурс поддерживает операции Read, Update и Delete.

5.17.10 Ресурс EventLogCollection

Ресурс EventLogCollection представляет собой Набор Ресурсов EventLog Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"eventLogs": [
{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты EventLog...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru /EventLogCollection"
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<EventLog>
<id>*xs:anyURI*</id>
... остальные атрибуты EventLog...
</EventLog> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.11 РесурсEventLogTemplate

Ресурс EventLogTemplate представляет информацию, необходимую для создания нового экземпляра EventLog. Атрибуты EventLogTemplate приведены в таблице 55.

Таблица 55 — Атрибуты EventLogTemplate

Наименование	EventLogTemplate	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
targetResource	*ref*	Ссылка на Ресурс, к которому должен быть подключен EventLog. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

Окончание таблицы 55

Атрибут	Тип	Описание
persistence	*string*	Значение, определяющее длительность хранения событий в EventLog. Например daily (день), weekly (неделя), monthly (месяц) или yearly (год). События, которые хранятся дольше указанного срока, могут быть удалены. **Ограничения:** **Поставщик:** обязательная поддержка; изменяемый. **Потребитель:** обязательная поддержка; чтение-запись

При реализации или использовании EventLogTemplate Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 55, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"targetResource": { string },
"persistence": string,
"operations": [
{ "rel": "edit", "href": string }, ?
{ "rel": "delete", "href": string } ?
] ?
…
}

Тип медиа XML: application/xml:

Сериализация XML :

<EventLogTemplate xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<targetResource href="*xs:anyURI*"/>
<persistence>*xs:string*</persistence>
<operation rel="edit" href="*xs:anyURI*"/> ?
<operation rel="delete" href="*xs:anyURI*"/> ?
*<xs:any>**
</MeterTemplate>

5.17.12 РесурсEventLogTemplateCollection

Ресурс EventLogTemplateCollection представляет собой Набор Ресурсов EventLogTemplate Поставщика и соответствует спецификации Набора, приведенной в 5.5.12. Данный Ресурс должен быть сериализован следующим образом:

Сериализация JSON:

{ "resourceURI":
"gost.e-ecolog.ru ",
"id": *string*,
"count": *number*,
"eventLogTemplates": [

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
... остальные атрибуты EventLogTemplate...
}, +
], ?
"operations": [{ "rel": "add", "href": *string* } ?]
...
}

Сериализация XML:

<Collection
resourceURI="gost.e-ecolog.ru "
xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<count>*xs:integer*</count>
<EventLogTemplate>
<id>*xs:anyURI*</id>
... остальные атрибуты EventLogTemplate...
</EventLogTemplate> *
<operation rel="add" href="*xs:anyURI*"/> ?
*<xs:any>**
</Collection>

5.17.12.1 Операции

Данный Ресурс поддерживает операции Read и Update. Создание нового Ресурса EventLogTemplate поддерживается путем отправки запроса POST к URI операции «add»в соответствии с 4.2.1.1.

5.17.13 Ресурс Event

Ресурс, который представляет возникновение события в пределах управляемой инфраструктуры. Примерами Ресурсов Event являются следующие:

- машина X была перезагружена гостевой ОС;
- машина X не отвечает на службы платформы;
- к машине X был добавлен новый vCPU вследствие применения правил эластичности.

Областью применения понятия Event является любая информация, которую Поставщик может отследить в пределах своей инфраструктуры и которая может быть полезна Потребителю. Возможные примеры включают в себя ошибки, происходящие в (виртуальных) ресурсах, присвоенных Потребителям; инициируемые Поставщиками действия, такие как задачи сопровождения и т.д. Атрибуты Event приведены в таблице 56.

Таблица 56 — Атрибуты Event

Наименование	Event	
Тип URI	gost.e-ecolog.ru	
Атрибут	Тип	Описание
timestamp	*dateTime*	Момент времени, когда произошло событие. Примечание — Данный атрибут не идентичен времени создания экземпляра Ресурса Event, которое сохранено в атрибуте "created". **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения
type	*URI*	URI, уникально идентифицирующий тип события. Если присутствует атрибут "content", данный атрибут отпределяет фактическую структуру данных для этого содержимого, например схему, связанную с ним. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения

Окончание таблицы 56

Атрибут	Тип	Описание
content	*any*	Полиморфный атрибут, представляющий подробные данные о событии, тип которых изменяется в зависимости от атрибута "type". Как правило, таким атрибутом является структура данных, например: При проведении мониторинга содержимое атрибута должно включать в себя целевой ID и тип Ресурса, измеренный(ые) атрибут(ы) и значение(я) состояния. При проведении аудита, соответствующего модели CADF, содержимое атрибута должно включать в себя подробную структуру события, которая удовлетворяет схеме события CADF. Для индикаторов CIM содержимое должно включать в себя структуру и атрибуты, определенные для таких событий. **Ограничения:** **Поставщик:** обязательная поддержка; неизменяемый. **Потребитель:** обязательная поддержка; только для чтения
outcome	*string*	Значение строки, характеризующее общую значимость события. Определен базовый набор, который может использоваться независимо от типа Event. Для каждого типа Event определение значения базового результата может быть уточнено в контексте данного типа, если это не входитв противоречие с общим значением outcome, приведенного далее.
outcome	*string*	Базовые значения outcome: **Pending**: Event сообщает о действии или процессе, который все еще продолжается. **Unknown**: Event сообщает о запросе или действии, которое неизвестно Поставщику. **Status**: Event дает отчет о состоянии или статусе Ресурса. **Success**: Event дает отчет об успешном результате некоторого действия или процесса. **Warning**: Event дает отчет о ситуации, которая требует внимания или мероприятий по восстановлению. **Failure**: Event дает отчет о неудаче некоторого действия или процесса. Эта совокупность базовых значений может быть расширена, чтобы адаптировать к возможным исходам в зависимости от типа события. В этом случае расширенный набор значений должен относиться ко всем Ресурсам Event этого типа. **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения
severity	*string*	Значение, указывающее на важность события. Возможными значениями являются: **critical (критический),** **high (высокий),** **medium (средний),** **low (низкий).** Интерпретация уровня важности может зависеть от атрибута "type". Если данный атрибут не применим к некоторому типу Event, его следует опустить. **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения
contact	*string*	Ссылка на контактную точку или точку обработки для обработки Event. Фактический тип этого содержимого (например, адрес электронной почты, телефонный номер справочной службы или сотрудника, очередь сообщений, URL и т. д.) зависит от атрибута "type" и определяется им. Данный атрибут является изменяемым, поскольку он может быть определен после создания Event Поставщиком. **Ограничения:** **Поставщик:** необязательная поддержка; неизменяемый. **Потребитель:** необязательная поддержка; только для чтения

Примечание — Существуют несколько исторических моделей событий, которые были стандартизированы или разработаны для различных областей IT. Цель CIMI состоит не в том, чтобы выбрать одну особую модель событий, а в том, чтобы выбрать такие атрибуты Event, которые содержат наиболее полезные данные для обработки событий в облачной среде. Тем не менее дополнительные данные о событии могут быть представлены в атрибуте Content, который допускает отображение другой модели событий в EventCIMI.

При реализации или использовании Event Поставщики и Потребители должны использовать синтаксис и семантику его атрибутов согласно таблице 56, а также таблицам, описывающим встроенные Ресурсы или связанные Наборы. Сериализацию Ресурса в JSON и в XML (см. 1.3) представляют следующим образом:

Тип медиа JSON: application/json

Сериализация JSON:

{ "resourceURI": "gost.e-ecolog.ru ",
"id": *string*,
"name": *string*, ?
"description": *string*, ?
"created": *string*, ?
"updated": *string*, ?
"properties": { *string*: *string*, + }, ?
"timestamp": *string*,
"type": *string*,
"content": *any*, ?
"outcome": *string*, ?
"severity": *string*, ?
"contact": *string*, ?
...
}

Тип медиа XML: application/xml

Сериализация XML:

<Event xmlns="gost.e-ecolog.ru ">
<id>*xs:anyURI*</id>
<name>*xs:string*</name> ?
<description>*xs:string*</description> ?
<created>*xs:dateTime*</created> ?
<updated>*xs:dateTime*</updated> ?
<property key="*xs:string*">*xs:string*</property> *
<timestamp>*xs:dateTime*</timestamp>
<type>*xs:string*</type>
<content>*xs:any** </content> ?
<outcome>*xs:string*</outcome> ?
<severity>*xs:string*</severity> ?
<contact>*xs:string*</contact> ?
*<xs:any>**
</Event>

В таблице 57 приведены URI для атрибута "type", на которые распространяется настоящий стандарт. Дополнительные типы могут быть добавлены Поставщиком, например с целью охарактеризовать внешние события, отображенные в инфраструктуре EventCIMI. Рекомендуется, чтобы эти URI указывали на документы с описанием, из которого Потребители смогут получить сведения о типе. Типы Event, приведенные в настоящем стандарте, используют базовый URI: gost.e-ecolog.ru . Если в графе «Тип события» в таблице приведен только относительный URI (например, state), такой URI должен быть добавлен в конец базового URI.

Таблица 57 — URI типов

<table>
<tr><th>Тип события</th><th>Описание</th></tr>
<tr><td>state</td><td>События этого типа предоставляют информацию о состоянии и времени выполнения ресурсов CIMI, таких как экземпляры Машин, Систем, Сетей и Томов. Эта информация включает в себя отчеты о любом изменении в атрибуте "state" этих Ресурсов.
Элемент content, связанный с данным типом Event, имеет следующую структуру:
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>resName</td><td>string</td><td>Наименование Ресурса, о состоянии которого сообщается.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
</table>

Продолжение таблицы 57

<table>
<tr><th>Тип события</th><th>Описание</th></tr>
<tr><td>state</td><td>
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>resource</td><td>ref</td><td>Ссылка на Ресурс, о состоянии которого сообщается.
Примечание — Ссылка может оказаться недопустимой, так как событие может пережить Ресурс.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>resType</td><td>URI</td><td>URI, обозначающий данный тип Ресурса (такой же, как и тип URI, связанный с типом Ресурса для этого Ресурса).
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения.</td></tr>
<tr><td>state</td><td>string</td><td>Состояние Ресурса должно совпадать со значением атрибута "state" (если он присутствует) Ресурса в момент времени начала события.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>previous</td><td>string</td><td>Предыдущее значение состояния, если событие сообщает об изменении состояния.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения.</td></tr>
</table>
</td></tr>
<tr><td>alarm</td><td>
События этого типа сообщают об ошибках или авариях, происходящих впроцессе операций менеджмента Облачных ресурсов. Эта информация включает в себя отказы обеспечения ресурсов, отказы выполнения запросов к интерфейсу CIMI и любую критическую ситуацию, которую необходимо своевременно рассмотреть.
Элемент content, связанный с этим типом события, имеет следующую структуру:
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>resName</td><td>string</td><td>Наименование Ресурса, связанное с этим событием, если возможно.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>resource</td><td>ref</td><td>Ссылка на Ресурс, связанная с этим событием, если возможно.
Примечание — Ссылка может стать недопустимой, так как событие может пережить Ресурс.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>res
type</td><td>URI</td><td>URI, обозначающий этот тип Ресурса, связанный с этим событием, если применимо (так же, как тип URI, связанный с типом Ресурса для этого Ресурса).
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>code</td><td>string</td><td>Код аварийного сигнала.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
</table>

Продолжение таблицы 57

<table>
<tr><th>Тип события</th><th>Описание</th></tr>
<tr><td>alarm</td><td>
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>detail</td><td>string</td><td>Подробная информация, связанная с этим событием.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
<tr><td>model</td><td>
События этого типа сообщают об изменениях в модели ресурса CIMI, которая включает в себя создание, изменение и разрушение экземпляров Ресурса; а также обновления метаданных (расширения Ресурса, возможности и ограничения и т.д.).
Элемент content, связанный с этим типом события, имеет следующую структуру:
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>resName</td><td>string</td><td>Наименование главной модели Ресурса, затронутого изменением.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>resource</td><td>ref</td><td>Ссылка на главную модель Ресурса, затронутого изменением.
Примечание — Ссылка может стать недопустимой, так как событие может пережить Ресурс.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>restype</td><td>URI</td><td>URI, обозначающий этот тип ресурса (так же, как тип URI, связанный с типом ресурса для этого Ресурса).
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>change</td><td>string</td><td>Вид изменения, о котором сообщает событие (create/update/delete).
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>detail</td><td>string</td><td>Подробная информация, связанная с изменением, обычно данные для обновления или создания, предоставленные в запросе.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
<tr><td>access</td><td>
События этого типа отслеживают все запросы доступа к некоторому Ресурсу поставщика CIMI.
Элемент content, связанный с этим типом события, имеет следующую структуру:
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>operation</td><td>string</td><td>Метод или наименование операции, использованной для доступа (для протокола HTTP это метод запроса HTTP).
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>resource</td><td>ref</td><td>Ссылка на первичный Ресурс, поддерживающий операцию (для протокола HTTP—URI Ресурса или URI, связанный с операцией).
Примечание — Ссылка может стать недопустимой, так как событие может пережить Ресурс.
<u>Ограничения:</u>
Поставщик: обязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
</table>

Окончание таблицы 57

<table>
<tr><th>Тип события</th><th>Описание</th></tr>
<tr><td></td><td>
<table>
<tr><th>Данные</th><th>Тип</th><th>Описание</th></tr>
<tr><td>detail</td><td>string</td><td>Подробная информация, связанная с изменением, как правило, данные для обновления или создания, которые используется в запросе.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
<tr><td>initiator</td><td>string</td><td>Детали, определяющие инициатора запроса в случае, если эта информация может быть связана с запросом.
<u>Ограничения:</u>
Поставщик: необязательная поддержка; неизменяемый.
Потребитель: необязательная поддержка; только для чтения</td></tr>
</table>
</td></tr>
<tr><td>gost.e-ecolog.ru
dmtf.org/cloud/
audit/1.0/</td><td>бытия этого типа представляют события, имеющие значение для аудита, в соответствии с CADF (…). Этот тип может подразделяться далее через продолжение пути в URI (например, gost.e-ecolog.ru для событий аудита безопасности).
Элемент content, связанный с этим типом события, имеет ту же структуру, что и сериализация события, определенная в CADF.</td></tr>
</table>

Сериализацию свойства "content" для различных типов событий представляют следующим образом:

событие "state":

Сериализация JSON:

{ "id": *string*,
…
"type": "gost.e-ecolog.ru /event/state",
"content": {
"resName": *string*,
"resource" : { "href" : *string* },
"resType" : *string*,
"state" : *string*,
"previous" : *string* ?
}
…
}

Сериализация XML:

<Event xmlns="gost.e-ecolog.ru ">
…
<type> gost.e-ecolog.ru </type>
<content>
<resName>*xs:string*</resName>
<resource href="*xs:anyURI*"/>
<resType>*xs:anyURI*</resType>
<state>*xs:string*</state>
<previous>*xs:string*</previous> ?
</content> ?
…
</Event>

Событие "alarm":

Сериализация JSON:

{ "id": *string*,
…
"type": "gost.e-ecolog.ru /event/alarm",
"content": {

"resName": *string* ?
"resource" : { "href" : *string* }, ?
"resType" : *string* ?
"code" : string,
"detail" : string ?
}
...
}

Сериализация XML:

<Event xmlns="gost.e-ecolog.ru ">
...
<type> gost.e-ecolog.ru </type>
<content>
<resname>*xs:string*</resname> ?
<resource href="*xs:anyURI*"/> ?
<restype>*xs:anyURI*</restype> ?
<code>*xs:string*</code>
<detail>*xs:string*</detail> ?
</content> ?
...
</Event>

Событие "model":

Сериализация JSON:

{ "id": *string*,
...
"type": "gost.e-ecolog.ru ",
"content": {
"resName": *string*, ?
"resource" : { "href" : *string* }, ?
"resType" : *string*, ?
"change" : *string*,
"detail" : *string* ?
}
...
}

Сериализация XML:

<Event xmlns="gost.e-ecolog.ru ">
...
<type> gost.e-ecolog.ru </type>
<content>
<resname>*xs:string*</resname> ?
<resource href="*xs:anyURI*"/> ?
<restype> xs:anyURI </restype> ?
<change> xs:string </change>
<detail> xs:string </detail> ?
</content> ?
...
</Event>

Событие "access":

Сериализация JSON:

{ "id": *string*,
...
"type": "gost.e-ecolog.ru ",
"content": {
"operation": *string*,
"resource" : { "href" : *string* },

"detail" : *string*, ?
"initiator" : *string* ?
}
...
}
Сериализация XML:
<Event xmlns="gost.e-ecolog.ru ">
...
<type> gost.e-ecolog.ru </type>
<content>
<operation>*xs:string*</operation>
<resource href="*xs:anyURI*"/>
<detail>*xs:string*</detail> ?
<initiator>*xs:string*</initiator> ?
</content> ?
...
</Event>
5.17.13.1 Операции
Данный Ресурс поддерживает операции Read, Update и Delete.

6 Вопросы безопасности

Существуют разнообразные методы защиты, которые могут использоваться. Настоящий стандарт не устанавливает определенных требований к конкретным методам. Поставщики должны предоставить достаточное количество информации о применяемых методах защиты, чтобы Потребитель мог реализовать необходимые алгоритмы для успешной связи с Поставщиком.

Приложение А
(обязательное)

Поддержка OVF в CIMI

В настоящем приложении определено, каким образом элементы дескриптора OVF отображаются на ресурсы CIMI и их атрибуты. Данное определение дает возможность создавать различные ресурсы CIMI путем импорта пакетов OVF. Это осуществляется путем задания ссылки на пакет OVF в операции import Ресурса SystemCollection или SystemTemplateCollection (тип медиа в таком URI должен быть "application/ovf"). Дополнительная информация по OVF приведена в [2].

Поддержка импорта и экспорта OVF является необязательной для Поставщика; число атрибутов OVF, доступных через CIMI, зависит от реализации. Поставщик может поддерживать импорт пакета OVF либо только для System, либо только для SystemTemplate, либо для обоих. Поддержку фактического импорта и экспорта пакета OVF осуществляет гипервизор под управлением реализации CIMI, и созданные таким образом ресурсы CIMI отражают работу гипервизора по импорту и формированию представления результатов.

Импорт пакета OVF может быть отражен при создании Шаблонов, которые позднее могут использоваться для создания экземпляров System, Machine и других компонентов Ресурсов. Импорт пакета OVF может также использоваться для непосредственного создания экземпляров System, Machine и других компонентов Ресурсов без создания Шаблонов.

В 5.13.4 приведено описание того, как следует импортировать файл OVF, чтобы создать SystemTemplate (и составляющие Ресурсы). Созданный таким образом шаблон SystemTemplate содержит ссылку на MachineTemplate для каждой VirtualSystem, которая определена в дескрипторе OVFVirtualSystemCollection.

Примечание — В настоящее время CIMI позволяет создавать экземпляры Systems, таким образом для каждого набора VirtualSystemCollection во вложенной совокупности наборов создается отдельный шаблон SystemTemplate в рамках «родительского» SystemTemplate с шаблонами MachineTemplate для каждой из систем VirtualSystem, содержащихся в данном наборе VirtualSystemCollection.

Значения атрибутов для MachineTemplate извлекаются из раздела VirtualHardwareSection описания системы VirtualSystem (обязательного в OVF). Если используется более одной секции VirtualHardwareSection для данной системы VirtualSystem (что допускается в OVF), результат зависит от реализации, но реализация может выбрать шаблон MachineTemplate из существующей (возможно, статической) совокупности, которая наилучшим образом подходит для VirtualHardwareSection. Элементы в VirtualHardwareSection отображаются в свойствах CIMI MachineConfiguration, при этом создается соответствующий Ресурс MachineConfiguration, который подсоединяется к созданному шаблону MachineTemplate для этой системы VirtualSystem.

Шаблоны CIMI VolumeTemplate созданы в соответствии с DiskSection дескриптора OVF и могут совместно использоваться в более чем в одной системе VirtualSystem (MachineTemplate CIMI), определенных в пакете OVF. Кроме того, из DiskSection может быть создан новый ресурс CIMI MachineImage, если определен ovf:fileRef для содержимого виртуального диска.

Шаблоны CIMI NetworkTemplate созданы в соответствии с NetworkSection дескриптора OVF наряду с элементами Connection в элементах VirtualHardwareSection, которые относятся к имеющим соответствующее наименование сетям.

В 5.13.2.1 приведено описание метода импортирования файла OVF для создания System (и составляющих Ресурсов). System, созданная таким образом, содержит ссылку на Machine для каждой VirtualSystem, которая определена в дескрипторе OVFVirtualSystemCollection.

Примечание — В настоящее время CIMI позволяет создавать экземпляры Systems, таким образом для каждого набора VirtualSystemCollection во вложенной совокупности наборов создается отдельный экземпляр System в рамках «родительского» экземпляра System, в котором для каждой из содержащихся систем VirtualSystem в этом наборе VirtualSystemCollectionсоздается экземпляр Machine.

Значения атрибутов для Machine извлекаются из раздела VirtualHardwareSection описания системы VirtualSystem (обязательного в OVF). Если используется более одной секции VirtualHardwareSection для данной системы VirtualSystem (что допускается в OVF), результат зависит от реализации. Элементы в VirtualHardwareSection отображаются в свойствах CIMIMachineConfiguration, при этом создается соответствующий Ресурс MachineConfiguration, который подсоединяется к созданному экземпляру Machine для этой системы VirtualSystem.

Экземпляры Volume CIMI создаются в соответствии с DiskSection дескриптора OVF и могут совместно использоваться в более чем в одной системе VirtualSystem (Machine CIMI), определенной в пакете OVF. Кроме того, на базе DiskSection может быть создан новый ресурс CIMI MachineImage, если для содержимого виртуального диска определен атрибут ovf:fileRef.

Сети Networks CIMI созданы в соответствии с NetworkSection дескриптора OVF, наряду с элементами Connection в VirtualHardwareSection, которые обращаются к данным сетям, имеющим наименование.

Приложение В
(справочное)

XML-схема

Расположение схемы XML для сериализации XML модели CIMI—gost.e-ecolog.ru DSP8009_1.0.xsd

Предоставленная схема не предназначена для отражения любых ограничений моделирования и любых требований, определенных в модели. Данная схема разработана преимущественно для применения в сериализованных материалах в запросах Потребителя, а также в ответах Поставщика и предназначена для обеспечения их предварительной неисчерпывающей синтаксической проверки. В частности, обновления данной спецификации в будущем могут привести к смешиванию новых элементов XML в Ресурсах, используя для Ресурсов текущее пространство имен CIMI. Предоставленная схема является отправной точкой для тех, кто может счесть ее полезной, и, возможно, она должна быть модифицирована, исходя из потребностей конкретного приложения.

Приложение С
(справочное)

История изменений

Версия	Дата	Описание
1.0.0	2012-08-28	—
1.0.1	2012-09-12	Опечатки
1.1.0	2013-10-22	Опубликовано как стандарт DMTF

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов национальным стандартам

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
IEC 80000-13:2008	—	*
ISO 8601:2004	—	*
ISO/IEC 14977:1996	—	*
* Соответствующий национальный стандарт отсутствует.		

УДК 004.057.8 ОКС 35.100.05

Ключевые слова: служба облачных вычислений, потребитель службы облачных вычислений, инфраструктура как услуга, Набор, Ресурс

Редактор *М.Ю. Сухинина*
Технический редактор *В.Н. Прусакова*
Корректор *С.И. Фирсова*
Компьютерная верстка *Е.Е. Кругова*

Сдано в набор 28.02.2017. Подписано в печать 20.04.2017. Формат 60 ×84 1/8. Гарнитура Ариал.
Усл. печ. л. 20,00. Уч.-изд. л. 19,46. Тираж 29 экз. Зак. 505.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Издано и отпечатано во ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.
gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru