ФЕДЕРАЛЬНОЕ АГЕНТСТВО
ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО/МЭК 29362— 2013

Информационная технология

Функциональная совместимость веб-служб

Профиль вложений WS-1 Версия 1.0

ISO/IEC 29362:2008
Information technology – Web Services Interoperability — WS-I Attachments Profile Version 1.0

(IDT)

Издание официальное

Москва
Стандартинформ
2014

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Обществом с ограниченной ответственностью «Информационный аналитический вычислительный центр» (ООО «ИАВЦ») на основе собственного аутентичного перевода на русский язык 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 8 ноября 2013 г. № 1540-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/МЭК 29362:2008 «Информационная технология. Функциональная совместимость веб-служб. Профиль вложений WS-I. Версия 1.0» (ISO/IEC 29362:2008 «Information technology – Web Services Interoperability — WS-I Attachments Profile Version 1.0»)

Наименование настоящего стандарта изменено относительно наименования указанного международного стандарта для приведения в соответствие с ГОСТ Р 1.5 (пункт 3.5).

5 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в ГОСТ Р 1.0—2012 (раздел 8). 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 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.ru)

© Стандартинформ, 2014

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

Содержание

1 Область применения и введение 1
2 Соответствие профилю 2
3 Упаковка вложения 4
4 Описание вложений 9
Приложение A. Ссылки на спецификации 22
Приложение B: Возможные расширения 23
Приложение C. Нормативные ссылки 24
Приложение D: Определения терминов 25
Приложение E: Благодарности 26

Введение

Настоящий международный стандарт определяет профиль вложений WS-I 1.0 (далее «Профиль»), состоящий из набора открытых спецификаций Веб-служб вместе с разъяснениями и усилениями тех спецификаций, которые предназначены для обеспечения функциональной совместимости. Этот профиль дополняет основной профиль WS-I 1.1, обеспечивая поддержку передачи сообщений SOAP (*Simple Object Access Protocol* — простой протокол доступа к объектам) с вложениями, соответствующими спецификации «Сообщения SOAP с вложениями».

ИСО/МЭК 29362 был подготовлен Организацией функциональной совместимости Веб-служб – Web Services Interoperability Organization (WS-I) и был принят Объединенным Техническим Комитетом СТК 1 ИСО/МЭК «Информационные технологии» в соответствии с процедурой для опубликованных технических спецификаций параллельно с его одобрением национальными органами ИСО и МЭК.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационная технология Функциональная совместимость веб - служб Профиль вложений WS-1. Версия 1.0

Information technology – Web Services Interoperability — WS-I Attachments Profile Version 1.0

Дата введения — 2015—01—01

1 Область применения и введение

1.1 Область применения

Настоящий международный стандарт определяет профиль вложений WS-I 1.0 (далее «профиль»), состоящий из набора открытых спецификаций Веб-служб вместе с разъяснениями и усилениями тех спецификаций, которые предназначены для обеспечения функциональной совместимости. Этот профиль дополняет основной профиль WS-I 1.1, обеспечивая поддержку передачи сообщений SOAP (*Simple Object Access Protocol* — простой протокол доступа к объектам) с вложениями, соответствующими спецификации «Сообщения SOAP с вложениями».

Сообщения SOAP с вложениями (SwA) определяют составную/связанную структуру MIME (*Multipurpose Internet Mail Extensions* — Многоцелевое расширение возможностей почты в сети Интернет) для упаковки вложений с сообщениями SOAP. Данный профиль дополняет основной профиль WS-I 1.1, добавляя поддержку передачи вложений на базе SwA, обеспечивающих функциональную совместимость с сообщениями SOAP.

В разделе 1 приводится описание профиля и объясняется его отношения с другими профилями.

Раздел 2 «Соответствие профилю» объясняет, что значит быть совместимым с профилем.

Каждый последующий раздел рассматривает компонент профиля и состоит из двух частей: общего описания, в котором детализируются спецификации компонентов и вопросы их расширяемости, и последующих подразделов, в которых рассматриваются отдельные части спецификаций компонентов.

1.2 Отношения с другими профилями

Этот профиль добавляет поддержку SOAP с вложениями и привязку MIME (MIME binding) и предназначен для использования в комбинации с основным профилем 1.1.

1.3 Условные обозначения

Ключевые слова «ДОЛЖЕН» (MUST), «НЕ ДОЛЖЕН» (MUST NOT), «ТРЕБУЕМЫЙ» (REQUIRED), «БУДЕТ» (SHALL), «НЕ БУДЕТ» (SHALL NOT), «СЛЕДУЕТ» (SHOULD), «НЕ СЛЕДУЕТ» (SHOULD NOT), «РЕКОМЕНДУЕМЫЙ» (RECOMMENDED), «МОЖЕТ» (MAY) и «ДОПОЛНИТЕЛЬНЫЙ» (OPTIONAL) в данном документе должны интерпретироваться в соответствии с RFC2119

Нормативные положения для требований профиля (те, которые влияют на соответствие, как описано в «Требованиях соответствия»), представлены следующим способом:

Rnnnn *Текст положения*

где nnnn - уникальное среди требований профиля число, в результате чего образуется уникальный идентификатор требования.

Таким образом, идентификаторы требований могут рассматриваться, как квалифицированное пространство имен, совместимое с QNames из Пространства имен в XML. Если идентификатор требования не содержит никакого явного префикса пространства имен (например, «R9999» в отличие от «bp10:R9999»), то он должен рассматриваться как принадлежащий пространству имен, определяемому соответствующим универсальным идентификатором ресурса (URI) раздела документа, в котором он присутствует. Если эти условия соблюдены, то префикс следует интерпретировать согласно фактическим отображениям пространства имен, как показано далее.

Некоторые требования используются для пояснения соответствующих спецификаций, но не накладывают дополнительных ограничений на реализацию. Для удобства такие пояснения обозначаются следующим способом: C

Отдельные требования являются результатом постоянно продолжающейся работы по стандартизации спецификаций, на которые они ссылаются. Для удобства такие положения, которые

могут претерпеть изменения, обозначаются как: xxxx, где xxxx является идентификатором спецификации (например, «WSDL20» для версии 2.0 WSDL). Необходимо отметить, что поскольку эта работа не была завершена на момент публикации данного документа, спецификация, к которой относится требование, может измениться, и данная информация приводится только для удобства разработчиков.

Возможные расширения в базовых спецификациях (см. «Область соответствия») представлены следующим образом:

Ennnn *Название возможного расширения – Описание,*

где nnnn заменятся числом, которое уникально среди возможных расширений в профиле. Аналогично требованиям, идентификаторы расширений можно рассматривать как квалифицированные имена в пространстве имен.

В данной спецификации используется несколько префиксов пространств имен, которым соответствуют универсальные идентификаторы ресурсов URI, перечисленных далее. Необходимо отметить, что выбор любого префикса пространства имен произволен и не является семантически существенным.

- **soap** - gost.e-ecolog.ru
- **xsi** - http://www.w3.org/2001/XMLSchema-instance»
- **xsd** - gost.e-ecolog.ru
- **soapenc** - gost.e-ecolog.ru
- **wsdl** - gost.e-ecolog.ru
- **soapbind** - gost.e-ecolog.ru
- **mime** - gost.e-ecolog.ru
- **uddi** - urn:uddi-org:api_v2»
- **wsi** - gost.e-ecolog.ru
- **ref** – gost.e-ecolog.ru

1.4 Идентификация профиля и управление версиями

Данный документ идентифицирован наименованием (в данном случае, «Профиль вложений») и номером версии (здесь, 1.0). В совокупности они идентифицируют конкретный *экземпляр профиля*

Номера версий составлены из основной и дополнительной частей в форме «основной.дополнительный». Они могут использоваться, чтобы определить приоритет экземпляра профиля; больший номер версии (с учетом и основного, и дополнительного компонентов) указывает, что данный экземпляр новее, и поэтому заменяет более ранние экземпляры.

Для экземпляров профилей с одинаковыми именами (например, «Пример Профиля 1.1» и «Пример Профиля 5.0») проблемы функциональной совместимости рассматриваются в одной и той же общей области действия, хотя при некоторых разработках может потребоваться, чтобы точная область действия профиля была различна для различных экземпляров.

Данная информация может также быть использована, чтобы определить, являются ли два экземпляра профиля обратно совместимыми, т.е. можно ли считать, что соответствие более раннему экземпляру профиля подразумевает соответствие более позднему. Экземпляры профиля с одним тем же именем и номером основной версии (например, «Пример Профиля 1.0» и «Пример Профиля 1.1»), МОГУТ считаться совместимым. Совместимость в другом направлении не подразумевается, т.е. нельзя предполагать, что соответствие более позднему экземпляру профиля подразумевает соответствие более раннему.

2 Соответствие профилю

Соответствие профилю определено точным выполнением набора требований, определенных в рамках области применения профиля для определенной цели. Настоящий раздел объясняет эти термины и описывает, как определяется и используется соответствие.

2.1 Требования соответствия

Требования устанавливают критерии соответствия профилю. Они обычно ссылаются на существующую спецификацию и добавляют усовершенствования, усиления, интерпретации и разъяснения к ней, чтобы улучшить функциональную совместимость. Все требования в профиле считаются нормативными. Требования в спецификациях, на которые профиль ссылается в области применения (см. «Область соответствия») также должны рассматриваться как нормативные. В случае, если требования в профиле и спецификациях, на которые он ссылаются, противоречат друг другу, требования профиля имеют приоритет с целью обеспечения соответствия профиля.

Уровни обязательности требования, обозначенные в соответствии с RFC2119 (ДОЛЖЕН, МОЖЕТ, СЛЕДУЕТ и т.п.), указывают на природу требования и его воздействие на соответствие. Для удобства каждое требование имеет индивидуальный идентификатор (например, R9999).

Пример – R9999 Виджетам (WIDGETs) СЛЕДУЕТ (SHOLUD) быть круглой формы.

Это требование, идентифицированное как «R9999», применяется к целевому ВИДЖЕТУ (WIDGET) (см. далее), и предъявляет условное требование к совокупности виджетов, т.е. хотя это требование в большинстве случаев должно быть удовлетворено, чтобы реализация соответствовала профилю, имеются отдельные ситуации, когда могут быть уважительные причины нарушения требования. Такие причины объясняются непосредственно в требовании или в сопровождающем его тексте.

Каждое положение требований содержит одно и только одно ключевое слово уровня обязательности требования (например, «ДОЛЖЕН» (MUST)), и одно ключевое слово цели соответствия (например, «СООБЩЕНИЕ» (MESSAGE)). Ключевое слово цели соответствия показано жирным шрифтом (например, **«СООБЩЕНИЕ» (MESSAGE)).** Другие цели соответствия, показанные не жирным шрифтом, используются строго для их определения, а НЕ для цели соответствия. Для объяснения требования или группы требований может быть включен дополнительный текст (например, обоснование и примеры); однако, текст, не входящий непосредственно в положения требований, не должен рассматриваться при определении соответствия.

Определения терминов профиля считаются надежными для целей определения соответствия.

Ни одно из требований профиля, независимо от уровня его обязательности, не должно быть истолковано как ограничивающее возможность реализации соответствующих контрмер безопасности в ответ на реальные или предполагаемые угрозы (например, на атаку отказа в обслуживании).

2.2 Цели соответствия

Цели соответствия определяют, к каким артефактам применяются требования. Примерами целей являются: сообщение SOAP (*Simple Object Access Protocol* — простой протокол доступа к объектам), описание WSDL (*Web Services Description Language* — язык описания Веб-служб и доступа к ним), данные реестра UDDI (*Universal Description Discovery & Integration* — инструмент для расположения описаний Веб-служб) или стороны, например, процессор SOAP, конечный пользователь.

Это позволяет определить соответствие в различных контекстах для гарантии однозначной интерпретации применимости требований и обеспечения проверки артефактов (например, сообщений SOAP и описаний WSDL) и функциональных возможностей различных сторон на соответствие Веб-службе (например, конкретным клиентам и службам).

Для обеспечения однозначности и упрощения тестирования целями соответствия в требованиях являются, где это возможно, физические артефакты.

В профиле используются следующие цели соответствия:

• **СООБЩЕНИЕ** (MESSAGE) – элементы протокола, посредством которых осуществляется передача КОНВЕРТА (например, сообщение SOAP/HTTP) (см. ИСО/МЭК 29361);

• **КОНВЕРТ** (ENVELOPE) – преобразование в сериализованную форму элемента soap:Envelope и его содержимого (см. ИСО/МЭК 29361);

• **ОПИСАНИЕ** (DESCRIPTION) – описания типов, сообщений, интерфейсов и их конкретных протоколов, привязки форматов данных, и сетевых точек доступа, связанных с Веб-сервисами (например, описания WSDL) (см. ИСО/МЭК 29361);

• **РЕАЛИЗАЦИЯ** (INSTANCE) – программное обеспечение, которое осуществляет выполнение связывания *wsdl:port* или *uddi:bindingTemplate* (см. ИСО/МЭК 29361);

• **ПОЛЬЗОВАТЕЛЬ** (CONSUMER) – программное обеспечение, которое обращается к РЕАЛИЗАЦИИ (см. ИСО/МЭК 29361);

• **ОТПРАВИТЕЛЬ** (SENDER) – программное обеспечение, которое генерирует сообщение в соответствии со связанными с ним протоколом или протоколами (см. ИСО/МЭК 29361);

• **ПОЛУЧАТЕЛЬ** (*RECEIVER*) – программное обеспечение, которое использует сообщение в соответствии со связанными с ним протоколом или протоколами (например, процессоры SOAP) (см. ИСО/МЭК 29361).

2.3 Область соответствия

Область применения профиля ограничивается технологиями, для которых он предназначен. Другими словами профиль всего лишь пытается улучшить функциональную совместимость в пределах своей собственной области применения. Обычно область применения профиля ограничена спецификациями, на которые он ссылается.

Область применения профиля уточняется далее возможными расширениями. Технические спецификации зачастую обеспечивают механизмы расширения и неопределенные или открытые параметры конфигурации. В том случае, если в профиле определено возможное расширение, то

такой механизм или параметр не входит в область применения профиля, и его использование или неиспользование выходит за рамки требований соответствия.

Следует обратить внимание на то, что профиль все же может выдвинуть требования по использованию расширений. Кроме того, конкретные случаи использования расширения могут быть еще более ограничены другими профилями для улучшения функциональной совместимости при использовании сочетания профилей.

Поскольку использование возможных расширений может нарушить функциональную совместимость, то их использование должно быть согласовано или задокументировано потребителями Веб-служб в некоторой форме, например, в виде дополнительного соглашения.

Область применения профиля определена техническими условиями, перечисленными в Приложении А, и уточняется возможными расширениями, указанными в Приложении В.

2.4 Декларация о соответствии

Декларации о соответствии профилю могут быть сделаны с использованием механизмов, описанных в спецификации «Механизмы вложения деклараций о соответствии», в случае, если были удовлетворены применимые требования профиля, связанные со следующими целями:

• **WSDL 1.1 Механизм вложения деклараций для реализаций Веб-служб** (Claim Attachment Mechanism for Web Services Instances) –

ПОЛУЧАТЕЛЬ ЭКЗЕМПЛЯРА ОПИСАНИЯ СООБЩЕНИЯ;

• **WSDL 1.1 Механизм вложения деклараций для конструкций описания** (Claim Attachment Mechanism for Description Constructs) –

ОПИСАНИЕ;

• **UDDI Механизм вложения деклараций для конкретных Веб-служб** (Claim Attachment Mechanism for Web Services Instances –

ПОЛУЧАТЕЛЬ ЭКЗЕМПЛЯРА ОПИСАНИЯ СООБЩЕНИЯ.

URI декларации о соответствии для этого профиля –«gost.e-ecolog.ru

3 Упаковка вложения

Данный раздел профиля включает следующие ссылки на спецификации и определяет возможные в них расширения:

• Сообщения SOAP с вложениями (SOAP Messages with Attachments)__ Возможные расширения:

o E0001 – части MIME – Сообщения SOAP с вложениями не накладывают никаких ограничений на тип некорневой части составных / связанных сообщений.

• Расширяемый язык разметки (XML) 1.0 (Вторая Редакция) (Extensible Markup Language (XML) 1.0 (Second Edition)).

• Пространства имен в XML 1.0 (Namespaces in XML 1.0).

• RFC2557 Инкапсуляция составных документов MIME, таких как HTML (MHTML) (RFC2557 MIME Encapsulation of Aggregate Documents, such as HTML (MHTML)).

• RFC2045 Многоцелевое расширение возможностей почты в сети Интернет (MIME). Часть 1: Формат интернет-сообщений (RFC2045 Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies).

• RFC2046 Многоцелевое расширение возможностей почты в сети Интернет (MIME). Часть 2: Типы медиа (RFC2046 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types).

• RFC2392 Единые указатели ресурсов идентификатора контента и идентификатора сообщения (RFC2392 Content-ID and Message-ID Uniform Resource Locators).

Сообщения SOAP с вложениями определяют составную/связанную (multipart/related) структуру MIME, предназначенную для формирования конверта SOAP с вложениями. Профиль требует использования этой структуры и ставит приведенные далее ограничения на ее использование:

3.1 Корневая часть

R2931 Собственно тело корневой части составного/связанного **СООБЩЕНИЯ** ДОЛЖНО быть *soap:Envelope*.

R2945 Значение поля Content-Type HTTP-заголовка в **СООБЩЕНИИ** ДОЛЖНО быть либо «*multipart/related*», либо «*text/xml*». С

R2932 Если в значении Content-Type HTTP-заголовка **СООБЩЕНИЯ** указан тип медиа (*media-type*) «*multipart/related*», то в значении Content-Type HTTP-заголовка ДОЛЖЕН присутствовать параметр type со значением «*text/xml*». С

Любая часть MIME может содержать *soap:Envelope*, однако только собственно тело корневой части пакета MIME обрабатывается как основной конверт SOAP. Некорневые части рассматриваются как вложения.

Пример:
ПРАВИЛЬНО
В показанном далее сообщении в значении поля Content-Type HTTP-заголовка указан тип медиа (media-type) «multipart/related», и параметр type имеет значение «text/xml».

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
Content-Description: This is the optional message description.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru >
<?xml version='1.0' ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV= «gost.e-ecolog.ru >
...
</SOAP-ENV:Envelope>
--MIME_boundary
...
--MIME_boundary--
```

3.2 Кодирование корневой части

R2915 Собственно тело корневой части составного/связанного **СООБЩЕНИЯ** ДОЛЖНО быть преобразовано в сериализованную форму с использованием кодировки символов UTF-8 или UTF-16.

R2916 Для некорневых частей составного/связанного **СООБЩЕНИЯ** МОЖЕТ быть использована любая кодировка символов.

3.3 Тип медиа(*media-type*) сообщения

R2925 Если в описании WSDL упомянута хотя бы одна некорневая часть MIME, то в значении поля Content-Type HTTP-заголовка соответствующего **СООБЩЕНИЯ** тип медиа (*media-type*) ДОЛЖЕН быть «*multipart/related*».

3.4 Сообщения без вложений

Если получатель ожидает ноль или более вложений в сообщении, то отправитель этого сообщения может использовать для сообщения без каких-либо вложений тип медиа text/xml.

R2917 **СООБЩЕНИЕ**, содержащее ноль частей вложения, ДОЛЖНО быть отправлено, с использованием типа контента или «text/xml», если использовалось HTTP-связывание SOAP, или «*multipart/related*» в случае, если описание WSDL для сообщения определяет элемент *mime:multipartRelated* в соответствующем элементе *wsdl:input* или *wsdl:output* в его привязке *wsdl:binding*.

R2902 **ОТПРАВИТЕЛЬ** НЕ ДОЛЖЕН посылать сообщения с использованием SOAP с вложениями, если соответствующий элемент *wsdl:input* или *wsdl:output* в *wsdl:binding* не определяет привязку WSDL MIME.

Это может произойти только в случае, если описание WSDL определяет *mime:multipartRelated*, у которого есть только один дочерний элемент *mime:part*, содержащий *soapbind:body*.

Пример:
ПРАВИЛЬНО:
Результатом следующего описания WSDL:

```
<wsdl:definitions xmlns:xsd="gost.e-ecolog.ru )1/XMLSchema"
xmlns:soapbind="gost.e-ecolog.ru "
xmlns:wsdl="gost.e-ecolog.ru
xmlns:mime="gost.e-ecolog.ru
targetNamespace="http://example.com/mimewsdl"
xmlns:tns="gost.e-ecolog.ru >
...
 <wsdl:binding name="aBinding" type="tns:aPortType">
   <soapbind:binding style="rpc"
                     transport="gost.e-ecolog.ru >
   <wsdl:operation name="anOperation">
       <soap:operation soapAction="gost.e-ecolog.ru >
       <wsdl:input>
           <mime:multipartRelated>
               <mime:part>
```

```
                    <soapbind:body use="literal"
                      namespace="gost.e-ecolog.ru                    >
                  </mime:part>
                </mime:multipartRelated>
              </wsdl:input>
              <wsdl:output>
                <soapbind:body use="literal"
                      namespace="http://example.com/mimetypes"/>
              </wsdl:output>
            </wsdl:operation>
          </wsdl:binding>
        </wsdl:definitions>
```

может быть входящее сообщение, которое использует HTTP-связывание SOAP следующим образом:

```
<?xml version='1.0' ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="gost.e-ecolog.ru                                    >
   <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                    >
      <types:anOperation>
...
      </types:anOperation>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

но не должно быть исходящее сообщение, которое использует следующее связывание MIME:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
              start="<gost.e-ecolog.ru                >"
Content-Description: This is the optional message description.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                >

<?xml version='1.0' ?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="gost.e-ecolog.ru                                    >
   <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                    >
      <types:anOperationResponse>
...
      </types:anOpertionResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
--MIME_boundary--
```

3.5 Разрешение ссылок вложений

Приложения, использующие Веб-службы, могут использовать URI различными способами, включая и средства извлечения контента из сети. Хотя вложения и могут быть использованы для обеспечения контента, который идентифицирован соответствующим URI, не требуется, чтобы реализации оказывали предпочтение присоединенному контенту, использовали бы его исключительно или использовали его вообще. Таким же образом приложения могут игнорировать указанную функцию разрешения ссылок URI (например, загрузку контента из сети) и использовать только присоединенный контент. При использовании схемы идентификации контента (Content-ID, CID) в URI применяются синтаксис и правила, определенные в RFC 2392

R2918 **ПОЛУЧАТЕЛЬ** МОЖЕТ игнорировать ссылку URI на вложение в конверте (envelope)

3.6 Пересылка дополнительных конвертов SOAP *(SOAP Envelopes)*

Профиль не накладывает ограничений на содержимое отдельных частей с вложениями. Как вложения могут передаваться дополнительные документы XML, содержащие конверт SOAP (*soap:Envelope*), однако только корневая часть сообщения MIME должна рассматриваться, как основной конверт SOAP(soap:Envelope) в пакете MIME.

R2919 СООБЩЕНИЕ МОЖЕТ содержать конверты SOAP(*soap:Envelope*), пересылаемые в качестве вложений в тех частях сообщения, которые не являются корневой частью.

3.7 Сообщения об ошибках с вложениями

R2920 **РЕАЛИЗАЦИЯ** (INSTANCE) МОЖЕТ послать сообщение об ошибке с вложением тогда и только тогда, если элемент *wsdl:output* описан с использованием привязки *WSDL MIME*.

3.8 Пространство значений поля заголовка *Content-Id*

Определение: кодирование идентификатора контента части (content-id part encoding)

«Кодирование идентификатора контента части» представляет собой операцию конкатенации величин:

• Значения атрибута name элемента *wsdl:part*, на который ссылается *mime:content*, и в котором символы, недопустимые в заголовках Content-Id (не ASCII символы, представленные кодами большими 0x7F), представлены следующим образом:

o Каждый недопустимый символ преобразован в один или более байтов в соответствии с UTF-8.

o Все байты, соответствующие недопустимому символу, экранируются согласно правилам экранирования в URI (т.е. преобразованы в %HH, где HH – шестнадцатеричная нотация значения байта),

o Исходный символ заменяется полученной последовательностью символов .

• Символ '=' (0x3D).

• Глобально уникальное значение, такое как UUID.

• Символ '@' (0x40).

• Действительное имя домена, от имени которого создается сообщение.

R2933 Если описание связывает часть wsdl:message с элементом mime:content, то соответствующее значение поля Content-Id части MIME в **СООБЩЕНИИ** ДОЛЖНО удовлетворять правилам кодирования части Content-Id.

Пример:

ПРАВИЛЬНО:

Во фрагменте WSDL далее, имя части, привязанной к mime:content, является значением, добавленным к значению Content-Id.

```
<wsdl:message name="fooMsg">
<wsdl:part name="body" type="ns1:Claim"/>
<wsdl:part name="fooPart" type="xs:base64binary"/>
</wsdl:message>
...
<wsdl:binding
...
  <mime:multipartRelated>
    <mime:part>
      <soapbind:body parts="body" use="literal"/>
    </mime:part>
    <mime:part>
      <mime:content part="fooPart" type="application/octet-stream"/>
    </mime:part>
  </mime:multipartRelated>
...
</wsdl:binding>
```

Вот фрагмент составного пакета, содержащего двоичный поток *fooPart*, показывающий, как атрибут «имени» *wsdl:part* включен в значение Content-Id.

```
...
--MIME_boundary
Content-Type: application/octet-stream
Content-Transfer-Encoding: 8bit
```

Content-ID: <fooPart=somereallybi gost.e-ecolog.ru >

...

3.9 Порядок частей MIME

Возможно, посредники могут переупорядочить части составного/связанного сообщения. Следовательно, семантика не должна ни требовать определенного порядка частей сообщения, ни подразумевать такое упорядочивание

R2921 **ПОЛУЧАТЕЛЬ** НЕ ДОЛЖЕН делать какие-либо выводы по семантике из порядка некорневых частей MIME в сообщении.

R2929 **СООБЩЕНИЯ** МОГУТ содержать части MIME в любом порядке при условии сохранения идентичности корневой части.

Получатель не должен предполагать, что порядок элементов mime:part, определенных в описании WSDL, является тем же самым, что и порядок частей MIME в сообщении. Порядок частей MIME, определенных в описании WSDL, нужно считать независимым от порядка частей MIME в сообщении.

3.10 *Расположение корневой части*

Если присутствует параметр start, то значение параметра start является идентификатором контента корневой части сообщения. При отсутствии параметра start корневой частью считается первая часть тела пакета, как определено разделом 3.2 RFC 2387.

R2922 Если значения поля Content-Type HTTP-заголовка сообщения не содержат параметра start, то **ПОЛУЧАТЕЛЬ** ДОЛЖЕН обработать первую часть тела пакета MIME как корневую часть. С

Пример:

ПРАВИЛЬНО:

В сообщении далее первая часть MIME (у которой есть заголовок идентификатора контента "< gost.e-ecolog.ru ***>") является корневой частью.***

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
Content-Description: This is the optional message description.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru >
<?xml version='1.0' ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="gost.e-ecolog.ru >
  <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru ">
    <types:SendClaim>
      <ClaimDetail>.....</ClaimDetail>
      <photo>
        <href>cid:claimphoto@example.com</href>
      </photo>
    </types:SendClaim>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
--MIME_boundary
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
Content-ID: <gost.e-ecolog.ru >
...цифровая фотография в двоичном представлении...
--MIME_boundary—
```

3.11 Параметр Content-Transfer-Encoding

Параметр Content-Transfer-Encoding обеспечивает передачу сообщений через транспортные системы, которые не поддерживают двоичный контент. Например, некоторые системы электронной почты могут передавать только символьные сообщения. Поскольку сообщения Веб-служб могут происходить от или быть предназначены для таких систем, профиль допускает использование этого механизма.

Если в части составного сообщения MIME нет параметра Content-Transfer-Encoding, то тело этой части должно соответствовать 7-битной кодировке ascii, как определено в RFC 2045.

R2934 Поле Content-Transfer-Encoding части составного/связанного **СООБЩЕНИЯ** ДОЛЖНО иметь значение «7bit», «8bit», «binary», «quoted-printable» или «base64».

R2935 Кодирование тела части составного/связанного **СООБЩЕНИЯ** ДОЛЖНО соответствовать кодированию, определенному значением поля Content-Transfer-Encoding, как определено в RFC2045.c

С целью повышения функциональной совместимости профиль ограничивает допустимые значения только широко распространенными.

3.12 Строка разграничения MIME

Как показывает практика, определенные реализации создают сообщения, в которых строке, разделяющей части MIME (MIME encapsulation boundary string),не предшествуют символы CRLF (перевод строки и возврат каретки). Этот факт создает проблемы для реализаций, которые правильно ожидают, что строке, разделяющей части MIME предшествуют символы CRLF.

R2936 Всем строкам, разделяющим части MIME в **СООБЩЕНИИ,** должны предшествовать символы ascii CR (13) и LF (10) строго в такой последовательности. С

В разделе 5.5.1 RFC2046, ясно изложены требования того, чтобы всем строкам, разделяющим части MIME, предшествовали символы CRLF (перевод строки и возврат каретки).

4 Описание вложений

Данный раздел профиля содержит ссылку на следующую спецификацию:

• WSDL 1.1, Раздел 5.0

Раздел 5 WSDL 1.1 определяет привязку MIME. Профиль разрешает использование привязки MIME WSDL, но ограничивает ее сообщениями SOAP с протоколом вложений. Профиль накладывает ограничения на его использование, которые перечислены далее.

4.1 Использование расширения привязки MIME

Возможны случаи, когда отправитель может быть способен отправлять сообщения, используя SOAP с вложениями, однако не способен получать и обрабатывать такие сообщения.

R2901 В **ОПИСАНИИ,** в каждом из элементов *wsdl:input* или *wsdl:output* соответствующего *wsdl:binding,* ДОЛЖНА использоваться или привязка WSDL MIME, как описано в разделе 5 WSDL 1.1, или привязка WSDL SOAP, как описано в разделе 3 WSDL 1.

4.2 Несвязанное содержимое элемента *portType*

WSDL 1.1 не определяет явно, допустимо ли для *wsdl:binding* оставить неопределенной привязку для частей контента, определенных *wsdl:portType.*

R2941 Привязка *wsdl:binding* в **ОПИСАНИИ** ДОЛЖНА связывать каждую часть *wsdl:part* сообщения *wsdl:message,* на которые ссылается, в *wsdl:portType,*с одним из: *soapbind:body, soapbind:header, soapbind:fault, soapbind:headerfault* или *mime:content.*

portType определяет абстрактный интерфейс (contract) с именованным набором операций и соответствующих абстрактных сообщений. Хотя это и не запрещено, ожидается, что каждая часть абстрактного входящего, исходящего сообщения или сообщения об ошибке, специфицированого в *PortType,* связана с *soapbind:body* или *soapbind:header* (и прочими) или с *mime:content,* соответствующим образом с использованием привязки MIME, как это определено в разделе 5 WSDL 1.1. Несвязанные части *wsdl:parts* должны игнорироваться потребителем.

4.3 Ссылки на части сообщения

Часть сообщения в WSDL может быть связана с определенной частью MIME с использованием *mime:content.* В отличие от заголовка *soapbind:header*, который может сослаться на части, содержавшиеся в сообщении, которое не является частью соглашения, определенного в *wsdl:porttype,* контент *mime:content* не должен сослаться на часть *wsdl:part*, не определенную в сообщении *wsdl:message*, ссылка на которое имеется в *wsdl:operation.* Кроме того, части сообщения в WSDL рассматриваются как единый неделимый модуль. Компоненты части сообщения, которые имеет сложный контент, не могут быть выборочно связаны с определенной частью MIME.

R2903 Элемент *mime:content* в **ОПИСАНИИ** НЕ ДОЛЖЕН сослаться на часть *wsdl:part*, которая не присутствует в соответствующем *wsdl:input* или *wsdl:output* соответствующего *wsdl:operation* соответствующего *wsdl:portType.*

R2904 Элемент *mime:content* в **ОПИСАНИИ** НЕ ДОЛЖЕН быть связан с субкомпонентом элемента или типа, на которые ссылается *wsdl:part.*

R2946 Элемент *mime:content* в **ОПИСАНИИ** *ДОЛЖЕН* включать в себя атрибут части *part.*

Пример:

НЕПРАВИЛЬНО:

```
<definitions xmlns="gost.e-ecolog.ru                                    >
 <types ...>
   <schema xmlns="gost.e-ecolog.ru
             targetNamespace="http://example.org/foo"
             xmlns:ns="gost.e-ecolog.ru
    <element name='foo'>
     <complextContent>
       <sequence>
        <element ref='bar1'/>
        <element ref='bar2'/>
       </sequence>
     </complexContent>
    </element>
   </schema>
 </types>
...
<message name='aMsg'>
  <part name='apart' element='ns:foo' />
  <part name="body" element="ns:bar"/>
</message>
<portType>
  <operation>
   <input>
     <part name="apart">
   </input>
...
  </operation>
</portType>
<binding>
  <operation>
   <input>
     <mime:multipartRelated>
        <mime:part>
            <soapbind:body part="body" use="literal"/>
        </mime:part>
        <mime:part>
            <mime:content part="ns:bar1"/>
        </mime:part>
     </mime:multipartRelated>
   </input>
...
  </operation>
</binding>
</definitions>
```

4.4 Ссылка на вложения в конверте SOAP

Одним из преимуществ использования вложений является возможность включать данные в отдельную часть MIME и поместить ссылку на нее в конверт SOAP, который размещается в корневой части того же самого пакета MIME.

Данный профиль определяет тип схемы *ref:swaRef*, которая может использоваться в описании WSDL, чтобы определить часть сообщения. Если часть сообщения описана с использованием типа *ref:swaRef* в конкретном экземпляре документа, то URI указывает на вложение в том же самом пакете MIME. Этот тип предназначен для разработчиков приложений/инструментов/платформ, поскольку является интероперабильным способом размещения ссылок на вложения в описаниях. Однако допускается и использование других механизмов, которые не делают описание несовместимым.

XML-схема для типа, используемого для ссылок на вложения из конверта SOAP:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema targetNamespace="gost.e-ecolog.ru
```

```
    xmlns:xsd="gost.e-ecolog.ru                                    ">
 <xsd:simpleType name="swaRef">
  <xsd:restriction base="xsd:anyURI" />
 </xsd:simpleType>
</xsd:schema>
```

Для удобства схема данного типа опубликована WS-I по адресу: gost.e-ecolog.ru

Необходимо отметить, что в описании WSDL1.1 нет средств, чтобы сопоставить ссылку вложения (определенную с использованием *swaRef*) с вложением (определенным с использованием части *wsdl:part*, связанной с mime:content). Для эффективной работы в профиле рекомендуется, чтобы при использовании *ref:swaRef* соответствующее вложение не описывалось, и наоборот.

R2940 Элемент *wsdl:part*, определенный с помощью схемы типа *ref:swaRef* в **ОПИСАНИИ**, ДОЛЖЕН БЫТЬ связан в привязке MIME только с *soapbind:body* или *soapbind:header*.

R2928 Ссылка URI в **КОНВЕРТЕ**, определенная с использованием типа схемы *ref:swaRef*, ДОЛЖНА указывать на часть MIME, размещенную в том же сообщении, что и конверт.

Тип swaRef может использоваться, чтобы представить ссылку на вложение в виде либо элемента (как показано в примере далее), либо в виде атрибута. В данном случае предпочтения тому или иному подходу нет.

Пример:

ПРАВИЛЬНО:

Описание WSDL для связывания rpc/literal:

```
<?xml version="1.0"?>
<wsdl:definitions xmlns:types="gost.e-ecolog.ru
    xmlns:ref="gost.e-ecolog.ru
    xmlns:xsd="gost.e-ecolog.ru                                    "
    xmlns:soapbind="gost.e-ecolog.ru
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:mime="gost.e-ecolog.ru
    targetNamespace="gost.e-ecolog.ru
    xmlns:tns="gost.e-ecolog.ru                                    ">
 <wsdl:types>
      <xsd:schema targetNamespace="gost.e-ecolog.ru
                  xmlns:xsd="gost.e-ecolog.ru                                    >
          <xsd:import namespace="gost.e-ecolog.ru                                    >
          <xsd:complexType name="ClaimDetailType">
              <xsd:sequence>
                  <xsd:element name="Name" type="xsd:string"/>
                  <xsd:element name="ClaimForm" type="ref:swaRef"/>
              </xsd:sequence>
          </xsd:complexType>
      </xsd:schema>
 </wsdl:types>
 <wsdl:message name="ClaimIn">
      <wsdl:part name="ClaimDetail" type="types:ClaimDetailType"/>
      <wsdl:part name="ClaimPhoto" type="xsd:base64Binary"/>
 </wsdl:message>
 <wsdl:message name="ClaimOut">
      <wsdl:part name="ClaimRefNo" type="xsd:string"/>
 </wsdl:message>

 <wsdl:portType name="ClaimPortType">
      <wsdl:operation name="SendClaim">
          <wsdl:input message="tns:ClaimIn"/>
          <wsdl:output message="tns:ClaimOut"/>
       </wsdl:operation>
 </wsdl:portType>
<wsdl:binding name="ClaimBinding" type="tns:ClaimPortType">
      <soapbind:binding style="rpc"
```

```
                    transport="gost.e-ecolog.ru                    >
        <wsdl:operation name="SendClaim">
            <soapbind:operation soapAction="gost.e-ecolog.ru                    >
            <wsdl:input>
                <mime:multipartRelated>
                    <mime:part>
                        <soapbind:body use="literal"
                                       parts="ClaimDetail"
                                       namespace="gost.e-ecolog.ru                    >
                    </mime:part>
                    <mime:part>
                        <mime:content part="ClaimPhoto"
                            type="image/jpeg"/>
                    </mime:part>
                </mime:multipartRelated>
            </wsdl:input>
            <wsdl:output>
                <soapbind:body use="literal"
                               namespace="gost.e-ecolog.ru                    >
            </wsdl:output>
        </wsdl:operation>
    </wsdl:binding>
</wsdl:definitions>
```

Результирующее входящее сообщение операции "SendClaim" для rpc/literal:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
        start="<gost.e-ecolog.ru                    >"
Content-Description: This is the optional message description.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                    >
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                    >
    <SOAP-ENV:Body xmlns:types="http://example.com/mimetypes">
        <types:SendClaim>
            <ClaimDetail>
                <Name>...</Name>
                <ClaimForm>cid:claimform@example.com</ClaimForm>
            </ClaimDetail>
        </types:SendClaim>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
--MIME_boundary
Content-Type: text/xml
Content-Transfer-Encoding: 8bit
Content-ID: <cl gost.e-ecolog.ru                    >
...Форма претензии,  на которую указывает ссылка swaRef...
--MIME_boundary
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-ID: <ClaimPhoto=gost.e-ecolog.ru                    >
...Вложение MIME цифровой фотографии в двоичном представлении...
--MIME_boundary—
```

Результирующее исходящее сообщение операции "SendClaim" для rpc/literal:

```
MIME-Version: 1.0
Content-Type: text/xml; charset=UTF-8
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                    >
      <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                          >
            <types:SendClaimResponse>
                  <ClaimRefNo>.....</ClaimRefNo>
            </types:SendClaimResponse>
      </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

ПРАВИЛЬНО:

Описание WSDL для связывания document/literal

```
<?xml version="1.0" encoding="utf-8" ?>
<wsdl:definitions xmlns:types="gost.e-ecolog.ru                              "
      xmlns:ref="gost.e-ecolog.ru                        /xsd"
      xmlns:xsd="gost.e-ecolog.ru                  )1/XMLSchema"
      xmlns:soapbind="gost.e-ecolog.ru
      xmlns:wsdl="gost.e-ecolog.ru
      xmlns:mime="gost.e-ecolog.ru
      targetNamespace="gost.e-ecolog.ru
      xmlns:tns="gost.e-ecolog.ru                        >
  <wsdl:types>
      <xsd:schema targetNamespace="gost.e-ecolog.ru                              "
          xmlns:xsd="gost.e-ecolog.ru                              ">

            <xsd:import namespace="http://ws-i.org/profiles/basic/1.1/xsd" />
            <xsd:element name="ClaimDetail" type="types:ClaimDetailType"/>
            <xsd:complexType name="ClaimDetailType">
            <xsd:sequence>
                  <xsd:element name="Name" type="xsd:string"/>
                  <xsd:element name="ClaimForm" type="ref:swaRef"/>
            </xsd:sequence>
             </xsd:complexType>
             <xsd:element name="ClaimRefNo" type="xsd:string"/>
      </xsd:schema>
  </wsdl:types>

  <wsdl:message name="ClaimIn">
      <wsdl:part name="body" element="types:ClaimDetail"/>
      <wsdl:part name="ClaimPhoto" type="xsd:base64Binary"/>
  </wsdl:message>

  <wsdl:message name="ClaimOut">
      <wsdl:part name="out" element="types:ClaimRefNo"/>
  </wsdl:message>

  <wsdl:portType name="ClaimPortType">
      <wsdl:operation name="SendClaim">
            <wsdl:input message="tns:ClaimIn"/>
            <wsdl:output message="tns:ClaimOut"/>
      </wsdl:operation>
  </wsdl:portType>

  <wsdl:binding name="ClaimBinding" type="tns:ClaimPortType">
```

```
            <soapbind:binding style="document"
                        transport="http://schemas.xmlsoap.org/soap/http"/>
            <wsdl:operation name="SendClaim">
                <soapbind:operation soapAction="gost.e-ecolog.ru                    >
                <wsdl:input>
                 <mime:multipartRelated>
                        <mime:part>
                            <soapbind:body parts="body" use="literal"/>
                        </mime:part>
                        <mime:part>
                            <mime:content part="ClaimPhoto" type="image/jpeg"/>
                        </mime:part>
                       </mime:multipartRelated>
                </wsdl:input>
                <wsdl:output>
                     <soapbind:body use="literal" />
                </wsdl:output>
            </wsdl:operation>
    </wsdl:binding>
</wsdl:definitions>
```

Результирующее входящее сообщение операции "SendClaim" для document/literal:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
      start="<gost.e-ecolog.ru                    >"
Content-Description: Это необязательное описание сообщения.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                    >
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                    >
     <SOAP-ENV:Body xmlns:types="http://example.com/mimetypes">
                <types:ClaimDetail>
                 <Name>...</Name>
                 <ClaimForm>cid:claimform@example.com</ClaimForm>
           </types:ClaimDetail>
     </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
--MIME_boundary
Content-Type: text/xml
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                    >
... Форма претензии, на которую указывает ссылка swaRef...
--MIME_boundary
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-ID: <ClaimPhoto=gost.e-ecolog.ru                                                    >
... Вложение MIME цифровой фотографии в двоичном представлении ..
--MIME_boundary--
```

Результирующее исходящее сообщение операции "SendClaim" для document/literal:

```
MIME-Version: 1.0
Content-Type: text/xml; charset=UTF-8
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                    ">
```

```
 <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                    ">
    <types:ClaimRefNo>.....</types:ClaimRefNo>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.5 Спецификация корневой части

Сообщения SOAP с вложениями требуют, чтобы корневая часть составного/связанного пакета содержала конверт SOAP, но привязка WSDL MIME не объясняет, как это должно определяться.

R2911 Элемент *mime:multipartRelated* в **ОПИСАНИИ** ДОЛЖЕН содержать в числе его дочерних элементов *mime:part* ровно один элемент *mime:part*, содержащий дочерний элемент *soapbind:body*. С

В привязке WSDL MIME элемент *mime:part*, содержащий *soapbind:body*, описывает корневую часть MIME, необходимую для сообщений SOAP с вложениями.

Пример:

НЕПРАВИЛЬНО:

```
<wsdl:definitions xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:soapbind="gost.e-ecolog.ru
 xmlns:wsdl="gost.e-ecolog.ru
 xmlns:mime="gost.e-ecolog.ru
 targetNamespace="gost.e-ecolog.ru
 xmlns:tns="gost.e-ecolog.ru                    >
...
 <wsdl:binding name="aBinding" type="tns:aPortType">
      <soapbind:binding style="rpc"
                    transport="gost.e-ecolog.ru                    >
      <wsdl:operation name="anOperation">
            <soap:operation soapAction="gost.e-ecolog.ru                    >
            <wsdl:input>
             <mime:multipartRelated>
                  <mime:part>
                      <soapbind:body use="literal"
                                  namespace="gost.e-ecolog.ru                    >
                        </mime:part>
                  <mime:part>
                      <soapbind:body use="literal"
                                  namespace="gost.e-ecolog.ru                    >
                   </mime:part>
                  </mime:multipartRelated>
      </wsdl:input>
      <wsdl:output>
...
      </wsdl:output>
  </wsdl:operation>
 </wsdl:binding>
</wsdl:definitions>
```

4.6 Спецификация заголовков SOAP в корневой части

В спецификации WSDL1.1 не указано, допускается ли элемент *soapbind:header* как дочерний элемента *mime:part* наряду с элементом *soapbind:body*. Спецификация сообщений SOAP с вложениями требует, чтобы корневая часть составного сообщения содержала конверт SOAP, однако в спецификации WSDL1.1 нет четких указаний, как определить эту часть. Поскольку спецификация WSDL1.1 определяет, что элемент *mime:part* используется для описания каждой части составного/связанного сообщения, содержание элемента *mime:part*, представляющего основную часть составного сообщения, должно полностью определить конверт SOAP, включая элементы *soapbind:body* и *soapbind:header*, таким же образом, как это было бы сделано в случае отсутствия расширения привязки WSDL MIME.

R2905 Элемент *soapbind:header* МОЖЕТ быть включен в **ОПИСАНИЕ** как дочерний элемент элемента *mime:part*. С

R2906 Элемент *soapbind:header* **ОПИСАНИЯ** НЕ ДОЛЖЕН быть включен в часть *mime:part*, которая не является корневой, содержащей элемент *soapbind:body*. С

Пример:

НЕПРАВИЛЬНО:

```
<wsdl:definitions xmlns:xsd="gost.e-ecolog.ru                                    "
 xmlns:soapbind="gost.e-ecolog.ru
 xmlns:wsdl="gost.e-ecolog.ru                              "
 xmlns:mime="gost.e-ecolog.ru                                        "
 targetNamespace="gost.e-ecolog.ru
 xmlns:tns="gost.e-ecolog.ru                              >
...
  <wsdl:binding name="aBinding" type="tns:aPortType">
    <soapbind:binding style="rpc"
 transport="gost.e-ecolog.ru                                    >
     <wsdl:operation name="anOperation">
       <soap:operation soapAction="gost.e-ecolog.ru                                    >
         <wsdl:input>
 <mime:multipartRelated>
   <mime:part>
     <soapbind:body use="literal"
           namespace="gost.e-ecolog.ru                              >
   </mime:part>
   <mime:part>
      <soapbind:header message="tns:headerMessage"
          part="aPart"
          use="literal"/>
 </mime:part>
           </mime:multipartRelated>
     </wsdl:input>
     <wsdl:output>
...
     </wsdl:output>
   </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

4.7 Исправления схемы привязки MIME

Имеется ряд несоответствий между спецификацией WSDL1.1 и схемой привязки WSDL MIME. В случае элемента *mime:part* схема неправильно идентифицирует его как определение локального элемента, следствием чего является неправильное добавление атрибута имени (*name*), который не предусмотрен в спецификации WSDL1.1. Это и другие исправления к расширению привязки WSDL MIME нашли отражение в пересмотренной схеме, расположенной по адресу:

"gost.e-ecolog.ru   ".

R2907 Части MIME в **ОПИСАНИИ** ДОЛЖНЫ быть определены с использованием элементов с локальными именами part в пространстве имен расширения привязки WSDL MIME. С

R2908 У элемента *mime:part* в **ОПИСАНИИ** НЕ ДОЛЖНО быть атрибута name.

4.8 Определение альтернативных типов медиа (*Media Types*)

Многократные дочерние элементы *mime:content* элемента *mime:part* считаются допустимой альтернативной сериализацией части *wsdl:part*, на которую они ссылаются.

R2909 Многократные дочерние элементы *mime:content* элемента *mime:part* в **ОПИСАНИИ** ДОЛЖНЫ сослаться на один и тот же элемент *wsdl:part.*

Пример:

НЕПРАВИЛЬНО:

```
<mime:part>
  <mime:content part="ns:foo" type="image/jpeg"/>
  <mime:content part="ns:bar" type="image/jpeg"/>
</mime:part>
```

ПРАВИЛЬНО:

```
<mime:part>
  <mime:content part="ns:foo" type="image/jpeg"/>
  <mime:content part="ns:foo" type="image/gif"/>
</mime:part>
```

4.9 Части WSDL

R2910 Элемент *mime:content* в **ОПИСАНИИ** ДОЛЖЕН ссылаться на *wsdl:part*, который определен с использованием либо атрибута *type*, либо атрибута *element*.

R2942 В **СООБЩЕНИИ** часть сообщения, связанная с элементом *mime:content*, который ссылается на глобальное объявление элемента (через атрибут element элемента *wsdl:part*), ДОЛЖНА быть сериализирована в пределах части MIME как преобразование в последовательную форму инфо-набора XML, корневой элемент которого описан элементом, на который ссылаются.

R2943 В **ОПИСАНИИ** если часть сообщения связана с элементом *mime:content*, который ссылается на тип (через атрибут type элемента *wsdl:part*), то значение этого атрибута типа ДОЛЖНО быть проигнорировано в пользу типа медиа *(media type)* атрибута типа элемента *mime:content*.

R2944 *В* **ОПИСАНИИ** если *wsdl:part* элемент ссылается на глобальное объявление элемента (через атрибут type элемента *wsdl:part*), то значение атрибута type элемента *mime:content*, который связывает эту часть, ДОЛЖНО быть типом контента, пригодным для XML сериализации.

4.10 Порядок частей

R2912 **ПОЛУЧАТЕЛЬ** НЕ ДОЛЖЕН полагать, что порядок элементов *mime:part*, определенных в описании WSDL, является тем же самым, что и порядок частей MIME в сообщении.

R2947 В ОПИСАНИИ элемент *mime:part*, который содержит дочерний элемент *soapbind:body*, МОЖЕТ находиться в любой позиции среди других дочерних элементов элемента *mime:multipartRelated*.

Порядок частей MIME, определенных в описании WSDL, нужно считать независимым от порядка частей MIME в сообщении.

4.11 Отправка сообщений об ошибках

R2913 **СООБЩЕНИЕ** об ошибке МОЖЕТ быть сериализировано либо как *text/xml*, либо как *multipart/related* в случае, если у дочернего элемента *wsdl:output* соответствующей операции привязки имеется дочерний элемент *mime:multipartRelated*.

4.12 Описание ошибок

R2930 Элемент *wsdl:fault* в **ОПИСАНИИ** НЕ ДОЛЖЕН иметь дочерних элементов *mime:multipartRelated*.

4.13 Отправка дополнительных частей, не описанных в WSDL

Помимо описанных в WSDL, в сообщение могут быть включены дополнительные части MIME и их позиция, или их порядок в пределах пакета MIME не имеет значения.

R2923 **ОТПРАВИТЕЛЬ** МОЖЕТ посылать некорневые части MIME, не описанные в привязке WSDL MIME. С

R2926 **В СООБЩЕНИЕ** ДОЛЖНЫ входить все части MIME, описанные в его привязке WSDL MIME.

4.14 Соответствие сообщений SOAP

Критерии соответствия профиля для цели соответствия КОНВЕРТ применимы только для конверта SOAP, который содержится в корневой части пакета MIME. Конверты SOAP в некорневых частях могут быть определены в описании WSDL как вложения, и тогда применимы критерии соответствия для неосновных частей, перечисленных в описании WSDL.

R2927 Корневая часть **СООБЩЕНИЯ** ДОЛЖНА быть совместима по всем требованиям для конверта версии 1.1 Основного Профиля.

4.15 Пример описания вложения с использованием *mime:conent*

Пример

ПРАВИЛЬНО:

Описание WSDL ***для*** document/literal:

```
<?xml version="1.0" encoding="utf-8" ?>
<wsdl:definitions xmlns:types="gost.e-ecolog.ru                    "
        xmlns:ref="gost.e-ecolog.ru                "
        xmlns:xsd="gost.e-ecolog.ru                    "
        xmlns:soapbind="gost.e-ecolog.ru                        "
        xmlns:wsdl="gost.e-ecolog.ru
        xmlns:mime="gost.e-ecolog.ru
        targetNamespace="gost.e-ecolog.ru
        xmlns:tns="gost.e-ecolog.ru                    >
  <wsdl:types>
     <xsd:schema targetNamespace="gost.e-ecolog.ru
          xmlns:xsd="gost.e-ecolog.ru                    ">
```

```
            <xsd:element name="ClaimDetail" type="types:ClaimDetailType"/>
           <xsd:complexType name="ClaimDetailType">
            <xsd:sequence>
                  <xsd:element name="Name" type="xsd:string"/>
     <!—Подробности описания претензии -->
 </xsd:sequence>
   </xsd:complexType>

   <xsd:element name="ClaimRefNo" type="xsd:string"/>
  </xsd:schema>
</wsdl:types>

<wsdl:message name="ClaimIn">
      <wsdl:part name="body" element="types:ClaimDetail"/>
      <wsdl:part name="ClaimPhoto" type="xsd:base64Binary"/>
</wsdl:message>

<wsdl:message name="ClaimOut">
     <wsdl:part name="out" element="types:ClaimRefNo"/>
</wsdl:message>
<wsdl:portType name="ClaimPortType">
     <wsdl:operation name="SendClaim">
          <wsdl:input message="tns:ClaimIn"/>
          <wsdl:output message="tns:ClaimOut"/>
    </wsdl:operation>
</wsdl:portType>

<wsdl:binding name="ClaimBinding" type="tns:ClaimPortType">
     <soapbind:binding style="document"
               transport="gost.e-ecolog.ru                                    >
    <wsdl:operation name="SendClaim">
         <soapbind:operation soapAction="http://example.com/soapaction"/>
        <wsdl:input>
         <mime:multipartRelated>
              <mime:part>
                    <soapbind:body parts="body" use="literal"/>
              </mime:part>
              <mime:part>
                   <mime:content part="ClaimPhoto" type="image/jpeg"/>
              </mime:part>
       </mime:multipartRelated>
       </wsdl:input>
       <wsdl:output>
       <soapbind:body use="literal" />
       </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

Результирующее входящее сообщение для операции "SendClaim" для document/literal:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
       start="<rootpart@example.com>"
Content-Description: This is the optional message description.
--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                    >
```

```
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                    ">
    <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                        >
 <types:ClaimDetail>
   <Name>...</Name>
   <!-- Подробности описания претензии -->
 </types:ClaimDetail>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

--MIME_boundary
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-ID: <ClaimPhoto=gost.e-ecolog.ru                                                    >
```

... Вложение MIME цифровой фотографии в двоичном представлении...
--MIME_boundary--

Результирующее исходящее сообщение для операции "SendClaim" для document/literal:

```
MIME-Version: 1.0
Content-Type: text/xml; charset=UTF-8

<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                    >
    <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                        >
 <types:ClaimRefNo>.....</types:ClaimRefNo>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope
```

4.16 Пример описания вложения с использованием *swaRef*

Пример
ПРАВИЛЬНО:
Описание WSDL для документа/литерала (document/literal):

```
<?xml version="1.0" encoding="utf-8" ?>
<wsdl:definitions xmlns:types="gost.e-ecolog.ru
        xmlns:ref="gost.e-ecolog.ru                                  "
        xmlns:xsd="gost.e-ecolog.ru                                    "
        xmlns:soapbind="gost.e-ecolog.ru                                  "
        xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
        xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
        targetNamespace="gost.e-ecolog.ru
        xmlns:tns="gost.e-ecolog.ru                              ">

   <wsdl:types>
    <xsd:schema targetNamespace="gost.e-ecolog.ru                                  "
        xmlns:xsd="gost.e-ecolog.ru                                  >

     <xsd:import namespace="gost.e-ecolog.ru                                  " />
    <xsd:element name="ClaimDetail" type="types:ClaimDetailType"/>
    <xsd:complexType name="ClaimDetailType">
    <xsd:sequence>
        <xsd:element name="Name" type="xsd:string"/>
        <!-- Подробности описания претензии -->
        <xsd:element name="ClaimPhoto" type="ref:swaRef"/>
    </xsd:sequence>
    </xsd:complexType>
```

```
      <xsd:element name="ClaimRefNo" type="xsd:string"/>

    </xsd:schema>
  </wsdl:types>

  <wsdl:message name="ClaimIn">
      <wsdl:part name="body" element="types:ClaimDetail"/>
  </wsdl:message>
  <wsdl:message name="ClaimOut">
      <wsdl:part name="out" element="types:ClaimRefNo"/>
  </wsdl:message>

  <wsdl:portType name="ClaimPortType">
    <wsdl:operation name="SendClaim">
      <wsdl:input message="tns:ClaimIn"/>
      <wsdl:output message="tns:ClaimOut"/>
    </wsdl:operation>
  </wsdl:portType>

  <wsdl:binding name="ClaimBinding" type="tns:ClaimPortType">
    <soapbind:binding style="document"
                      transport="gost.e-ecolog.ru                                    >
    <wsdl:operation name="SendClaim">
          <soapbind:operation soapAction="gost.e-ecolog.ru                          >
          <wsdl:input>
                <mime:multipartRelated>
              <mime:part>
                      <soapbind:body parts="body" use="literal"/>
              </mime:part>
        </mime:multipartRelated>
      </wsdl:input>
      <wsdl:output>
                  <soapbind:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

Результирующее входящее сообщение для операции «SendClaim» для document/literal:

```
MIME-Version: 1.0
Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
      start="<gost.e-ecolog.ru                    >"
Content-Description: Это необязательное описание сообщения.

--MIME_boundary
Content-Type: text/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit
Content-ID: <gost.e-ecolog.ru                    >

<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="gost.e-ecolog.ru                                          >
   <SOAP-ENV:Body xmlns:types="gost.e-ecolog.ru                             ">
      <types:ClaimDetail>
         <Name>...</Name>
         <!-- Подробности описания претензии -->
         <ClaimPhoto>cid:claimphoto@example.com</ClaimPhoto>
    </types:ClaimDetail>
```

```
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
--MIME_boundary
Content-Type: image/jpeg
Content-Transfer-Encoding: binary
Content-ID: <gost.e-ecolog.ru                    >
... Вложение MIME цифровой фотографии в двоичном представлении...
--MIME_boundary--
```

Результирующее исходящее сообщение для операции «SendClaim» для document/literal:

```
MIME-Version: 1.0
Content-Type: text/xml; charset=UTF-8
<?xml version='1.0' ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Body xmlns:types="http://example.com/mimetypes">
 <types:ClaimRefNo>.....</types:ClaimRefNo>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Приложение А

Ссылки на спецификации

В профиль включены ссылки на требования следующих спецификаций, за исключением тех случаев, когда требования заменяются требованиями самого профиля:

- Сообщения SOAP с вложениями (SOAP Messages with Attachments)
- Расширяемый язык разметки (XML) 1.0 (Вторая Редакция) (Extensible Markup Language (XML) 1.0 (Second Edition))
- Пространства имен в XML 1.0 (Namespaces in XML 1.0)
- RFC2557 Инкапсуляция MIME составных документов, таких как HTML (MHTML) (RFC2557 MIME Encapsulation of Aggregate Documents, such as HTML (MHTML))
- RFC2045 Многоцелевое расширение возможностей почты в сети Интернет (MIME). Часть 1 : Формат интернет- сообщений (RFC2045 Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies)
- RFC2046 Многоцелевое расширение возможностей почты в сети Интернет (MIME). Часть 2: Типы медиа (RFC2046 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types)
- RFC2392 Универсальные локаторы ресурса «Content-ID» и «Message-ID» (RFC2392 Content-ID and Message-ID Uniform Resource Locators)
- Язык описания Веб-служб WSDL 1.1, Раздел 5.0 (WSDL 1.1, Section 5.0)

Приложение В

Возможные расширения

Этот раздел идентифицирует возможные расширения, как определено в «Области применения профиля», для спецификаций компонентов профиля.

Данные механизмы находятся вне области применения профиля; их использование может повлиять на функциональную совместимость и может потребовать частного соглашения сторон с Веб-сервисами.

В сообщении SOAP с вложением:

• E0001 – **Части MIME** — Сообщения SOAP с вложениями не устанавливают ограничения для типа любой некорневой части в составном/связанном (multipart/related) сообщении

Приложение С

Нормативные ссылки

В дополнение ко всем профильным спецификациям, перечисленным в Приложении А, в материале использовались также ссылки на следующие спецификации:

• RFC2119 Ключевые слова для использования в RFC для указания уровня требования, gost.e-ecolog.ru , S. Bradner, Март 1997.

• Базовый профиль WS-I, версия 1.0, gost.e-ecolog.ru gost.e-ecolog.ru ar и др., Апрель 2004.

• Пространства имен в XML 1.0 (Второе издание). gost.e-ecolog.ru , T. Bray и др., Август 2006.

• Механизм вложения декларации о соответствии WS-I, версия 1.0, gost.e-ecolog.ru gost.e-ecolog.ru , M. Nottingham и др., Ноябрь 2004.

Приложение D

Определения терминов

Здесь приведен список терминов, которые имеют конкретные определения, используемые в контексте для данного профиля:

- кодирование идентификатора контента части *(content-id part encoding)*

«Кодирование идентификатора контента части» представляет собой операцию конкатенации величин:

• Значения атрибута name элемента *wsdl:part*, на который ссылается *mime:content*, и в котором символы, недопустимые в заголовках Content-ID (не ASCII символы, представленные кодами большими 0x7F), представляется следующим образом:

o Каждый недопустимый символ преобразован в один или более байтов в соответствии с UTF-8.

o Все байты, соответствующие недопустимым символам, экранируются согласно правилам экранирования символов в URI (т.е. преобразуются в %HH, где HH – шестнадцатеричная нотация значения байта),

o Исходный символ заменяется полученной последовательностью символов .

• Символ '=' (0x3D).

• Глобально уникальное значение, такое, как UUID.

• Символ '@' (0x40).

• Действительное имя домена, от имени которого создается сообщение

Приложение Е

Благодарности

Этот документ является результатом работы рабочей группы по базовому профилю WS-I, в которую входили:

Mark Allerton (Crystal Decisions Corp), Steve Anderson (OpenNetwork), George Arriola (Talking Blocks, Inc.), Siddharth Bajaj (Verisign), Keith Ballinger (Microsoft Corp.), David Baum (Kantega AS), Ilya Beyer (KANA), Rich Bonneau (IONA Technologies), Don Box (Microsoft Corp.), Andrew Brown (Verisign), Heidi Buelow (Quovadx), David Burdett (Commerce One, Inc.), Luis Felipe Cabrera (Microsoft Corp.), Maud Cahuzac (France Telecom), Mike Chadwick (Kaiser Permanente), Martin Chapman (Oracle Corporation), Richard Chennault (Kaiser Permanente), Roberto Chinnici (Sun Microsystems), Dipak Chopra (SAP AG), Jamie Clark (OASIS), David Cohen (Merrill Lynch), Ugo Corda (SeeBeyond Tech), Paul Cotton (Microsoft Corp.), Joseph Curran (Accenture), Alex Deacon (Verisign), Mike DeNicola (Fujitsu Limited), Paul Downey (BT Group), Jacques Durand (Fujitsu Limited), Aladin Eajani (Hummingbird, Ltd.), Michael Eder (Nokia), Dave Ehnebuske (IBM), Mark Ericson (Mindreef Inc), Colleen Evans (Microsoft Corp.), Tim Ewald (Microsoft Corp.), Chuck Fay (FileNET Corp.), Chris Ferris (IBM), Daniel Foody (Actional Corporation), Satoru Fujita (NEC Corporation), Shishir Garg (France Telecom), Yaron Goland (BEA Systems Inc), Marc Goodner (SAP AG), Pierre Goyette (Hummingbird, Ltd.), Hans Granqvist (Verisign), Martin Gudgin (Microsoft Corp.), Marc Hadley (Sun Microsystems), Norma Hale (Webify Solutions Inc), Bob Hall (Unisys Corporation), Scott Hanselman (Corillian), Muir Harding (Autodesk Inc.), Loren Hart (Verisign), Andrew Hately (IBM), Harry Holstrom (Accenture), Lawrence Hsiung (Quovadx), Hemant Jain (Tata Consultancy), Steve Jenisch (SAS Institute), Erik Johnson (Epicor Software), Bill Jones (Oracle Corporation), Anish Karmarkar (Oracle Corporation), Dana Kaufman (Forum Systems), Takahiro Kawamura (Toshiba), Oldre Kepka (Systinet), Bhushan Khanal (WRQ Inc.), Sandy Khaund (Microsoft Corp.), Jacek Kopecky (Systinet), Sanjay Krishnamurthi (Informatica), Sundar Krishnamurthy (Verisign), Eva Kuiper (Hewlett-Packard), Sunil Kunisetty (Oracle Corporation), Christopher Kurt (Microsoft Corp.), Lars Laakes (Microsoft Corp.), Canyang Kevin Liu (SAP AG), Ted Liu (webMethods Inc.), Donna Locke (Oracle Corporation), Brad Lund (Intel), Michael Mahan (Nokia), Ron Marchi (EDS), Jonathan Marsh (Microsoft Corp.), Eric Matland (Hummingbird, Ltd.), Barbara McKee (IBM), Derek Medland (Hummingbird, Ltd.), David Meyer (Plumtree Software Inc.), Jeff Mischkinsky (Oracle Corporation), Ray Modeen (MITRE Corp.), Tom Moog (Sarvega Inc.), Gilles Mousseau (Hummingbird, Ltd.), Greg Mumford (MCI), Jim Murphy (Mindreef Inc), Bryan Murray (Hewlett-Packard), Richard Nikula (BMC Software, Inc.), Eisaku Nishiyama (Hitachi, Ltd.), Mark Nottingham (BEA Systems Inc), David Orchard (BEA Systems Inc), Vivek Pandey (Sun Microsystems), Jesse Pangburn (Quovadx), Eduardo Pelegri-Llopart (Sun Microsystems), Mike Perham (Webify Solutions Inc), Eric Rajkovic (Oracle Corporation), Shaan Razvi (MITRE Corp.), Rimas Rekasius (IBM), Mark Richards (Fidelity), Graeme Riddell (Bowstreet), Sam Ruby (IBM), Tom Rutt (Fujitsu Limited), Saikat Saha (Commerce One, Inc.), Roger Sanborn (Crystal Decisions Corp), Matt Sanchez (Webify Solutions Inc), Krishna Sankar (Cisco Systems Inc.), Jeffrey Schlimmer (Microsoft Corp.), Don Schricker (Micro Focus), Dave Seidel (Mindreef Inc), AKIRA SHIMAYA (NTT), David Shoaf (Hewlett-Packard), Yasser Shohoud (Microsoft Corp.), David Smiley (Ascential Software), Seumas Soltysik (IONA Technologies), Joseph Stanko (Plumtree Software Inc.), Andrew Stone (Accenture), Julie Surer (MITRE Corp.), YASUO TAKEMOTO (NTT), Nobuyoshi Tanaka (NEC Corporation), Jorgen Thelin (Microsoft Corp.), Sameer Vaidya (Talking Blocks, Inc.), William Vambenepe (Hewlett-Packard), Claus von Riegen (SAP AG), Rick Weil (Eastman Kodak Company), Scott Werden (WRQ Inc.), Ajamu Wesley (IBM), Ian White (Micro Focus), Dave Wilkinson (Vignette), Mark Wood (Eastman Kodak Company), Prasad Yendluri (webMethods Inc.), and Brandon Zhu (NetManage Inc).

УДК: 001.4:025.4:006.354 ОКС: 35.100.05

Ключевые слова: интероперабельность, профиль вложений

Подписано в печать 01.12.2014. Формат 60x84 1/8.

Усл. печ. л. 3,72. Тираж 31 экз. Зак. 3592.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ФГУП «СТАНДАРТИНФОРМ»

123995 Москва, Гранатный пер., 4.

gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru