МЕЖГОСУДАРСТВЕННЫЙ СОВЕТ ПО СТАНДАРТИЗАЦИИ, МЕТРОЛОГИИ И СЕРТИФИКАЦИИ
(МГС)

INTERSTATE COUNCIL FOR STANDARDIZATION, METROLOGY AND CERTIFICATION
(ISC)

МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ

ГОСТ ISO/IEC 24824-2— 2013

Информационные технологии

ОБЩИЕ ПРАВИЛА ПРИМЕНЕНИЯ ASN. 1

Быстрые сетевые услуги

Часть 2

(ISO/IEC 24824-2:2006, IDT)

Издание официальное

Москва
Стандартинформ
2015

gost.e-ecolog.ru

Предисловие

Цели, основные принципы и основной порядок проведения работ по межгосударственной стандартизации установлены в ГОСТ 1.0—92 «Межгосударственная система стандартизации. Основные положения» и ГОСТ 1.2—2009 «Межгосударственная система стандартизации. Стандарты межгосударственные, правила и рекомендации по межгосударственной стандартизации. Правила разработки, принятия, применения, обновления и отмены»

Сведения о стандарте

1 ПОДГОТОВЛЕН Федеральным государственным унитарным предприятием Государственный научно-исследовательский и конструкторско-технологический институт «ТЕСТ» (ФГУП ГосНИИ «ТЕСТ»)

2 ВНЕСЕН Федеральным агентством по техническому регулированию и метрологии (Росстандарт)

3 ПРИНЯТ Межгосударственным советом по стандартизации, метрологии и сертификации (протокол от 14 ноября 2013 г. № 44)

За принятие проголосовали:

Краткое наименование страны по МК (ИСО 3166) 004—97	Код страны по МК (ИСО 3166) 004—97	Сокращенное наименование национального органа по стандартизации
Армения	AM	Минэкономики Республики Армения
Киргизия	KG	Кыргызстандарт
Россия	RU	Росстандарт

4 Приказом Федерального агентства по техническому регулированию и метрологии от 11 июня 2014 г. № 566-ст межгосударственный стандарт ГОСТ ISO/IEC 24824-2—2013 введен в действие в качестве национального стандарта Российской Федерации с 1 сентября 2015 г.

5 Настоящий стандарт идентичен международному стандарту ISO/IEC 24824-2:2006 Information technology — Generic applications of ASN.1: Fast Web Services — Part 2 (Информационные технологии. Общие правила применения ASN.1. Быстрые сетевые услуги. Часть 2).

Перевод с английского языка (en).

Сведения о соответствии межгосударственных стандартов ссылочным международным стандартам приведены в дополнительном приложении ДА.

Степень соответствия — идентичная (IDT)

6 ВВЕДЕН ВПЕРВЫЕ

Информация об изменениях к настоящему стандарту публикуется в ежегодном информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячном информационном указателе «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

© Стандартинформ, 2015

В Российской Федерации 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
2 Нормативные ссылки . . . 1
2.1 Идентичные рекомендации и международные стандарты . . . 2
2.2 Дополнительные ссылки . . . 3
3 Термины и определения . . . 4
3.1 Заимствованные термины . . . 4
3.2 Дополнительные термины . . . 4
4 Сокращения . . . 5
5 Нотация . . . 6
6 Обработка ASN.1 SOAP сообщений . . . 6
7 Отображение компонентов типа **Envelope** в единицах информации . . . 8
7.1 Общие положения . . . 8
7.2 Отображение типа **Header** . . . 8
7.3 Отображение типа **Body** . . . 9
7.4 Отображение типа **Fault** . . . 9
7.5 Отображение типа **Content** . . . 10
8 Отображение инфо-наборов W3C SOAP сообщений в абстрактные значения типа **Envelope** . . . 11
8.1 Общие положения . . . 11
8.2 Отображение **Header EII** . . . 11
8.3 Отображение **Body EII** . . . 12
8.4 Отображение **Fault EII** . . . 12
8.5 Отображение содержимого EII в значение типа **Content** . . . 13
9 Расширенная SOAP обработка встроенных в ASN.1 закодированных значений . . . 14
9.1 Общие положения . . . 14
9.2 Определение типа ASN.1 встроенного в ASN.1 закодированного значения . . . 15
9.3 Формирование значений ASN.1 из определенных встроенных в ASN.1 закодированных значений 15
9.4 Вставка значений ASN.1 (с идентификатором) в W3C SOAP сообщение . . . 15
9.5 Ошибка «неидентифицируемый тип ASN.1» . . . 16
10 ASN.1 SOAP HTTP привязка . . . 17
10.1 HTTP медиа тип . . . 17
10.2 Поведение отвечающих SOAP узлов . . . 17
11 SOAP сообщения быстрого инфо-набора и SOAP HTTP привязка . . . 18
12 SOAP-ориентированные описания сервисов, поддерживающие интерфейсную привязку ASN.1 SOAP . . . 18
12.1 Общие положения . . . 18
12.2 Схемы . . . 18
12.3 Абстрактные интерфейсы и абстрактные операции . . . 19
12.4 Интерфейсные и операционные привязки . . . 19
12.5 RPC схема . . . 21
13 Использование SOAP-ориентированных описаний сервисов с интерфейсными привязками ASN.1 SOAP . . . 22
Приложение А (обязательное) ASN.1 модуль для ASN.1 SOAP . . . 24
Приложение В (обязательное) MIME медиа типы для быстрых веб-сервисов . . . 25
Приложение С (справочное) Учебное пособие по использованию быстрых веб-сервисов . . . 28
Приложение D (справочное) Общее предоставление сервисов с использованием быстрых веб-сервисов и веб-сервисов XML . . . 34
Приложение Е (справочное) SOAP-ориентированное описание сервиса в WSDL 1.1 . . . 36
Приложение F (справочное) Присвоение значений идентификаторов объектов . . . 40
Приложение ДА (справочное) Сведения о соответствии межгосударственных стандартов ссылочным международным стандартам (международным документам) . . . 41
Библиография . . . 42

М Е Ж Г О С У Д А Р С Т В Е Н Н Ы Й С Т А Н Д А Р Т

Информационные технологии

ОБЩИЕ ПРАВИЛА ПРИМЕНЕНИЯ ASN.1

Быстрые сетевые услуги

Часть 2

Information technologies. Generic applications of ASN.1. Fast web services. Part 2

Дата введения — 2015—09—01

1 Область применения

В настоящем стандарте определены типы сообщений и принципы кодирования, обеспечивающие применение быстрых веб-сервисов (Fast Web Services), и способы описания таких сервисов.

Использованный для поддержки этих сервисов протокол соответствует требованиям модели обработки сервис-ориентированной архитектуры (SOAP) (см. W3C SOAP Часть 1, раздел 2) и основан на передаче:

a) ASN.1 SOAP сообщений, которые содержат закодированные значения ASN.1 и документы быстрого инфо-набора; и

b) SOAP сообщений быстрого инфо-набора.

В стандарте также специфицированы:

- модуль ASN.1 для ASN.1 SOAP, который определяет тип **`Envelope`** (значение этого типа соответствует ASN.1 SOAP сообщению);
- концептуальное отображение между ASN.1 SOAP сообщениями и W3C SOAP сообщениями (определенные как экземпляр инфо-набора, см. W3C SOAP Часть 1, раздел 5);
- расширение модели обработки W3C SOAP для обработки встроенных закодированных значений ASN.1;
- ASN.1 SOAP HTTP привязка, которая является модификацией и расширением SOAP HTTP привязки, для передачи ASN.1 SOAP сообщений;
- поддержка для передачи инфо-наборов W3C SOAP сообщений, сериализованных в виде документов быстрых инфо-наборов;
- описания ориентированных на SOAP сервисов, которые определяют интерфейс и семантику быстрых веб-сервисов (Fast Web Services).

В настоящем стандарте выделены два медиа типа многоцелевых расширений интернет-почты (MIME) для идентификации.

- ASN.1 SOAP сообщений, закодированных с помощью базовых правил уплотненного кодирования (PER) с выравниванием;
- сообщений быстрого инфо-набора SOAP.

2 Нормативные ссылки

Для применения настоящего стандарта необходимы следующие ссылочные документы. Для датированных ссылок применяют только указанное издание ссылочного документа, для недатированных ссылок применяют последнее издание ссылочного документа (включая все его изменения).

Издание официальное

2.1 Идентичные рекомендации и международные стандарты

- ITU-T Recommendation X.660 (2004) | ISO/IEC 9834-1:2005[1] Information technology — Open Systems Interconnection — Procedures for the operation of OSI Registration Authorities: General procedures and top arcs of the ASN.1 Object Identifier tree (Рекомендация МСЭ-Т X.660 (2004) | ISO/IEC 9834-1:2005 Информационные технологии. Процедуры для работы регистрационных органов по идентификации объекта. Часть 1. Общие процедуры и высшие разряды дерева идентификаторов объекта международного объекта)

- ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002[2] Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation (Рекомендация МСЭ-Т X.680 (2002) | ISO/IEC 8824-1:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 1. Спецификация базовой нотации)

- ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002[3] Information technology — Abstract Syntax Notation One (ASN.1): Information object specification (Рекомендация МСЭ-Т X.681 (2002) | ISO/IEC 8824-2:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 2. Спецификация информационных объектов) †

- ITU-T Recommendation X.682 (2002) | ISO/IEC 8824-3:2002[4] Information technology — Abstract Syntax Notation One (ASN.1): Constraint specification (Рекомендация МСЭ-Т X.682 (2002) | ISO/IEC 8824-3:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 3. Спецификация ограничений) †

- ITU-T Recommendation X.683 (2002) | ISO/IEC 8824-4:2002[5] Information technology — Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications (Рекомендация МСЭ-Т X.683 (2002) | ISO/IEC 8824-4:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 4. Спецификация для параметризации ASN.1) †

- ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002[6] Information technology — ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER), and Distinguished Encoding Rules (DER) (Рекомендация МСЭ-Т X.690 (2002) | ISO/IEC 8825-1:2002 Информационные технологии. Правила кодирования ASN.1. Часть 1. Спецификация основных (BER), канонических (CER) и различительных правил кодирования (DER)) †

- ITU-T Recommendation X.691 (2002) | ISO/IEC 8825-2:2002[7] Information technology — ASN.1 encoding rules: Specification of Packed Encoding Rules (PER) (Рекомендация МСЭ-Т X.691 (2002) | ISO/IEC 8825-2:2002 Информационные технологии. Правила кодирования ASN.1. Часть 2. Спецификация правил уплотненного кодирования (PER))

- ITU-T Recommendation X.692 (2002) | ISO/IEC 8825-3:2002[8] Information technology — ASN.1 encoding rules: Specification of Encoding Control Notation (ECN) (Рекомендация МСЭ-Т X.692 (2002) | ISO/IEC 8825-3:2002 Информационные технологии. Правила кодирования ASN.1. Часть 3. Спецификация нотации контроля кодирования (ECN)) †

- ITU-T Recommendation X.693 (2001) | ISO/IEC 8825-4:2002[9] Information technology — ASN.1 encoding rules: XML Encoding Rules (XER) (Рекомендация МСЭ-Т X.693 (2001) | ISO/IEC 8825-4:2002 Информационные технологии. Правила кодирования ASN.1. Часть 4. Правила кодирования XML (XER)) †

- ITU-T Recommendation X.694 (2004) | ISO/IEC 8825-5:2004[10] Information technology — ASN.1 encoding rules: Mapping W3C XML Schema Definitions into ASN.1 (Рекомендация МСЭ-Т X.694 (2004) | ISO/IEC 8825-5:2004 Информационные технологии. Правила кодирования ASN.1. Часть 5. Определения схемы отображения W3C XML в ASN.1) †

---

[1] Отменен. Действует ISO/IEC 9834-1:2012.

[2] Отменен. Действует ISO/IEC 8824-1:2008.

[3] Отменен. Действует ISO/IEC 8824-2:2008.

[4] Отменен. Действует ISO/IEC 8824-3:2008.

[5] Отменен. Действует ISO/IEC 8824-4:2008.

[6] Отменен. Действует ISO/IEC 8825-1:2008.

[7] Отменен. Действует ISO/IEC 8825-2:2008.

[8] Отменен. Действует ISO/IEC 8825-3:2008.

[9] Отменен. Действует ISO/IEC 8825-4:2008.

[10] Отменен. Действует ISO/IEC 8825-5:2008.

- ITU-T Recommendation X.891 (2005) | ISO/IEC 24824-1:2007, Information technology — Generic applications of ASN.1: Fast infoset (Рекомендация МСЭ-Т X.891 (2005) | ISO/IEC 24824-1:2007 Информационные технологии. Общие правила применения ASN.1: Быстрые команды. Часть 1)

Примечание — Приведен полный перечень рекомендаций и международных стандартов по ASN.1, так как все они могут быть применены в конкретных случаях использования настоящего стандарта. Если в тексте настоящего стандарта нет прямых ссылок на какой-либо документ, то в приведенном списке данный документ помечен символом †.

2.2 Дополнительные ссылки

- W3C SOAP:2003, SOAP Version 1.2 Part 1: Messaging Framework, W3C Recommendation, Copyright © [24 June 2003] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

gost.e-ecolog.ru P:2003, SOAP Версия 1.2 Часть 1: Структуры обмена сообщениями, Рекомендация консорциума W3C, © [24.07.2003] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо) gost.e-ecolog.ru)

- W3C SOAP:2003, SOAP Version 1.2 Part 2: Adjuncts, W3C Recommendation, Copyright © [24 June 2003] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

gost.e-ecolog.ru P:2003, SOAP Версия 1.2 Часть 2: Дополнения, Рекомендация консорциума W3C, © [24.06.2003] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), gost.e-ecolog.ru)

- W3C XML 1.0:2004, Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation, Copyright © [4 February 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

gost.e-ecolog.ru XML 1.0:2004, Расширяемый язык разметки (XML) 1.0 (третья редакция), Рекомендация консорциума W3C, © [4.02.2004] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), gost.e-ecolog.ru)

- W3C XML Information Set:2004, XML Information Set (Second Edition), W3C Recommendation, Copyright © [04 February 2004] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

gost.e-ecolog.ru формационный набор XML:2004, Информационный набор XML (вторая редакция), Рекомендация консорциума W3C, © [04.02.2004] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), http://www.w3.org/TR/2004/REC-xml-infoset-20040204/)

- W3C XML Namespaces 1.0:1999, Namespaces in XML, W3C Recommendation, Copyright © [14 January 1999] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

gost.e-ecolog.ru ства имен XML 1.0:1999, Пространства имен в XML, Рекомендация консорциума W3C, © [14.01.1999] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), http://www.w3.org/TR/1999/REC-xm-lnames-19990114/)

- W3C XML Schema:2001, XML Schema Part 1: Structures, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

(XML-схема:2001, XML-схема Часть 1: Структуры, Рекомендация консорциума W3C, © [2.05.2001] Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/)

- W3C XML Schema:2001, XML Schema Part 2: Datatypes, W3C Recommendation, Copyright © [2 May 2001] World Wide Web Consortium (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University), gost.e-ecolog.ru

(XML-схема:2001, XML-схема Часть 2: Типы данных, Рекомендация консорциума W3C, © [2.05.2001]

Консорциум Всемирной паутины, (Массачусетский технологический институт, Национальный институт исследований в области компьютерной обработки данных и автоматики, Университет Кэйо), gost.e-ecolog.ru w3.org/TR/2001/REC-xmlschema-2-20010502/)

Примечание — При упоминании схемы W3C XML в настоящем стандарте имеется в виду первая и вторая части схем W3C XML.

- IETF RFC 2045 (1996), Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies (IETF RFC 2045 (1996), (Многоцелевые расширения интернет-почты (MIME) Часть 1: Формат текста интернет-сообщения)
- IETF RFC 2616 (1999), Hypertext Transfer Protocol — HTTP/1.1 (IETF RFC 2616 (1999), Протокол передачи гипертекста — HTTP/1.1)

3 Термины и определения

В настоящем стандарте применены следующие термины по международным стандартам и документам:

3.1 Заимствованные термины

3.1.1 В настоящем стандарте применены следующие термины по ISO/IEC 8824-1:

a) абстрактное значение;
b) модуль;
c) идентификатор объекта;
d) условный идентификатор объекта;
e) тип.

3.1.2 В настоящем стандарте применены следующие термины по Рекомендации W3C XML-схема:

a) **complex type definition**;
b) **element declaration**;
c) схема;
d) компонент схемы;
e) **simple type definition**.

3.1.3 В настоящем стандарте применены следующие термины по Рекомендации W3C Инфо-набор XML:

a) информационный элемент abstract;
b) информационный элемент character;
c) информационный элемент element;
d) информационный элемент;
e) информационный элемент namespace (пространство имен);
f) свойство (информационного элемента).

3.1.4 В настоящем стандарте применены следующие термины по Рекомендации W3C SOAP Часть 1, 1.5.1:

a) простой протокол доступа к объекту (SOAP);
b) SOAP привязка;
c) SOAP шаблон обмена сообщениями;
d) SOAP узел.

3.1.5 В настоящем стандарте применены следующие термины по ISO/IEC 24824-1:

a) Base64;
b) документ быстрого инфо-набора;
c) инфо-набор XML.

3.2 Дополнительные термины

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.2.1 **интерфейсная привязка ASN.1 SOAP** (ASN.1 SOAP interface binding): Конкретный интерфейс описания сервиса, определяющий семантику быстрых веб-сервисов, которые должны быть предоставлены посредством обмена ASN.1 SOAP сообщениями.

3.2.2 **конечная точка ASN.1 SOAP** (ASN.1 SOAP endpoint): Сетевая папка быстрых веб-сервисов, указанная в описании сервиса.

3.2.3 **блок заголовка ASN.1 SOAP** (ASN.1 SOAP header block). Значение типа `HeaderBlock` (см. Приложение А).

3.2.4 **ASN.1 SOAP HTTP привязка** (ASN.1 SOAP HTTP binding): Привязка SOAP к HTTP для передачи ASN.1 SOAP сообщений.

3.2.5 **ASN.1 SOAP сообщение** (ASN.1 SOAP message): Значение типа `Envelope`, взятое из W3C SOAP сообщения (см. раздел 8).

3.2.6 **встроенное в ASN.1 закодированное значение** (embedded ASN.1 encoded value): Абстрактное значение типа ASN.1, кодирование которого объявлено в сообщении W3C SOAP в строке Base64.

3.2.7 **встроенный документ быстрого инфо-набора (embedded fast infoset document)**. Элемент информации, включенный в ASN.1 SOAP сообщение и закодированный как документ быстрого инфо-набора.

3.2.8 **клиент быстрого веб-сервиса** (fast-enabled web service client): Узел SOAP, который может посылать запросы и получать ответы, используя как ASN.1 SOAP сообщения, так и XML SOAP сообщения.

3.2.9 **SOAP сообщение быстрого инфо-набора** (fast infoset SOAP message): W3C SOAP сообщение, сериализованное в виде документа быстрого инфо-набора.

3.2.10 **быстрые веб-сервисы** (fast web services): Сервисы, предоставляемые при помощи обмена ASN.1 SOAP сообщениями.

3.2.11 **описание сервиса** (service description): Набор документов, описывающих интерфейс и семантику веб-сервиса.

3.2.12 **блок заголовка W3C SOAP** (W3C SOAP header block). Блок заголовка SOAP, описанный в W3C SOAP Часть 1, 1.5.2.

3.2.13 **W3C SOAP сообщение** (W3C SOAP message): SOAP сообщение, описанное в W3C SOAP Часть 1, 1.5.2.

3.2.14 **пространство имен W3C SOAP** (W3C SOAP namespace): Пространство имен «**w3.org/2003/05/soap-envelope**» (см. W3C SOAP Часть 1, 1.1).

gost.e-ecolog.ru

3.2.15 **веб-сервисы XML** (XML web services): Сервисы, предоставляемые при помощи обмена XML SOAP сообщениями.

3.2.16 **XML SOAP сообщение** (XML SOAP message): W3C SOAP сообщение или сообщение, описанное любой предыдущей или последующей версией SOAP, выпускаемой серийно как документ XML.

4 Сокращения

В настоящем стандарте применены следующие сокращения:

AII — Attribute Information Item (информационный элемент attribute) (см. W3C Инфо-набор XML, п. 2.3);

ASN.1 — Abstract Syntax Notation One (абстрактная синтаксическая нотация версии 1);

CII — Character Information Item (информационный элемент character) (см. W3C Инфо-набор XML, п. 2.6);

EII — Element Information Item (информационный элемент element) (см. Рекомендации W3C Инфо-набор XML, п. 2.2);

HTTP — HyperText Transfer Protocol (протокол передачи гипертекста) (см. IETF RFC 2616);

MIME — Multipurpose Internet Mail Extensions (многоцелевые расширения интернет-почты);

NII — Namespace Information Item (информационный элемент namespace) (см. W3C Инфо-набор XML, п. 2.11);

PER — Packed Encoding Rules of ASN.1 (правила пакетного кодирования ASN.1);

RPC — Remote Procedure Call (удаленный вызов процедур);

URI — Uniform Resource Identifier (унифицированный идентификатор ресурса);

WSDL — Web Services Description Language (язык описания веб-сервисов);

XML — eXtensible Markup Language (расширяемый язык разметки);

XSD — W3C XML Schema (W3C XML схема).

5 Нотация

5.1 В настоящем стандарте используют нотацию ASN.1 по ISO/IEC 8824-1.

5.2 В настоящем стандарте **`полужирный шрифт Courier New`** используют для нотации ASN.1

5.3 Для следующих нотаций используют **полужирный шрифт Arial**.

a) синтаксис XML;

b) имена EII и AII;

c) поля заголовков HTTP и параметры полей заголовков HTTP.

5.4 Имена свойств информационных элементов пишут **полужирным шрифтом Arial** и заключают в квадратные скобки (например, свойство **[children]**).

5.5 MIME медиа типы и URI пишут **полужирным шрифтом Arial** и заключают в кавычки (например, URI «gost.e-ecolog.ru»).

6 Обработка ASN.1 SOAP сообщений

6.1 Сообщения ASN.1 SOAP — абстрактные значения типа **`Envelope`**, определенного в модуле ASN.1 **`ASN1SOAP`** (см. приложение A). Абстрактные значения типа **`Envelope`** семантически эквивалентны экземплярам инфо-набора XML, определены в W3C SOAP Часть 1, пункт 5 (называемые инфо-набором W3C SOAP сообщений).

Примечание — Тип **`Envelope`** включает оптимальное двоичное кодирование W3C SOAP сообщения инфо-набора.

6.2 ASN.1 SOAP сообщения могут использоваться как вместе с описаниями веб-сервиса, так и независимо от любого описания веб-сервиса. Для описания веб-сервиса XML SOAP сообщения не требуется изменений для обеспечения описания быстрых веб-сервисов для сообщений SOAP ASN.1 (см. приложение E).

6.3 Модель обработки SOAP, модель расширяемости и обязательная модель (см. W3C Часть 1 SOAP, пункты 2, 3, и 4) должны быть применены SOAP узлом к абстрактным значениям типа **`Envelope`** посредством отображения, определенного в 6.4 между компонентами типа **`Envelope`** и единицами информации W3C SOAP сообщения инфо-набора.

6.4 Применение этих моделей SOAP к абстрактным значениям типа **`Envelope`** возможно в результате следующих концептуальных шагов:

a) абстрактные значения компонентов типа **`Envelope`** (ASN.1 SOAP сообщение) отображаются в виде единицы информации W3C SOAP сообщения инфо-набора, как определено в разделе 7 и таблице 1;

b) модели SOAP, применяемые к этому инфо-набору (см. W3C Часть 1 SOAP, пункты 2, 3, и 4), обычно производят новое W3C SOAP сообщение инфо-набора, которое приспосабливает Часть 5 W3C SOAP и вводит ограничения, описанные в 6.6, и

c) единицы информации нового W3C SOAP сообщения инфо-набора отображаются обратно на абстрактные значения компонентов типа **`Envelope`**, как определено в разделе 8 и таблице 1, обычно производя новое абстрактное значение для типа **`Envelope`** (новое ASN.1 SOAP сообщение).

Примечание — Эти три шага только концептуальны. Нет никакого требования, чтобы фактически формировать описание W3C SOAP сообщения инфо-набора. И W3C SOAP сообщение инфо-набора и ASN.1 SOAP сообщение — абстрактные значения, не зависимые от любой сериализации или кодирования, используемого для их представления в компьютерной системе или для передачи между системами.

6.5 Приложение моделей SOAP к W3C SOAP сообщениям инфо-набора (см. 6.4, b)) должно включать расширенную обработку встроенного в ASN.1 закодированного значения, как определено в пункте 9.

6.6 Следующие ограничения применяются к W3C SOAP сообщению инфо-набора в результате преобразования, упомянутого в 6.4, b):

a) AII не должны присутствовать среди элементов свойства **[attributes] Body** EII и **Detail** EII, и

b) не более чем один EII должен присутствовать среди элементов свойства **[children] Body** EII и **Detail** EII.

6.7 Компонент типа **`Envelope`** (в любой глубине до наличия значения типа **`Content`**) должен быть отображен на единицу информации (или наоборот), как определено в таблице 1. В таблице 1 в графе 1 перечислены компоненты типа **`Envelope`**, в графе 2 дана ссылка на структурные элементы Части 1 W3C SOAP, где определена(ы) семантически эквивалентная(ые) единица(ы) информации.

В графе 3 перечислены структурные элементы настоящего стандарта, которые определяют отображение из компонента в семантически эквивалентную единицу информации. В графе 4 перечислены структурные элементы настоящего стандарта, которые определяют отображение из единицы информации в компонент.

Таблица 1 — Отображение зависимости между компонентами типа **Envelope** и единицами информации W3C SOAP сообщения инфо-набора

ASN.1 модуль для ASN.1 SOAP	Ссылка на W3C SOAP Часть 1	Отображение из ASN.1	Отображение в ASN.1
`Envelope ::= SEQUENCE {`	5.1	Раздел 7	Раздел 8
`header Header,`	5.2	Пункт 7.1.3	Пункт 8.1.2

`body-or-fault CHOICE {`
`body Body,`
`fault Fault`
	5.3, 5.4	Пункты 7.1.4 и 7.1.5	Пункты 8.1.3 и 8.1.4

`}` `}`			
`Header ::= SEQUENCE OF HeaderBlock`	5.2	Подраздел 7.2	Подраздел 8.2
`HeaderBlock ::= SEQUENCE {`	5.2.1	Пункт 7.2.2	Пункт 8.2.2
`mustUnderstand BOOLEAN OPTIONAL,`	5.2.2	Подпункт 7.2.2.1	Подпункт 8.2.2.1
`relay` `BOOLEAN OPTIONAL`	5.2.3	Подпункт 7.2.2.2	Подпункт 8.2.2.2

`role`
`XSD.AnyURI`
 	
`DEFAULT ultimateReceiver,`
	5.2.4	Подпункт 7.2.2.3	Подпункт 8.2.2.3

`content Content`		Подраздел 7.2	Подраздел 8.2
`}`			
`Body ::= SEQUENCE {`	5.3	Подраздел 7.3	Подраздел 8.3
`content Content OPTIONAL`	5.3	Пункт 7.3.2	Пункт 8.3.2
`}`			
`Fault :: = SEQUENCE {`	5.4	Подраздел 7.4	Подраздел 8.4
`code Code,`	5.4.1	Подпункт 7.4.1.2	Подпункт 8.4.1.2
`reason SEQUENCE SIZE(1..MAX) OF Text,`	5.4.2	Подпункт 7.4.1.3	Подпункт 8.4.1.3
`node XSD.AnyURI OPTIONAL,`	5.4.3	Подпункт 7.4.1.4	Подпункт 8.4.1.4
`role XSD.AnyURI OPTIONAL,`	5.4.4	Подпункт 7.4.1.5	Подпункт 8.4.1.5
`detail Content`	5.4.5	Подпункт 7.4.1.6	Подпункт 8.4.1.6
`}`			
`Code ::= SEQUENCE {`	5.4.1	Пункт 7.4.2	Пункт 8.4.2
`value Value,`	5.4.1.1	Подпункт 7.4.2.2	Подпункт 8.4.2.2
`subcodes SEQUENCE OF XSD.QName`	5.4.1.2, 5.4.1.3	Подпункт 7.4.2.3 и 7.4.2.4	Подпункт 8.4.2.3 и 8.4.2.4
`}`			

Окончание таблицы 1

ASN.1 модуль для ASN.1 SOAP	Ссылка на W3C SOAP Часть 1	Отображение из ASN.1	Отображение в ASN.1

`Value ::= ENUMERATED {`
`  versionMismatch,`
`  mustUnderstand,`
`  dataEncodingUnknown,`
`  sender,`
	5.4.1.1, 5.4.8	Пункт 7.4.3	Пункт 8.4.3

`  receiver`			
`}`			
`Text ::= SEQUENCE {`	5.4.2.1	Пункт 7.4.4	Пункт 8.4.4
`  lang XSD.Language,`		Подпункт 7.4.4.2	Подпункт 8.4.4.2
`  text UTF8String`		Подпункт 7.4.4.3	Подпункт 8.4.4.3
`}`			
`Content ::= ...`	N/A	Подраздел 7.5	Подраздел 8.5

7 Отображение компонентов типа Envelope в единицах информации

7.1 Общие положения

7.1.1 **Envelope** EII должен быть сформирован из значения типа `Envelope`.

7.1.2 Уникальное свойство **[prefix]** NII со свойством **[namespace name]**, равное имени W3C SOAP диапазона имен среди элементов свойства **[in-scope namespaces] Envelope** EII, должно быть сформировано в соответствии с его значением, выбранным SOAP узлом.

Примечания

1 Префикс «**env**» традиционно используется в W3C SOAP, Часть 1, 1.1, но может использоваться любой префикс.

2 Все EII и AII, определенные в SOAP, имеют свойство **[namespace name]**, равное имени W3C SOAP диапазона имен, как определено в W3C SOAP Часть 1, 1.1.

7.1.3 Значение компонента `header` должно быть отображено, как определено в 7.2.

7.1.4 Если у значения компонента `body-or-fault` будет существующая альтернатива `body`, то эта альтернатива должна быть отображена в **Body** EII, как определено в 7.3.

7.1.5 Если у значения компонента `body-or-fault` будет существующая альтернатива `fault`, то **Body** EII должен быть сформирован, а альтернатива должна быть отображена в **Fault** EII, как определено в 7.4.

Примечание — У W3C SOAP сообщения, содержащего информацию об отказе, может быть только один **Fault** EII как дочерний элемент **Body** EII (и не может быть никаких других дочерних EII). Схема ASN.1 отражает эти ограничения, обеспечивая отдельное содержимое и альтернативы отказа для выбора `body-or-fault`.

7.2 Отображение типа `Header`

7.2.1 **Header** EII должен быть сформирован из значения типа `Header`. Если тип `Header` будет содержать одно или более возникновений `HeaderBlock`, то каждое возникновение `HeaderBlock` должно быть отображено в определенном порядке по отношению к дочернему EII **Header** EII, как определено в 7.2.2. Если не будет никаких возникновений `HeaderBlock`, то **Header** EII не должен быть сформирован.

7.2.2 Значение компонента `content` должно быть отображено в блоке заголовка W3C SOAP, как определено в 7.5. Дополнительные AII среди элементов свойства **[attributes]** EII должны быть сформированы, как определено в 7.2.2.1—7.2.2.3.

7.2.2.1 **mustUnderstand** AII должен быть сформирован из значения компонента `mustUnderstand` при условии, что значение присутствует и не является `FALSE`, а свойство **[normalized value]** в **mustUnderstand** AII должно быть «**1**».

7.2.2.2 **relay** AII должен быть сформирован из значения компонента `relay` при условии, что значение присутствует и не является `FALSE`, а свойство **[normalized value]** в **relay** AII должно быть «**1**».

7.2.2.3 **role** AII должен быть сформирован из значения компонента `role` при условии, что это значение является отличным от `ultimateReceiver`, а свойство **[normalized value] role** AII должно быть значением символьной строки компонента `role`.

7.3 Отображение типа `Body`

7.3.1 **Body** EII должен быть сформирован из значения типа `Body`.

7.3.2 Значение компонента `content` должно быть отображено, как определено в 7.5.

7.4 Отображение типа `Fault`

7.4.1 Общие положения

7.4.1.1 **Fault** EII должен быть сформирован из значения типа `Fault`.

7.4.1.2 Значение компонента `code` должно быть отображено, как определено в 7.4.2.

7.4.1.3 **Reason** EII должен быть сформирован из значения компонента `reason`. Каждое появление `Text` в последовательности должно быть отображено в определенном порядке по отношению к дочернему **Text** EII **Reason** EII, как определено в 7.4.4.

Примечание — Рекомендуется, чтобы у всех возникновений `Text` в последовательности были уникальные компонентные значения **lang** (см. W3C SOAP Часть 1, 5.4.2).

7.4.1.4 **Node** EII должен быть сформирован из значения компонента `node` и должен иметь, как и его **child** CII, символы значения символьной строки компонента `node`.

7.4.1.5 **Role** EII должен быть сформирован из значения компонента `role` и должен иметь, как и его **child** CII, символы значения символьной строки компонента `role`.

7.4.1.6 **Detail** EII должен быть сформирован из значения `detail`, как определено в 7.5.

7.4.2 Отображение типа `Code`

7.4.2.1 **Code** EII должен быть сформирован из значения типа `Code`.

7.4.2.2 Значение компонента `value` должно быть отображено, как определено в 7.4.3, чтобы обеспечить первый (или единственный, если компонент `subcodes` будет пуст) дочерний EII **Code** EII.

7.4.2.3 Первый `XSD.QName` компонента `subcodes` должен формировать:

a) **Subcode** EII как второй дочерний EII **Code** EII;

b) **Value** EII (дочерний элемент **Subcode** EII, сформированный из значения первого возникновения `XSD.QName`, также определенного в 7.4.2.5—7.4.2.6) как первый дочерний EII **Subcode** EII, сформированного в a).

7.4.2.4 Каждый из следующих `XSD.QName` компонентов `subcodes` должен формировать:

a) **Subcode** EII как второй дочерний элемент **Subcode** EII, который был сформирован из значения предыдущего появления `XSD.QName`;

b) **Value** EII (дочерний элемент **Subcode** EII и сформированный из значения текущего появления `XSD.QName`, также определенного в 7.4.2.5—7.4.2.6) как первый дочерний EII **Subcode** EII, сформированного в a).

Примечание — У каждого **Subcode** EII есть последующий (**Subcode**) дочерний EII только в том случае, если есть последующий XSD.QName в `subcodes`.

7.4.2.5 **Value** EII (дочерний элемент **Subcode** EII) должен быть сформирован из появления `XSD.QName` (с представлением компонента `uri`) с:

a) NII среди элементов свойства **[in-scope namespaces]** со свойством **[namespace name]**, равным значению компонента `uri`, и свойства **[prefix]**, выбранного узлом SOAP;

b) последовательностью CII, которая является связью:

1) свойства **[prefix]** в a);

2) ДВОЕТОЧИЯ («:»);

3) значения символьной строки компонента `name`.

7.4.2.6 EII `Value` (дочерний элемент **Subcode** EII) должен быть сформирован из значения появлений `XSD.QName` (при отсутствии его компонента `uri`) с последовательностью дочерних CII, которая является значением компонента `name`.

7.4.3 Отображение типа `Value`

Value EII (дочерний элемент **Code** EII) должен быть сформирован из значения типа `Value` с последовательностью CII, которая в свою очередь формируется из значения перечисления в виде символов символьной строки, которая является связью:

a) свойства **[prefix]**, как определено в 7.1.2;
b) ДВОЕТОЧИЯ («:») и
c) локального имени, как определено в таблице 2.

Таблица 2 — Отображение локального имени типа `Value`

Значение перечисления типа **Value**	Локальное имя
`versionMismatch`	**VersionMismatch**
`mustUnderstand`	**MustUnderstand**
`dataEncodingUnknown`	**DataEncodingUnknown**
`sender`	**Sender**
`receiver`	**Receiver**

7.4.4 Отображение типа `Text`

7.4.4.1 **Text** EII должен быть сформирован из значения типа `Text`.

7.4.4.2 AII должен быть сформирован из компонента `lang` с помощью:

a) свойства **[local name]** «**lang**»;
b) свойства **[namespace name]** «gost.e-ecolog.ru »;
c) свойства **[prefix]** «**xml**»;
d) свойства **[normalized value]**, равного значению компонента `lang`.

7.4.4.3 Последовательность **child** CII из **Text** EII должна быть значением символьной строки компонента `text`.

7.5 Отображение типа `Content`

7.5.1 Общие положения

7.5.1.1 Содержимое EII должно быть сформировано из значения типа `Content` согласно 7.5.2, 7.5.3 или 7.5.4 для отображения документов быстрого инфо-набора, закодированных значений ASN.1 и «непонятых» блоков заголовков ASN.1 SOAP (см. 7.5.4) соответственно в инфо-набор XML.

7.5.1.2 Если присутствует альтернатива `fast-infoset-document` типа `Content`, то применяется 7.5.2.

7.5.1.3 Если присутствует альтернатива `encoded-value` типа `Content` и `encoded-value.id` не равен значению `notUnderstoodIdentifier`, то применяется 7.5.3.

7.5.1.4 Если присутствует альтернатива `encoded-value` типа `Content` и `encoded-value.id` равен значению `notUnderstoodIdentifier`, то применяется 7.5.4.

7.5.2 Содержание документа быстрого инфо-набора

7.5.2.1 Октеты компонента `fast-infoset-document` будут документом быстрого инфо-набора, определенным в МСЭ-Т X.891 | ISO/IEC 24824-1.

7.5.2.2 Содержимое EII должно быть сформировано путем следующих действий:

a) декодирования октетов `fast-infoset-content` для формирования инфо-набора XML, который является корневым EII (как определено в МСЭ-Т X.891 | ISO/IEC 24824-1), и
b) применения 7.5.2.3 к корневому EII.

7.5.2.3 Следующие AII (среди элементов свойства **[attributes]** корневого EII), если они сформированы из отображения значения быстрого инфо-набора к корневому EII, должны быть удалены из свойства **[attributes]** корневого EII:

a) **role** AII;
b) **mustUnderstand** AII;
c) **relay** AII.

Примечание — **role** AII, **mustUnderstand** AII и **relay** AII отображаются соответственно из компонентов `role`, `mustUnderstand` и `relay` типа `HeaderBlock` (см. 7.2.2). Удаление этих AII из свойства **[attributes]** корневого EII гарантирует, что только компоненты `HeaderBlock` будут использоваться для обработки блока заголовка W3C SOAP узлом SOAP.

7.5.3 Встроенное в ASN.1 содержимое закодированного значения

7.5.3.1 **encodingStyle** AII (см. W3C SOAP Часть 1, 5.1.1) среди членов свойства **[attributes]** содержимого EII должно быть сформировано со свойством **[normalized value]**:

«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper».

7.5.3.2 Если компонент `encoded-value.id` имеет альтернативу `qName`, то свойства **[namespace name]** и **[local name]** содержимого EII должны быть установлены из `qName`.

7.5.3.3 Если компонент `encoded-value.id` имеет альтернативу `roid`, то содержимое EII должно быть сформировано:

a) со свойством **[local name]** «roid»;

b) со свойством **[namespace name]**:

`«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope»;`

c) с **roid** AII среди членов свойства **[attributes]**, как указано в 7.5.3.4.

7.5.3.4 AII среди членов свойства **[attributes]** содержимого EII должен быть сформирован из значения типа `Content` (если компонент `encoded-value.id` имеет альтернативу `roid`) с помощью:

a) свойства **[local name]** «roid»;

b) свойства **[namespace name]**: «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**»;

c) свойства **[specified]** «true»;

d) свойства **[normalized value]**, которое должно быть значением компонента `roid`, закодированного как «XMLRelativeOIDValue», используя только «XMLNumberForm» (см. МСЭ-Т X.680 | ISO/IEC 8824-1, п. 32).

7.5.3.5 Последовательность дочерних CII содержимого EII должна быть сформирована из кодирования Base64 строки октета (как указано в IETF RFC 2045, 6.8), что является значением компонента `encoded-value.encoding`.

7.5.3.6 Компонент `schema-identifier`, если он присутствуют, должен быть проигнорирован и не отображен.

7.5.4 Непонятое содержимое блока заголовка W3C SOAP

7.5.4.1 `notUnderstoodIdentifier` будет определять `NotUnderstood` тип ASN.1, значение которого кодируется, используя Basic Aligned PER к октету строки, которая является значением компонента `encoded-value.encoding`.

7.5.4.2 Значение типа `NotUnderstood` должно быть сформировано путем декодирования, используя Basic Aligned PER, октеты компонента `encoded-value.encoding`.

7.5.4.3 **NotUnderstood** EII (см. W3C SOAP Часть 1, 5.4.8.1) должно быть сформировано как содержимое EII с помощью:

a) NII среди членов его свойства **[in-scope namespaces]** со свойством **[namespace name]**, равным значению компонента `NotUnderstood.uri`, и с уникальным свойством **[prefix]**, выбранным SOAP узлом, и

b) **qname** AII (см. W3C SOAP Часть 1, 5.4.8.2) со свойством **[normalized value]**, которое является объединением свойства **[prefix]** в a), двоеточием («:») и значением символьной строки компонента `NotUnderstood.name`.

8 Отображение инфо-наборов W3C SOAP сообщений в абстрактные значения типа `Envelope`

8.1 Общие положения

8.1.1 Значение типа `Envelope` должно быть сформировано из **Envelope** EII.

8.1.2 **Header** EII (при наличии) должен быть отображен в компонент `header`, как указано в 8.2.

8.1.3 Если **Body** EII не содержит **Fault** EII как единственный дочерний EII, то должно быть сформировано значение компонента `body-or-fault` с альтернативной `body` и **Body** EII должен быть отображен в альтернативе `body`, как указано в 8.3.

8.1.4 Если **Body** EII содержит **Fault** EII как единственный дочерний EII, то должно быть сформировано значение компонента `body-or-fault` с альтернативной `fault` и **Fault** EII должен быть отображен в альтернативе `fault`, как указано в 8.4.

8.2 Отображение Header EII

8.2.1 Значение типа `Header` должно быть сформировано из **Header** EII, и каждый дочерний EII (блок заголовка W3C SOAP) должен быть отображен в порядке появления `Content` в последовательности, как указано в 8.2.2.

8.2.2 Значение типа **HeaderBlock** должно быть сформировано из блока заголовка W3C SOAP, который должен быть отображен в значение компонента **content**, как указано в 8.5. Дополнительные компоненты типа **HeaderBlock** должны быть сформированы, как указано в 8.2.2.1—8.2.2.3.

8.2.2.1 Значение компонента **mustUnderstand** должно быть сформировано из **mustUnderstand** AII (если имеется) и должно быть **TRUE**, если свойством **[normalized value] mustUnderstand** AII является «**1**». В противном случае компонент должен отсутствовать.

8.2.2.2 Значение компонента **relay** должно быть сформировано из **relay** AII (если имеется) и должно быть **TRUE**, если свойством **[normalized value] relay** AII является «**1**». В противном случае компонент должен отсутствовать.

8.2.2.3 Значение компонента **role** должно быть сформировано из **role** AII (если имеется) и должно быть свойством **[normalized value] role** AII.

8.3 Отображение Body EII

8.3.1 Значение типа **Body** должно быть сформировано из **Body** EII.

8.3.2 Дочерний EII **Body** EII (если имеется) должен быть отображен в значение компонента **content**, как указано в 8.5.

8.4 Отображение Fault EII

8.4.1 Общие положения

8.4.1.1 Значение типа **Fault** должно быть сформировано из **Fault** EII.

8.4.1.2 **Code** EII должен быть отображен в значение компонента **code**, как указано в 8.4.2.

8.4.1.3 Значение компонента **reason** должно быть сформировано из **Reason** EII. Каждый дочерний **Text** EII должен быть отображен в порядке появления **Text** в последовательности, как указано в 8.4.4.

8.4.1.4 Значение компонента **node** должно быть сформировано из **Node** EII (если имеется) и должно иметь в качестве своего значения буквенную последовательность дочерних CII из **Node** EII.

8.4.1.5 Значение компонента **role** должно быть сформировано из **Role** EII (если имеется) и должно иметь в качестве своего значения буквенную последовательность дочерних CII **Role** EII.

8.4.1.6 Значение компонента **detail** должно быть сформировано из **Detail** EII (если имеется), и дочерний EII должен быть отображен, как указано в 8.5.

8.4.2 Отображение Code EII

8.4.2.1 Значение типа **Code** должно быть сформировано из **Code** EII.

8.4.2.2 **Value** EII (дочерний элемент **Code** EII) должен быть отображен в значение компонента **value**, как указано в 8.4.3.

8.4.2.3 Первый **Subcode** EII (если имеется) должен сформировать значение типа **XSD.QName** как первый элемент компонента **subcodes**. Значение должно быть сформировано из первого дочернего **Value** EII, как указано в 8.4.2.5 и 8.4.2.6.

8.4.2.4 Второй дочерний **Subcode** EII (если имеется) каждого **Subcode** EII должен сформировать значение типа **XSD.QName** в качестве следующего элемента компонента **subcodes**. Значение должно быть сформировано из первого дочернего **Value** EII второго дочернего **Subcode** EII, как указано в 8.4.2.5 или 8.4.2.6.

8.4.2.5 Значение типа **XSD.QName** должно быть сформировано из **Value** EII (дочернего **Subcode** EII), имеющего последовательность дочерних CII, которые являются объединением префикса (P, например), двоеточия («:») и локального имени со:

a) значением компонента **uri**, которое является свойством **[namespace name]** NII среди членов свойства **[in-scope namespaces] Value** EII (дочернего **Subcode** EII) со свойством **[prefix]** P;

b) значением компонента **name**, которое является локальным именем.

8.4.2.6 Значение типа **XSD.QName** должно быть сформировано из **Value** EII (дочернего **Subcode** EII), имеющего последовательность дочерних CII, которые не содержат двоеточия («:»), со значением компонента **name**, которое является значением символьной строки последовательности дочерних CII.

8.4.3 Отображение Value EII, являющегося дочерним Code EII

Значение типа **Value** должно быть сформировано из **Value** EII (дочернего **Code** EII) с локальным именем, как указано в таблице 2, и должно быть частью строки последовательности дочерних CII, которые представляет собой объединение из:

a) свойства **[prefix]**, описанного в 7.1.2;

b) двоеточия («:») и

c) локального имени.

8.4.4 Отображение Text EII

8.4.4.1 Значение типа `Text` должно быть сформировано из **Text** EII.

8.4.4.2 Значение компонента `lang` должно быть сформировано из AII со свойством **[local name]** «**lang**» и свойством **[namespace name]** «gost.e-ecolog.ru и должно быть свойством **[normalized value]** AII.

8.4.4.3 Значение компонента `text` должно быть сформировано из **Text** EII и должно быть последовательностью дочерних CII **Text** EII.

8.5 Отображение содержимого EII в значение типа `Content`

8.5.1 Общие положения

8.5.1.1 Значение типа `Content` должно быть сформировано из содержимого EII, как указано в 8.5.2, 8.5.3 или 8.5.4, для отображения из инфо-набора XML в документ быстрого инфо-набора закодированных значений ASN.1 и непонятых блоков заголовков ASN.1 SOAP соответственно.

8.5.1.2 Пункт 8.5.2 применяется, если:

a) **encodingStyle** AII (см. W3C SOAP Часть 1, 5.1.1) не входит в число членов свойства **[attributes]** содержимого EII и содержимое EII не является **NotUnderstood** EII (см. W3C SOAP Часть 1, 5.4.8.1) или

b) **encodingStyle** AII является одним из членов свойства **[attributes]** содержимого EII и **encodingStyle** AII имеет свойство **[normalized value]**, которое не равно **«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper»** (см. 7.5.3.1).

8.5.1.3 Если **encodingStyle** AII (см. W3C SOAP Часть 1, 5.1.1) является одним из членов свойства **[attributes]** содержимого EII и **encodingStyle** AII имеет свойство **[normalized value] «urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper»**, то применяется 8.5.3.

8.5.1.4 Если содержимое EII является **NotUnderstood** EII (см. W3C SOAP Часть 1, 5.4.8.1), то применяется 8.5.4.

Примечание — **encodingStyle** AII не может быть среди членов свойства **[attributes] NotUndestood** EII (см. W3C SOAP Часть 1, 5.4.8.1).

8.5.2 Встроенный документ быстрого инфо-набора

8.5.2.1 Значение типа `Content` должно быть сформировано с альтернативой `fast-infoset-document`.

8.5.2.2 Октеты компонента `fast-infoset-document` станут документом быстрого инфо-набора и должны быть сформированы путем выполнения следующих действий:

a) применения 8.5.2.3 к содержимому EII для формирования корневого EII из инфо-набора XML;

b) кодирования инфо-набора XML как документа быстрого инфо-набора (в соответствии с МСЭ-Т X.891 | ISO/IEC 24824-1).

8.5.2.3 Следующие AII, если они присутствует среди членов свойства **[attributes]** содержимого EII, должны быть удалены из свойства **[children]** EII:

a) **role** AII;

b) **mustUnderstand** AII;

c) **relay** AII.

Примечание — Удаление этих AII из свойства **[attributes]** содержимого EII гарантирует, что только компоненты `HeaderBlock` будут использоваться для обработки W3C SOAP блока заголовка SOAP узлом.

8.5.3 Встроенные в ASN.1 закодированные значения

8.5.3.1 Значение типа `Content` должно быть сформировано с альтернативой `encoded-value`.

8.5.3.2 Если **roid** AII (см. 7.5.3.4) не входит в число членов свойства **[attributes]** содержимого EII, то:

a) `encoded-value.id` должно быть сформировано с альтернативой `qName` и

b) его значение должно быть установлено из свойства **[local name]** и свойства **[namespace name]** содержимого EII.

8.5.3.3 Если **roid** AII (см. 7.5.3.4) является одним из членов свойства **[attributes]** содержимого EII, то:

a) должно быть сформировано `encoded-value.id` с альтернативой `roid` и

b) его значение должно быть установлено из свойства **[normalized value] roid** AII, закодированного как «XMLRelativeOIDValue», при этом используя только «XMLNumberForm» (см. МСЭ-Т X.680 | ISO/IEC 8824-1, п. 32).

Примечание — Условный идентификатор объекта может быть использован вместо классифицированного имени при наличии ограничений на размер ASN.1 SOAP сообщения.

8.5.3.4 Значение компонента **encoded-value.encoding** должно быть сформировано из последовательности дочерних CII содержимого EII, которое является кодированием Base64 строки октета, и должно быть строкой октета.

8.5.3.5 Компонент **schema-identifier** не должен отображаться и должен быть исключен.

8.5.4 Непонятое содержимое блока заголовка W3C SOAP

8.5.4.1 **encoded-value.id** должно быть сформировано с альтернативой **qName**, и его значение должно быть установлено из свойства **[local name]** и свойства **[namespace name] NotUnderstood** EII.

8.5.4.2 Значение типа **NotUnderstood** должно быть сформировано из **NotUnderstood** EII со свойством **[normalized value] qname** AII, которое является объединением префикса (P, например), двоеточия («:») и локального имени со:

a) значением компонента **uri**, которое является свойством **[namespace name]** NII среди членов свойства **[in-scope namespaces] NotUnderstood** EII со свойством **[prefix]** P, и

b) значением компонента **name**, которое является локальным именем.

8.5.4.3 Значение типа **NotUnderstood** должно быть закодировано с использованием Basic Aligned PER в строку октета, которая должна быть значением компонента **encoded-value.encoding**.

9 Расширенная SOAP обработка встроенных в ASN.1 закодированных значений

9.1 Общие положения

9.1.1 Обработка, описанная в следующих пунктах, расширяет обработку W3C SOAP сообщений, описанных в W3C SOAP Часть 1, что позволяет SOAP узлу делать дополнительные преобразования содержимого EII, которые были отображены из ASN.1 SOAP сообщения.

Примечание — Расширенная обработка требуется, поскольку содержимое EII, как последовательность дочерних CII, будет содержать встроенное в ASN.1 закодированное значение, которое является скрытым для SOAP узла, если дальнейшая обработка выполняется для формирования значения ASN.1 из последовательности дочерних CII.

9.1.2 Содержимое EII должно быть дочерним EII от **Body** EII, **Header** EII (блоков заголовков W3C SOAP), **Detail** EII.

Примечание — Содержимое EII, как правило, обрабатывается следующим образом:

a) конечный SOAP получатель обрабатывает дочерний **Body** EII или дочерний **Detail** EII и любые целевые блоки заголовков W3C SOAP;

b) SOAP посредник обрабатывает любые целевые блоки заголовков W3C SOAP;

c) SOAP узел обрабатывает целевые блоки заголовков W3C SOAP, как следствие a) или b);

d) SOAP узел (такой как активный посредник) обрабатывает элементы информации с помощью дополнительной обработки, не описываемой блоками заголовков W3C SOAP.

9.1.3 Содержимое (дочернее) EII будет иметь среди членов свойства **[attributes] encodingStyle** AII со свойством **[normalized value]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper**», как указано в 7.5.3.1.

9.1.4 Применение расширенной обработки содержимого EII должно быть результатом следующих шагов:

a) ASN.1 тип встроенного в ASN.1 закодированного значения определен, как указано в 9.2;

b) значение ASN.1 формируется из определенного встроенного в ASN.1 закодированного значения, учитывая тип ASN.1, как указано в 9.3;

c) сформированное значение ASN.1 обрабатывается SOAP узлом, как правило, производя одно или несколько значений ASN.1 с идентификаторами;

d) полученные ASN.1 значения с идентификаторами вставляются в новое W3C SOAP сообщение инфо-набора (см. 6.4, b)) как встроенные в ASN.1 закодированные значения, как описано в 9.4.

Примечание — Эти четыре шага концептуальны. На самом деле не существует никакого требования для осуществления формирования значения ASN.1 из определенного встроенного в ASN.1 значения, так как не существует никаких требований для реализации формирования представления W3C SOAP сообщения инфо-набора.

9.2 Определение типа ASN.1 встроенного в ASN.1 закодированного значения

9.2.1 Для целей идентификации значение типа **`Identifier`** должно быть сформировано из содержимого EII согласно 9.2.2 и 9.2.3, значение определяет тип ASN.1 встроенного в ASN.1 закодированного значения.

9.2.1.1 Узел обработки SOAP выдаст ошибку, как указано в 9.5, если тип ASN.1 не может быть определен из значения **`Identifier`**.

9.2.1.2 Средства, с помощью которых узел обработки SOAP получает и управляет, набор значений **`Identifier`** и определенных типов ASN.1 не описаны в настоящем стандарте.

Примечание — SOAP узел может получить (частично) набор определенных типов ASN.1 из описания сервиса (см. 13.8).

9.2.2 Если содержимое EII имеет свойство **[local name]** «roid» и свойство **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**», и **roid** AII (см. 7.5.3.4) является одним из членов свойства **[attributes]**, то:

a) значение типа **`Identifier`** должно быть сформировано с альтернативой **`roid`** и

b) его значение должно быть установлено из свойства **[normalized value] roid** AII, которое кодируется как «XMLRelativeOIDValue», при этои используя только «XMLNumberForm» (см. МСЭ-Т X.680 | ISO/IEC 8824-1, п. 32).

9.2.3 Если **roid** AII (см. 7.5.3.4) не входит в число членов свойства **[attributes]** содержимого EII, то:

a) значение типа **`Identifier`** должно быть сформировано с альтернативой **`qName`** и

b) его значение должно быть установлено из свойств **[local name]** и **[namespace name]** содержимого EII.

9.3 Формирование значений ASN.1 из определенных встроенных в ASN.1 закодированных значений

Значение ASN.1 должно быть сформировано из дочерних CII содержимого EII (встроенное в ASN.1 закодированное значение). Эти дочерние CII являются кодированием Base64 строки октета (как указано в IETF RFC 2045, 6.8), состоящей из Basic Aligned PER кодирования значения ASN.1, тип ASN.1 которого был определен, как указано в 9.2.

9.4 Вставка значений ASN.1 (с идентификатором) в W3C SOAP сообщение

9.4.1 Общие положения

9.4.1.1 ASN.1 значение с идентификатором, который является значением типа **`Identifier`**, и возможные дополнительные значения должны быть вставлены как встроенное в ASN.1 закодированное значение сформированного содержимого EII в W3C SOAP сообщении, как указано в следующих подпунктах.

Примечание — Значения типа **`Identifier`** и дополнительные значения могут быть получены из описания сервиса или приложения в SOAP узле.

9.4.1.2 Если значение ASN.1, которое должно быть вставлено как встроенное в ASN.1 закодированное значение сформированного содержимого EII, является дочерним **Header** EII (блока заголовка W3C SOAP), применяется 9.4.2.

9.4.1.3 Если значение ASN.1, которое должно быть вставлено как встроенное в ASN.1 закодированное значение сформированного содержимого EII, является дочерним **Body** EII, применяется 9.4.3.

9.4.1.4 Если значение ASN.1, которое должно быть вставлено как встроенное в ASN.1 закодированное значение сформированного содержимого EII, является дочерним **Detail** EII, применяется 9.4.4.

9.4.2 Вставка как дочерний EII Header EII

9.4.2.1 Содержимое EII должо быть сформировано из значения ASN.1 и значения **`Identifier`**, как указано в 9.4.5. Содержимое EII должно быть блоком заголовка W3C SOAP, который является дочерним **Header** EII. Дополнительные значения (если имеются) влекут за собой вставку AII между членами свойства **[attributes]** содержимого EII, как указано в трех подпунктах ниже.

Примечание — Порядок, в котором SOAP-блок заголовка EII вставляется, зависит от узла обработки SOAP.

9.4.2.2 Дополнительный URI (если имеется), соответствующий семантике **role** AII, должен формировать **role** AII, и его свойство **[normalized value]** должно быть значением символьной строки URI.

9.4.2.3 Дополнительное логическое значение (если имеется), соответствующее семантике **mustUnderstand** AII, должно формировать **mustUnderstand** AII, если логическим значением является `TRUE`, а его свойство **[normalized value]** должно быть «**1**».

9.4.2.4 Дополнительное логическое значение (если имеется), соответствующее семантике **relay** AII, должно формировать **relay** AII, если логическим значением является `TRUE`, а его свойство **[normalized value]** должно быть «**1**».

9.4.3 Вставка как дочерний EII Body EII

Содержимое EII должно быть сформировано из значения ASN.1 и значения `Identifier`, как указано в 9.4.5. Содержимое EII должно быть единственным дочерним содержимым **Body** EII и заменять любой существующий дочерний EII компонента **Body** EII.

9.4.4 Вставка как дочерний EII Detail EII

Содержимое EII должно быть сформировано из значения ASN.1 и значения `Identifier`, как указано в 9.4.5. Содержимое EII должно быть единственным дочерним содержимым **Detail** EII и заменять любой существующий дочерний EII компонента **Detail** EII.

9.4.5 Формирование содержимого EII из значения ASN.1 и значения `Identifier`

9.4.5.1 Содержимое EII должно быть сформировано из значения ASN.1 (должно быть вставлено как встроенное в ASN.1 закодированное значение) и значения `Identifier`, как указано в следующих четырех подпунктах.

9.4.5.2 Если значение типа `Identifier` имеет альтернативу `qName`, то свойства **[namespace name]** и **[local name]** содержимого EII должны быть установлены из `qName`.

9.4.5.3 Если значение типа `Identifier` имеет альтернативу `roid`, то свойство **[local name]** содержимого EII должно быть «roid», а свойство **[namespace name]** должно быть «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**». AII среди членов свойства **[attributes]** содержимого EII должен быть сформирован со:

a) свойством **[local name]** «roid»;

b) свойством **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**»;

c) свойством **[specified]** «true» и

d) свойством **[normalized value]**, которое должно быть значением компонента `roid`, закодированного как «XMLRelativeOIDValue», при этом используя только «XMLNumberForm» (см. МСЭ-Т X.680 | ISO/IEC 8824-1, п. 32).

9.4.5.4 Последовательность дочерних CII (встроенные в ASN.1 закодированные значения) содержимого EII должна быть сформирована из кодирования Base64 строки октета (как указано в IETF RFC 2045, 6.8), состоящей из Basic Aligned PER кодирования значения ASN.1.

9.4.5.5 **encodingStyle** AII (см. W3C SOAP Часть 1, 5.1.1) среди членов свойства **[attributes]** EII должно быть сформировано со свойством **[normalized value] «urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper».**

9.5 Ошибка «неидентифицируемый тип ASN.1»

9.5.1 SOAP узел должен выдать ошибку, если он не может определить тип ASN.1 встроенного в ASN.1 закодированного значения из значения типа `Identifier`, сформированного согласно 9.2. В последующих подпунктах описан инфо-набор для информации об ошибке W3C SOAP инфо-набора сообщения, который является ошибкой.

9.5.2 **Value** EII (дочерний **Code** EII) должен быть сформирован с последовательностью дочерних CII, которые представляет собой объединение следующих символьных строк:

a) свойства **[prefix]**, как описано в 7.1.2,

b) двоеточия («:») и

c) строки «Sender».

9.5.3 Уникальное свойство **[prefix]** NII со свойством **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**» среди членов свойства **[in-scope namespaces] Envelope** EII (или дочернего EII на любой глубине, вплоть до **Value** EII, сформированного в 9.5.4) должно быть сформировано с его значением, выбранным SOAP узлом.

9.5.4 **Subcode** EII (дочерний **Code** EII) должен быть сформирован с помощью одного дочернего **Value** EII.

9.5.4.1 Одиночный дочерний **Value** EII должен иметь последовательность дочерних CII, которые представляют собой объединение следующих символьных строк:

a) свойства **[prefix]**, как описано в 9.5.3;

b) двоеточия («:») и

c) строки «NotIdentified».

Примечание — Такая неидентифицируемая ошибка (W3C SOAP сообщение), сериализованная как документ XML, представлена следующим образом:

```
<env:Envelope xmlns:env=« gost.e-ecolog.ru «>
  <env:Body>
        <env:Fault>
            <env:Code>
                    <env:Value>env:Sender</env:Value>
                    <env:Subcode>
                            <env:Value
xmlns:fws=«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-
            envelope»>fws:NotIdentified</env:Value>
                    </env:Subcode>
            </env:Code>
        </env:Fault>
  </env:Body>
</env:Envelope>
```

10 ASN.1 SOAP HTTP привязка

ASN.1 SOAP HTTP привязка представляет собой модификацию и расширение SOAP HTTP привязки (см. W3C SOAP Часть 2, раздел 7) и обеспечивает привязку W3C SOAP к HTTP для передачи ASN.1 SOAP сообщения, закодированного с использованием Basic Aligned PER.

ASN.1 SOAP HTTP привязка соответствует SOAP протоколу binding framework (см. W3C SOAP Часть 1, п. 4). В приложении D представлен материал о взаимодействии быстрых веб-сервисов и веб-сервисов XML на основе возможностей ASN.1 SOAP HTTP привязки.

10.1 HTTP медиа тип

10.1.1 Для применения настоящего стандарта W3C SOAP Часть 2, 7.1.4 должна быть изменена, как описано в трех следующих пунктах.

10.1.2 Реализация ASN.1 SOAP HTTP привязки должна быть способна отправлять и получать ASN.1 SOAP сообщения с помощью «**application/fastsoap**» медиа типа, параметры и правильное использование которого описаны в B.1.

10.1.3 Реализация ASN.1 SOAP HTTP привязки может также отправлять запросы и ответы с помощью других видов медиа типов, гарантируя, что такие медиа типы идентифицируют W3C SOAP сообщения.

Примечание — Такие W3C SOAP сообщения могут быть, в частности, XML SOAP сообщениями или SOAP сообщениями быстрого инфо-набора.

10.1.4 ASN.1 SOAP HTTP привязка может, когда отправляется запрос, предоставлять поле заголовка HTTP **Accept** (см. IETF RFC 2616, 14.1). Этот заголовок:

a) должен указывать на возможность принять, по крайней мере, «**application/fastsoap**» медиа тип,

b) может дополнительно указывать на готовность принять другие виды медиа типов для передачи W3C SOAP сообщений.

Примечание — Реализация может отправлять XML SOAP сообщение-запрос с полем заголовка HTTP **Accept** «**Accept: application/fastsoap, application/soap+xml**», чтобы указать на способность принимать ASN.1 SOAP сообщения, и в дополнение указать на готовность принимать XML SOAP сообщения (см. пессимистическую стратегию, описанную в D.2).

10.2 Поведение отвечающих SOAP узлов

10.2.1 W3C SOAP Часть 2, 7.5.2 должна быть расширена с помощью двух следующих пунктов.

10.2.2 Отвечающий SOAP узел, который получает поле заголовка HTTP **Accept** с одним или более дополнительным медиа типом с привилегиями, равными «**application/fastsoap**» медиа типу, должен интерпретировать «**application/fastsoap**» медиа тип как имеющий самые высокие привилегии (см. IETF RFC 2616, 14.1) и ответить с помощью этого медиа типа.

10.2.3 Отвечающий SOAP узел должен добавить поле заголовка HTTP **Fast-Enabled** с пустым значением поля, для того чтобы указать на поддержку быстрого веб-сервиса, если отвечающий узел не может установить по запросу HTTP, отправитель может обрабатывать содержимое идентифицированного «**application/fastsoap**» медиа типа.

11 SOAP сообщения быстрого инфо-набора и SOAP HTTP привязка

В данном разделе описана SOAP HTTP привязка (см. W3C SOAP Часть 2, раздел 7), которая поддерживает передачу SOAP сообщений быстрого инфо-набора через HTTP (называемая SOAP HTTP привязкой быстрого инфо-набора).

11.1 Имплементация SOAP HTTP привязки быстрого инфо-набора должна быть способна отправлять и получать SOAP сообщения быстрого инфо-набора с помощью «**application/soap+fastinfoset**» медиа типа, параметры и правильное использование которого описаны в B.2.

Примечание — SOAP HTTP привязка поддерживает передачу запросов и ответов с использованием других медиа типов, если такие медиа типы описывают передачу инфо-наборов W3C SOAP сообщений.

11.2 SOAP HTTP привязка может при отправке запросов предоставлять поле заголовка HTTP **Accept** (см. IETF RFC 2616, 14.1). Этот заголовок:

a) должен указывать на возможность принять, по крайней мере, «**application/soap+fastinfoset**» медиа тип,

b) может дополнительно указывать на готовность принять другие медиа типы для передачи W3C SOAP сообщений.

Примечание — Реализация может отправлять XML SOAP сообщение-запрос с полем заголовка HTTP **Accept** «**Accept: application/soap+fastinfoset, application/soap+xml**», чтобы указать на способность принимать SOAP сообщения быстрого инфо-набора и в дополнение указать на готовность принимать XML SOAP сообщения.

12 SOAP-ориентированные описания сервисов, поддерживающие интерфейсную привязку ASN.1 SOAP

12.1 Общие положения

12.1.1 В данном разделе даны SOAP-ориентированные описания сервисов, которые поддерживают интерфейсные привязки ASN.1 SOAP (см. 12.4.7).

12.1.2 SOAP-ориентированное описание сервиса представляет собой набор документов, определяющих интерфейсы и семантику веб-сервиса, который должен быть предоставлен посредством обмена SOAP сообщениями.

12.1.3 SOAP-ориентированное описание сервиса должно удовлетворять требованиям, указанным в данном разделе, но с другой стороны, не существует никаких ограничений на форму этих документов или на язык (естественный или формальный), на котором они написаны.

Примечание — В приложении E описано использование WSDL 1.1 [2] в качестве языка для написания SOAP-ориентированных описаний сервисов.

12.1.4 SOAP-ориентированное описание сервиса должно устанавливать:

a) набор схем (см. 12.2);

b) набор абстрактных интерфейсов, каждый из которых содержит набор абстрактных операций (см. 12.3);

c) набор интерфейсных привязок (набор ASN.1 SOAP интерфейсных привязок), каждая из которых содержит набор операционных привязок (см. 12.4).

12.2 Схемы

12.2.1 SOAP-ориентированное описание сервиса для определенного веб-сервиса может включать определение одного или нескольких типов данных содержимого, которые должны быть введены в SOAP сообщения для обеспечения этого веб-сервиса. Это могут быть данные содержимого, которые должны быть введены в тело сообщения, в блоки заголовков и в ошибки.

12.2.2 Данные содержимого (если таковые имеются) должны быть определены одной или несколькими XSD схемами. Каждая схема должна быть или импортирована, предоставляя свое пространство имен URI, или встроена в описание сервиса.

Примечание — «XSD схема» в данном случае является не «документом-схемой», а абстрактной схемой (набор компонентов схемы — см. W3C XML схема), чьи XML представления состоят из одного или нескольких «xsd:schema» членов информации элемента. Как правило, схема встроена в описание сервиса с помощью добавления ее XML представления.

12.2.3 Набор всех XSD схем, импортированных или встроенных в описание сервиса, называется исходным набором схем.

12.2.4 Каждый тип данных содержимого в исходном наборе схем может быть описан либо высокоуровневым компонентом схемы **element declaration**, либо высокоуровневым компонентом схемы **complex type definition** или **simple type definition**.

12.3 Абстрактные интерфейсы и абстрактные операции

12.3.1 Абстрактный интерфейс описывается путем предоставления информации для набора абстрактных операций и неявно содержит следующую информацию о схеме (вытекающую из другой информации в описании сервиса):

a) RPC схема (см. 12.3.2), а также

b) ASN.1 набор схем (см. 12.3.3).

12.3.2 RPC схема представляет собой специально созданную XSD схему, поддерживающую конкретные интерфейсы RPC-вида, и должна быть получена, как указано в 12.5. Исходный набор схем с добавленной RPC схемой называется полным набором схем абстрактного интерфейса.

12.3.3 ASN.1 набор схем является ASN.1 отображением полного набора схем абстрактного интерфейса. Каждая XSD схема в полном наборе схем должна быть отображена независимо в ASN.1, как указано в МСЭ-Т X.694 | ISO/IEC 8825-5. ASN.1 модули, сформированные X.694 отображением, формируют ASN.1 набор схем абстрактного интерфейса.

12.3.4 Абстрактная операция определяется с помощью следующей информации:

a) названия операции (квалифицированное название);

b) (опционально) определения входного сообщения;

c) (опционально) определения выходного сообщения и

d) набора из нуля или более определений сообщений об ошибке.

12.3.5 Если присутствуют оба определения входного и выходного сообщения, то порядок, в котором они появлялись (например, запрос и ответ), не смотря на то что он не рассматривается в настоящем стандарте, должен быть указан описанием сервиса.

12.3.6 Определение входного сообщения и определение выходного сообщения должны иметь одну из следующих форм:

a) ноль или одно высокоуровневое **element declaration**, которое принадлежит к одной из схем в исходном наборе схем (см. 12.2.4), или

b) список из нуля или более различных неквалифицированных имен, каждое из которых связано с высокоуровневым **complex type definition** или **simple type definition**, которое принадлежит одной из схем в исходном наборе схем (см. 12.2.4).

Примечание — В некоторых языках описания сервисов (например WSDL 1.1 [2]) форма описания входного или выходного сообщения будет указана операционной привязкой, как ограничения на информацию (не описано в настоящем стандарте), которая предоставляется абстрактной операцией.

12.3.7 Каждое определение сообщения об ошибке должно указывать высокоуровневое **element declaration**, которое принадлежит одной из схем в исходном наборе схем (см. 12.2.4).

12.3.8 Интерфейс называется документо-ориентированным, если для каждой операции определение входного сообщения (если имеется) и определение выходного сообщения (если имеется) имеют вид согласно 12.3.6, a).

12.3.9 Интерфейс называется RPC-ориентированным, если для каждой операции определение входного сообщения (имеется) и определение выходного сообщения (если имеется) имеют вид согласно 12.3.6, b).

12.3.10 В противном случае интерфейс не является ни документо-ориентированным, ни RPC-ориентированным и описание сервиса, содержащее такие абстрактные интерфейсы, не является SOAP-ориентированным описанием сервиса, как описано в настоящем стандарте.

12.4 Интерфейсные и операционные привязки

12.4.1 Интерфейсная привязка связывает абстрактный интерфейс с дополнительной информацией, результатом чего является полная спецификация конкретного интерфейса.

12.4.2 Интерфейсная привязка описывается с помощью предоставления следующей информации:

a) (опционально) идентификатора объекта, присвоенного конкретной операции;

b) набора операционных привязок;

c) URI транспорта;

d) вида конкретного интерфейса (документ или RPC) и

e) сведения о том, поддерживает ли конкретный интерфейс быстрые веб-сервисы.

12.4.3 Идентификатор объекта, присваиваемый конкретной операции (если таковые имеются), должен быть присвоен в соответствии с МСЭ-Т X.660 | ISO/IEC 9834-1 и должен однозначно идентифицировать конкретную операцию.

Примечание — Две конкретные операции, основанные на одной и той же абстрактной операции, будут иметь разные идентификаторы объектов.

12.4.4 Операционная привязка связывает абстрактные операции с дополнительной информацией, результатом чего является полная спецификация конкретной операции, которая должна быть выполнена посредством обмена W3C SOAP сообщениями.

12.4.5 Транспортом является протокол, используемый для передачи SOAP сообщения от одного SOAP узла к другому SOAP узлу, он должен быть описан как URI.

Примечание — В контексте SOAP транспортом называют привязки. Типичными примерами являются XML SOAP HTTP привязка (см. W3C SOAP Часть 2, раздел 7) и ASN.1 SOAP HTTP привязка (см. раздел 10).

12.4.6 Если абстрактный интерфейс является документо-ориентированным, то видом конкретного интерфейса должен быть документ. Если абстрактный интерфейс является RPC-ориентированным, то видом конкретного интерфейса должен быть RPC.

12.4.7 К интерфейсной привязке, которая поддерживает быстрые веб-сервисы, относятся как к интерфейсной привязке ASN.1 SOAP, и конкретные операции могут быть выполнены посредством обмена ASN.1 SOAP сообщениями.

12.4.8 Операционная привязка описывается с помощью предоставления следующей информации:

a) (опционально) SOAP действия URI;

b) нуля или более определений блоков заголовков SOAP, каждое из которых состоит из высокоуровневого **element declaration**;

c) нуля или более идентификаторов объектов, присвоенных высокоуровневому **element declaration**; и

d) указания для каждого высокоуровневого **element declaration**, должно ли оно быть представлено как поддерево или как встроенное в ASN.1 значение.

12.4.9 SOAP действие URI является URI, который должен быть помещен (при наличии) в параметр **action** MIME типа «**application/fastsoap**» (см. B.1) для HTTP запроса, когда ASN.1 SOAP HTTP привязка (см. раздел 10) указана в качестве транспорта.

12.4.10 Каждое высокоуровневое **element declaration** в определении блока заголовка должно принадлежать одной из схем в исходном наборе схем (см. 12.2.4).

12.4.11 Если идентификатор объекта присваивается конкретной операции, то уникальный идентификатор объекта может быть присвоен одному или нескольким высокоуровневым **element declaration**, указанным в определении входного сообщения, определении выходного сообщения, определении сообщений об ошибке, определении блока заголовка, или конкретной операции, или неявно сформирован для конкретной операции RPC-вида. Если конкретная операция не имеет идентификатора объекта, то присвоение идентификаторов объектов таким **element declaration** не допускается.

12.4.12 Каждый идентификатор объекта, присваиваемый **element declaration**, должен быть присвоен в соответствии с МСЭ-Т X.660 | ISO/IEC 9834-1 и должен однозначно идентифицировать **element declaration**. Каждый такой идентификатор объекта должен быть таким же, как идентификатор объекта конкретной операции с одним или более дополнительными компонентами идентификатора объекта, добавленными справа.

Примечание — Это позволяет использовать условные идентификаторы объектов, чтобы определить тип встроенного в ASN.1 закодированного значения в SOAP сообщениях (см. 9.2.2). Каждый такой условный идентификатор объекта будет состоять только из дополнительных компонентов идентификатора объекта, в то время как ранние компоненты идентификатора объекта не будет переданы.

12.4.13 Любой **element declaration** с идентификатором объекта, присвоенным ему, должен служить дополнительным указанием на 12.4.8, d) — **element declaration** должен быть закодирован как встроенное в ASN.1 закодированное значение, а не как поддерево.

Примечание — Использование встроенных в ASN.1 закодированных значений не требует идентификатора объекта. При отсутствии идентификатора объекта для высокоуровневого **element declaration**, квалифицированное имя **element declaration** используется для определения типа встроенного в ASN.1 закодированного значения в SOAP сообщениях (см. 9.2.3).

12.5 RPC схема

12.5.1 RPC схема представляет собой специально созданную XSD схему, поддерживающую конкретные интерфейсы RPC-вида (см. 12.4.6). RPC схема является неявно сформированной и не импортированной или встроенной в описание сервиса.

Примечание — RPC схема неявно присутствует во всех SOAP-ориентированных описаниях сервисов, но она пуста, если в описании сервиса нет интерфейсов RPC-вида.

12.5.2 RPC схема для определенного абстрактного интерфейса (то есть связи с конкретным интерфейсом RPC-вида) должна быть построена следующим образом.

12.5.3 Для каждой абстрактной операции, указанной в RPC-ориентированном абстрактном интерфейсе, компонент схемы **element declaration** со следующими свойствами:

- **name**: локальное название операции;
- **target namespace**: название пространства имен операции;
- **type definition**: компонент схемы **complex type definition**, как указано в 12.5.4;
- **scope**: **global**

и с остальными свойствами либо **absent**, либо установленными как **false**, либо пустыми (в соответствующих случаях) должен быть добавлен к RPC схеме.

12.5.4 Компонент схемы **complex type definition** в свойстве **type definition** должен иметь следующие свойства:

- **name**: нет;
- **target namespace**: **absent**;
- **base type definition**: **ur-type**;
- **derivation method**: **restriction**;
- **content type**: **element only** и компонент схемы **particle**, как указано в 12.5.5, и

с остальными свойствами либо **absent**, либо установленными как **false**, либо пустыми (в соответствующих случаях) должен быть добавлен к RPC схеме.

12.5.5 Компонент схемы **particle** в свойстве **content type** должен иметь следующие свойства:

- **min occurs**: 1;
- **max occurs**: 1;
- **term**: компонент схемы **model group**, как указано в 12.5.6,

и должен быть добавлен к RPC схеме.

12.5.6 Компонент схемы **model group** в свойстве **term** должен иметь следующие свойства:

- **compositor**: **sequence**;
- **particles**: список из нуля или более компонентов схемы **particle**, как указано в 12.5.7 (см. 12.3.6, b)),

и должен быть добавлен к RPC схеме.

12.5.7 Каждая **particle** в списке частиц свойства **particle** должна обладать следующими свойствами:

- **min occurs**: 1;
- **max occurs**: 1;
- **term**: компонент схемы **element declaration**, как указано в 12.5.8,

и должна быть добавлена к RPC схеме.

12.5.8 Компонент схемы **element declaration** в свойстве **term** должен иметь следующие свойства:

- **name**: одно из неквалифицированных имен, указанных в определении входного сообщения абстрактной операции;
- **target namespace**: отсутствует;
- **type definition**: компонент схемы **complex type definition** или **simple type definition**, связанный с неквалифицированным именем в определении входного сообщения абстрактной операции (см. 12.3.6, b));

- **scope**: компонент схемы **complex type definition**, указанный в 12.5.4,

и должен быть добавлен к RPC схеме.

12.5.9 Для каждой абстрактной операции, указанной в абстрактном интерфейсе RPC-вида, с определением выходного сообщения компонент схемы **element declaration** со следующими свойствами:

- **name**: локальное название операции с суффиксом «Response»;
- **target namespace**: название пространства имен операции;
- **type definition**: компонент схемы **complex type definition**, как указано в 12.5.10;
- **scope**: **global**

и с остальными свойствами либо **absent**, либо установленными как **false**, либо пустыми (в соответствующих случаях) должен быть добавлен к RPC схеме.

12.5.10 Компонент схемы **complex type definition** в свойстве **type definition** должен иметь следующие свойства:

- **name**: нет;
- **target namespace**: **absent**;
- **base type definition**: **ur-type**;
- **derivation method**: **restriction**;
- **content type**: **element only** и компонент схемы **particle**, как указано в 12.5.11,

с остальными свойствами либо **absent**, либо установленными как **false**, либо пустыми (в соответствующих случаях) должен быть добавлен к RPC схеме.

12.5.11 Компонент схемы **particle** в свойстве **content type** должен иметь следующие свойства:

- **min occurs**: 1;
- **max occurs**: 1;
- **term**: компонент схемы **model group**, как указано в 12.5.15,

и должен быть добавлен к RPC схеме.

12.5.12 Компонент схемы **model group** в свойстве **term** должен иметь следующие свойства:

- **compositor**: **sequence**;
- **particles**. список из нуля или более компонентов схемы **particle**, как указано в 12.5.13 (см. 12.3.6, b)),

и должен быть добавлен к RPC схеме.

12.5.13 Каждая **particle** в списке частиц свойства **particle** должна обладать следующими свойствами:

- **min occurs**: 1;
- **max occurs**: 1;
- **term**: компонент схемы **element declaration**, как указано в 12.5.14,

и должна быть добавлена к RPC схеме.

12.5.14 Компонент схемы **element declaration** в свойстве **term** должен иметь следующие свойства:

- **name**: одно из неквалифицированных имен, указанных в определении выходного сообщения абстрактной операции;
- **target namespace**: **absent**;
- **type definition**: компонент схемы **complex type definition** или **simple type definition**, связанный с неквалифицированным именем в определении выходного сообщения абстрактной операции (см. 12.3.6, b));
- **scope**: компонент схемы **complex type definition**, указанный в 12.5.4,

и должен быть добавлен к RPC схеме.

12.5.15 Компонент схемы **complex type definition** или **simple type definition** свойства **type definition** в 12.5.8 и 12.5.14 должен быть копией компонента схемы в одной из XSD схем исходного набора схем. Этот компонент схемы должен быть добавлен к RPC схеме (если не был добавлен ранее) вместе с копией любого компонента схемы, который появится в одном из его свойств (если не был добавлен ранее) или в свойстве внутри свойства на любой глубине.

13 Использование SOAP-ориентированных описаний сервисов с интерфейсными привязками ASN.1 SOAP

13.1 SOAP-ориентированное описание сервиса, содержащее интерфейсные привязки ASN.1 SOAP для определенного быстрого веб-сервиса, влияет на форму и содержимое всех ASN.1 SOAP

сообщений, отображенных в и отображенных из W3C SOAP сообщений (описанных интерфейсными привязками ASN.1 SOAP), для обеспечения быстрого веб-сервиса.

13.2 Каждое W3C SOAP сообщение должно быть входным или выходным сообщением конкретной операции конкретного интерфейса (см. 12.4), указанного в описании сервиса. Входные сообщения должны идти от SOAP узла клиента к SOAP узлу сервиса, выходные сообщения — в противоположном направлении. W3C SOAP сообщения, которые являются выходными сообщениями конкретных операций конкретных интерфейсов RPC-вида, разрешены только для конкретных операций, которые имеют определение входного сообщения.

13.3 Любой блок заголовка (дочерний **Header** EII) или любая ошибка (дочерняя **Detail** EII) в W3C SOAP сообщении, которое является входным или выходным сообщением конкретной операции, должны быть встроенным информационным элементом в соответствии с одним из высокоуровневых **element declaration** для блоков заголовков и ошибок (соответственно) такой операции (см. 12.3.7 и 12.4.10).

13.4 Тело (дочернее **Body** EII) W3C SOAP сообщения, которое является входным или выходным сообщением определенной конкретной операции, должно быть встроенным членом информации элемента с соблюдением следующего **element declaration**:

a) если конкретная операция является членом конкретного интерфейса RPC-вида (см. 12.4.6), то высокоуровневое **element declaration** неявно сформировано (в RPC схеме, см. 12.5) для входного или выходного сообщения этой операции соответственно (см. 12.5.3 и 12.5.9), или

b) если конкретная операция является членом конкретного интерфейса документ-вида (см. 12.4.6), то высокоуровневое **element declaration** указано в определении входного или выходного сообщения этой операции соответственно (см. 12.3.6, a)).

13.5 Встроенный член информации элемента, который описывается интерфейсной привязкой ASN.1 SOAP, должен быть представлен в W3C SOAP сообщении (которое отображается в ASN.1 SOAP сообщении) следующим образом:

a) как член информации элемента, который является поддеревом.

Примечание 1 — Такие члены будут отображаться в компоненты ASN.1 SOAP сообщения, которое является встроенным документом быстрого инфо-набора;

b) или в качестве встроенного в ASN.1 закодированного значения, которое формируются из члена информации элемента.

Примечание 2 — Описание сервиса указывает, является ли встроенный член информации элемента представленным в виде поддерева или в виде встроенного в ASN.1 закодированного значения (см. 12.4.8, d)).

13.6 Формирование встроенных в ASN.1 закодированных значений должно потребовать следующую информацию:

a) ASN.1 тип;

b) идентификацию ASN.1 типа и

c) ASN.1 значение идентифицированного ASN.1 типа.

13.7 ASN.1 тип должен быть членом ASN.1 набора схемы абстрактных интерфейсов (см. 12.3.3), отображенных из высокоуровневого **element declaration**, а встроенный член информации элемента компилируется с высокоуровневым **element declaration**.

13.8 Идентификация ASN.1 типа должна являться значением типа `Identifier`.

13.8.1 Если идентификатор объекта присвоен высокоуровневому **element declaration** (см. 12.4.12), то должна применяться альтернатива `roid` значения `Identifier`, а значение `roid` должно быть установлено из соответственного идентификатора объекта, который является дополнительным компонентом идентификатора объекта к присвоенному идентификатору объекта (см. 12.4.12).

13.8.2 В противном случае должна применяться альтернатива `qName` значения `Identifier`, а значение `qName` должно быть установлено из квалифицированного имени высокоуровневого **element declaration**.

13.9 Учитывая ASN.1 тип, значение `Identifier` и значение ASN.1, встроенное в ASN.1 закодированное значение должно быть сформировано и вставлено в W3C SOAP сообщение, как указано в 9.4.

Примечание — В подразделе 9.4 определяется вставка ASN.1 значения (с идентификатором) в W3C SOAP сообщение. ASN.1 значение будет представлено в виде последовательности CII, которая является Base64 кодированием Basic Aligned PER кодирования значения ASN.1. Такое представление будет отображено в компонент ASN.1 SOAP сообщения, которое является строкой октета, значение которого является Basic Aligned PER кодированием соответствующего ASN.1 значения.

Приложение А
(обязательное)

ASN.1 модуль для ASN.1 SOAP

Ниже приведен ASN.1 модуль для ASN.1 SOAP. Схема повторяет некоторые типы, определенные в XSD модуле, как указано в МСЭ-Т X.694 | ISO/IEC 8825-5, и тип `Document`, определенный в `FastInfoset` модуле, как указано в МСЭ-Т X.891 | ISO/IEC 24824-1.

```
ASN1SOAP {joint-iso-itu-t(2) asn1(1) generic-applications(10) fast-web-services(1)
          modules(0) asn1soap(0)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
IMPORTS
        AnyURI,      Int, Language, QName
            FROM  XSD {joint-iso-itu-t asn1(1) specification(0) modules(0)
                      xsd-module(2)}
         Document, finf-doc-no-decl
             FROM FastInfoset {joint-iso-itu-t(2) asn1(1) generic-applications(10)
                               fast-infoset(0) modules(0) fast-infoset(0)};
Envelope ::= SEQUENCE {
        header Header,
        body-or-fault CHOICE {
             body Body,
             fault Fault}}
Header ::= SEQUENCE OF HeaderBlock
HeaderBlock ::= SEQUENCE {
        mustUnderstand BOOLEAN OPTIONAL,
        relay BOOLEAN OPTIONAL,
        role XSD.AnyURI DEFAULT ultimateReceiver,
        content Content}
ultimateReceiver XSD.AnyURI ::=
        «gost.e-ecolog.ru
Body ::= SEQUENCE {
      content Content OPTIONAL}
Fault ::= SEQUENCE {
        code     Code,
        reason SEQUENCE SIZE(1..MAX) OF Text,
        node     XSD.AnyURI OPTIONAL,
        role       XSD.AnyURI OPTIONAL,
        detail    Content OPTIONAL}
Code ::= SEQUENCE {
        value Value,
        subcodes SEQUENCE OF XSD.QName}
Value ::= ENUMERATED { versionMismatch, mustUnderstand,
                       dataEncodingUnknown, sender, receiver}
Text ::= SEQUENCE {
        lang XSD.Language,
        text UTF8String}
Content ::= CHOICE {
        encoded-value SEQUENCE {
             schema-identifier OCTET STRING (SIZE (16)) OPTIONAL,
             id Identifier,
             encoding OCTET STRING },
        fast-infoset-document OCTET STRING
             (CONTAINING Document ENCODED BY finf-doc-no-decl)}
Identifier ::= CHOICE {
        roid RELATIVE-OID,
        qName XSD.QName}
NotUnderstood ::= XSD.QName
 notUnderstoodIdentifier Identifier ::= qName : {
        uri «gost.e-ecolog.ru                                    ,
        name «NotUnderstood»}
END
```

Приложение B
(обязательное)

MIME медиа типы для быстрых веб-сервисов

Данное приложение определяет два MIME медиа типа для быстрых веб-сервисов:

a) медиа тип «**application/fastsoap**» описывает ASN.1 SOAP сообщения, в частности значения ASN.1 типа **Envelope**, закодированные с помощью Basic Aligned PER (см. B.1);

b) медиа тип «**application/soap+fastinfoset**» описывает W3C SOAP сообщения быстрого инфо-набора, сериализованные как документы быстрого инфо-набора (см. B.2).

Данные MIME медиа типы определены ниже с помощью IETF MIME шаблона регистрации и были зарегистрированы в соответствии с процедурами IETF.

B.1 Медиа тип «application/fastsoap»

Название MIME медиа типа:
application
Название MIME подтипа:
fastsoap
Обязательные параметры:
Нет.
Необязательные параметры:
«action»: Этот параметр должен быть использован для определения намерений ASN.1 SOAP сообщения, как указано для параметра «action» W3C SOAP 1.2 «application/soap+xml» MIME медиа типа (см. W3C SOAP 1.2 Часть 2 , Приложение A). Значение параметра «action» должно быть абсолютной URI-ссылкой, как указано в IETF RFC 2396. Никаких ограничений не должно возлагаться на спецификацию URI или то, что она разрешима.
Аспекты кодирования:
Этот медиа тип используется для идентификации содержимого, которое является значением ASN.1 типа Envelope, указанным в ASN.1 SOAP модуле MCD-T X.892 | ISO/IEC 24824-2, закодированным с помощью Basic Aligned PER, указанным в MCD-T X.691 ISO/IEC 8825-2. Использование данного MIME медиа типа потребует дополнительной спецификации, если он используется на транспорте, который не обеспечивает 8-битной двоичной прозрачности. (Для быстрых веб-сервисов из MCD-T X.892 | ISO/IEC 24824-2 этот медиа тип всегда используется с Basic Aligned PER, с HTTP в качестве транспортного механизма, и никаких больших спецификаций не требуется.)
Аспекты безопасности:
Из-за того что ASN.1 SOAP сообщения могут переносить данные, описанные приложением, семантика которых является независимой от любой MIME оболочки (или контекста, в котором MIME оболочка используется), не следует ожидать, что возможно понять семантику ASN.1 SOAP сообщения на основе семантики MIME оболочки самостоятельно. Поэтому при использовании «application/fastsoap» медиа типа настоятельно рекомендуется полностью осознавать возможные осложнения, связанные с безопасностью в контексте, в котором используется ASN.1 SOAP сообщение. Последствия для безопасности обычно связываются со специфичными ASN.1 SOAP привязками к основному протоколу, а также с описанной приложением семантикой данных, переносимых в ASN.1 SOAP сообщении.
Аспекты совместимости:
Известных проблем совместимости не существует.
Опубликованная спецификация:
Рек. MCD-T X.892 | ISO/IEC 24824-2
Приложения, использующие данный медиа тип:
Нет известных приложений, которые используют данный медиа тип.
Дополнительная информация:
Расширение(я) файла(ов):
Не требуется и не ожидается, что ASN.1 SOAP сообщения будут храниться в виде файлов.
Контактное лицо и адрес электронной почты для получения дополнительной информации:
ITU-T ASN.1 докладчик (tsbmail@itu.int)
ISO/IEC JTC1/SC6 ASN.1 докладчик (itt@iso.org)

Предполагаемое использование:
ОБЩЕЕ
Автор/Ответственный за изменения:
Совместные процедуры голосования ITU-T и ISO/IEC в соответствии с Рек. МСЭ-Т А.23 *Сотрудничество с Международной организацией по стандартизации (ИСО) и Международной электротехнической комиссией (МЭК) по вопросам информационных технологий*, приложение А и Директивы ISO/IEC JTC1, приложение К.

B.2 Медиа тип «application/soap+fastinfoset»

Название MIME медиа типа:
application
Название MIME подтипа:
soap+fastinfoset
Обязательные параметры:
Нет.
Необязательные параметры:
«action»: Этот параметр должен быть использован для определения намерений ASN.1 SOAP сообщения, как указано для параметра «action» W3C SOAP 1.2 «application/soap+xml» MIME медиа типа (см. W3C SOAP 1.2 Часть 2 , Приложение А). Значение параметра «action» должно быть абсолютной URI-ссылкой, как указано в IETF RFC 2396. Никаких ограничений не должно возлагаться специфику URI или то, что она разрешима.
Аспекты кодирования:
Этот медиа тип используется для идентификации W3C SOAP инфо-наборов сообщений, сериализованных как документы быстрого инфо-набора, указанные в Рекомендации МСЭ-Т X.892 | ISO/IEC 24824-2.
Использование данного MIME медиа типа потребует дополнительной спецификации, если он используется на транспорте, который не обеспечивает 8-битной двоичной прозрачности. (Для быстрых веб-сервисов из МСЭ-Т X.892 ISO/IEC 24824-2 этот медиа тип всегда используется с Basic Aligned PER, с HTTP в качестве транспортного механизма, и никаких больших спецификаций не требуется).
Аспекты безопасности:
Из-за того что W3C SOAP инфо-наборы сообщений могут переносить данные, описанные приложением, семантика которых является независимой от любой MIME оболочки (или контекста, в котором MIME оболочка используется), не следует ожидать, что возможно понять семантику W3C SOAP инфо-наборов сообщений на основе семантики MIME оболочки самостоятельно. Поэтому при использовании «application/soap+fastinfoset» медиа типа настоятельно рекомендуется полностью осознавать возможные осложнения, связанные с безопасностью в контексте, в котором используется W3C SOAP инфо-наборы сообщений. Последствия для безопасности обычно связываются со специфичными SOAP привязками к основному протоколу, а также с описанной приложением семантикой данных, переносимых в W3C SOAP инфо-наборах сообщений.
Аспекты совместимости:
Известных проблем совместимости не существует.
Опубликованная спецификация:
Рек. МСЭ-Т X.892 | ISO/IEC 24824-2
Приложения, использующие данный медиа тип:
Нет известных приложений, которые используют данный медиа тип.
Дополнительная информация:
Магическое(ие) число(а):
Для получения дополнительной информации об идентификации документа быстрого инфо-набора обратитесь к секции магическое число медиа типа «application/fastinfoset». Идентификация W3C SOAP инфо-наборов сообщений, сериализированного как документ быстрого инфо-набора, требует, чтобы такой документ быстрого инфо-набора был разобран и чтобы свойства элемента информации, соответствующего корню дерева элементов, соответствовали свойствам элемента информации SOAP Envelope, как указано в W3C SOAP 1.2, 5.1.
Расширение(я) файла(ов):

Не требуется и не ожидается, что W3C SOAP инфо-наборы сообщений, сериализованные как документы быстрого инфо-набора, будут храниться в виде файлов.

Контактное лицо и адрес электронной почты для получения дополнительной информации:

ITU-T ASN.1 докладчик (tsbmail@itu.int)

ISO/IEC JTC1/SC6 ASN.1 докладчик (ittf@iso.org)

Предполагаемое использование:

ОБЩЕЕ

Автор/Ответственный за изменения:

Совместные процедуры голосования ITU-T и ISO/IEC в соответствии с Рек. МСЭ-Т А.23 *Сотрудничество с Международной организацией по стандартизации (ИСО) и Международной электротехнической комиссией (МЭК) по вопросам информационных технологий*, приложение А и Директивы ISO/IEC JTC1, приложение К.

Приложение С
(справочное)

Учебное пособие по использованию быстрых веб-сервисов

Данное приложение содержит учебный материал по использованию быстрых веб-сервисов. В нем описаны некоторые преимущества использования быстрых веб-сервисов, освещены различия между концептуальной и оптимизированной обработкой SOAP сообщений. Это показано на примере простого обмена, когда клиент посылает сообщение-запрос и получает ответное сообщение. В данном приложении рассмотрено также использование описаний сервисов и приведен пример описания сервиса обмена сообщениями (в WSDL 1.1, см. [2]).

С.1 Преимущества быстрых веб-сервисов

Спецификация быстрых веб-сервисов основана на использовании ASN.1 определения SOAP сообщений и их содержимого, а также на использовании бинарных кодирований этих сообщений. Это обеспечивает главное преимущество быстрого веб-сервиса, быструю компьютерную обработку и малый пропускной канал под сообщения, а также ряд XML SOAP оптимизаций, которые будут рассмотрены ниже.

С.1.1 Инструменты ASN.1

Инструменты ASN.1 могут быть использованы в разработке ASN.1 SOAP процессоров, ввиду того что XML SOAP процессоры, по большей части, написаны вручную, используя лишь W3C XML схему для SOAP в качестве ориентира, так как инструменты XML привязки мало подходят для разработки оптимальных XML SOAP процессоров. ASN.1 подход предоставляет выбор: разрабатывать SOAP процессоры с помощью инструментов или же вручную, без серьезных потерь производительности и с потенциальной выгодой во времени для выхода на рынок.

С.1.2 Оптимизированные функции

ASN.1 SOAP предоставляет ряд функций оптимизации (помимо уплотнения и эффективной обработки за счет использования ASN.1 и PER — см. Рекомендацию МСЭ-Т X.691 | ISO/IEC 8825-2) для SOAP узлов:

a) тело ASN.1 SOAP сообщения явно отделено от кодирования ошибки ASN.1 SOAP сообщения. Это облегчает идентификацию и управление ошибками;

b) рекурсивные ошибочные подкоды (см. W3C SOAP Часть 1, 5.4.6) для W3C SOAP сообщений сплюснуты в последовательность ошибочных подкодов для ASN.1 SOAP сообщения. Это позволяет декодеру узнать, сколько существует ошибочных подкодов до декодирования;

c) вместо квалифицированных имен могут быть использованы условные идентификаторы объектов ASN.1. Сообщения для описания сервисов могут быть с аннотацией, состоящей из условных идентификаторов объектов. Такие идентификаторы после кодирования, как правило, более компактны, чем квалифицированные имена, что дает меньшие размеры сообщений;

d) для всех атрибут-связанных компонентов блока заголовка ASN.1 SOAP значения по умолчанию заданы;

e) для заданных в W3C SOAP кодов ошибок вместо квалифицированных имен используются перечисляемые значения.

С.1.3 Компактные сообщения и эффективная обработка

ASN.1 SOAP сообщения, закодированные с помощью ASN.1 PER, обычно предоставляют веб-сервисы, которые требуют меньшей вычислительной мощности (и, следовательно, обеспечивают более высокую скорость обработки транзакций) и меньшей пропускной способности сети, чем при использовании XML кодирования. Это может быть полезно в ряде областей:

a) в устройствах с ограничениями, таких как мобильные телефоны, смарт-карты или даже RFID устройства, которые имеют ограничения вычислительной мощности, памяти и аккумулятора.

Примечание 1 — Для технологий в области аккумуляторов не существует эквивалента закона Мура (их жизнь не удваивается каждые 18 месяцев);

b) в системах с ограниченной пропускной способностью, таких как беспроводные сети.

Примечание 2 — Радиочастоты для беспроводных сетей, таких как сеть GSM, могут не меняться в течение 10 лет. Для радиочастот не существует эквивалента закона Мура (пропускная способность не удваивается каждые 18 месяцев);

c) в высокопропускных транзакционных системах, таких как системы, необходимые для обработки определенного количества SOAP сообщений в секунду от многих клиентов.

С.1.4 Эффективная обработка SOAP посредников

SOAP посредники имеют потенциал для обработки намного большего числа SOAP сообщений, чем начальные SOAP отправители и конечные SOAP получатели. SOAP посредники, обрабатывая ASN.1 SOAP сообщения, могут легко определить блоки заголовков ASN.1 SOAP для обработки (включая декодирование), пропуская при этом (и копируя) другие блоки заголовков SOAP (предназначенные для других SOAP посредников или конечного SOAP получателя) и SOAP тело. (Вследствие того, что блоки заголовков SOAP и SOAP тело кодируются как последовательность с заданной длиной октетов).

Примечание — ASN.1 SOAP посредники также могут эффективно управлять **fault**, так как **fault** всегда будет появляться в конце сообщения (после блоков заголовков SOAP) и будет гарантированно начинаться на границе байта, если блоки заголовков присутствуют. Таким образом, у посредника нет необходимости декодировать **fault**, пока он выполняет процессы, не заданные моделью обработки SOAP.

С.2 Концептуальная и оптимизированная обработка ASN.1 SOAP сообщений

С.2.1 Общие положения

С.2.1.1 Концептуальное отображение ASN.1 SOAP сообщений в W3C SOAP сообщения и наоборот гарантирует, что модель обработки W3C SOAP может быть применена к ASN.1 SOAP сообщениям. В шести последующих подпунктах описаны концептуальные шаги, необходимые для обработки сообщений первоначальным SOAP отправителем, SOAP посредником и конечным SOAP получателем, и оптимизированные шаги, необходимые для SOAP посредника.

С.2.1.2 Первоначальный SOAP отправитель (см. W3C SOAP Часть 1, 1.5.3), имплементируя ASN.1 SOAP HTTP привязку, формирует ASN.1 SOAP сообщения в следующем порядке:

a) создает новое W3C SOAP сообщение и вставляет новые встроенные в ASN.1 абстрактные значения в W3C SOAP сообщение,

b) отображает W3C SOAP сообщение в ASN.1 SOAP сообщение,

c) кодирует ASN.1 SOAP сообщение, используя Basic Aligned PER, в последовательность октетов, которая составляет содержимое HTTP запроса.

С.2.1.3 Если первоначальный SOAP отправитель использует SOAP Request-Response Message Exchange Pattern (см. W3C SOAP Часть 2, 6.2), то SOAP отправитель (см. W3C SOAP Часть 1, 1.5.3) будет ждать ответа и меняться ролями, чтобы стать конечным SOAP получателем (см. W3C SOAP Часть 1, 1.5.3).

С.2.1.4 SOAP посредник (см. W3C SOAP Часть 1, 1.5.3), имплементируя ASN.1 SOAP HTTP привязку, обрабатывает ASN.1 SOAP сообщения в следующем порядке:

a) декодирует последовательность октетов, полученных из содержимого HTTP запроса или ответа, используя Basic Aligned PER, для получения входного ASN.1 SOAP сообщения;

b) отображает входное ASN.1 SOAP сообщение для получения входного W3C SOAP сообщения;

c) определяет и обрабатывает встроенные в ASN.1 абстрактные значения во входном W3C SOAP сообщении;

d) изменяет входное W3C SOAP сообщение, чтобы оно стало выходным W3C SOAP сообщением, и вставляет новые встроенные в ASN.1 абстрактные значения в выходное W3C SOAP сообщение;

e) отображает выходное W3C SOAP сообщение в выходное ASN.1 SOAP сообщение;

f) кодирует выходное ASN.1 SOAP сообщение, используя Basic Aligned PER, в последовательность октетов, которая составляет содержимое HTTP ответа или запроса.

С.2.1.5 Конечный SOAP получатель, имплементируя ASN.1 SOAP HTTP привязку, обрабатывает ASN.1 SOAP сообщения в следующем порядке:

a) декодирует последовательность октетов, полученную из содержимого HTTP запроса, используя Basic Aligned PER, чтобы получить ASN.1 SOAP сообщение;

b) отображает ASN.1 SOAP сообщение для получения W3C SOAP сообщения;

c) определяет и обрабатывает встроенные в ASN.1 абстрактные значения в W3C SOAP сообщении.

С.2.1.6 Если конечным SOAP получателем используется SOAP Request-Response Message Exchange Pattern, то SOAP узел будет меняться ролями, чтобы стать начальным SOAP отправителем, и будет отправлять ASN.1 SOAP сообщение в ответ.

С.2.1.7 Концептуальные шаги для отображения в W3C SOAP сообщения и из W3C SOAP сообщений и для обработки встроенных в ASN.1 значений (идентификация и обработка в W3C SOAP сообщении и вставка в W3C SOAP сообщение) указаны в разделах 6 — 9. SOAP узел, однако, может решить сделать оптимизацию процесса, пропуская концептуальные шаги до тех пор, пока результаты остаются такими же, как если бы концептуальные шаги были выполнены (см. 6.4). Например, шаги b) — e) в С.2.1.4 являются концептуальными шагами, и SOAP посредник может решить сделать оптимизацию путем обработки ASN.1 SOAP сообщений в следующем порядке:

a) декодировать последовательность октетов, полученную из содержимого HTTP запроса, используя Basic Aligned PER, для получения входного ASN.1 SOAP сообщения;

b) идентифицировать и обрабатывать встроенные в ASN.1 абстрактные значения во входном ASN.1 SOAP сообщении;

c) изменить входное ASN.1 SOAP сообщение, чтобы оно стало выходным ASN.1 SOAP сообщением (или создать новое выходное ASN.1 SOAP сообщение), и вставить новые встроенные в ASN.1 абстрактные значения в выходное ASN.1 SOAP сообщение;

d) закодировать выходное ASN.1 SOAP сообщение, используя Basic Aligned PER, в последовательность октетов, которая составляет содержимое HTTP ответа.

С.2.2 Пример

Пример приведен в следующих подпунктах с точки зрения приложения, отправляющего ASN.1 SOAP сообщение запроса и получающего ответ. Быстрый веб-сервис описан в С.3.2 с использованием WSDL 1.1 и основан на образце W3C SOAP сообщения в W3C SOAP Часть 1, 1.4. Сервис является одним из тех, где приложение может

запросить последнее предупреждение о некоторой информации, которая важна для приложения (или пользователя приложения). Запрашивающее приложение будет отправлять пустое ASN.1 SOAP сообщение (без содержимого, описанного приложением) и получать в ответ ASN.1 SOAP сообщение с двумя частями содержимого, описанного приложением (описано в C.3.2 с использованием WSDL 1.1) для предупреждения, которое соответствует:

a) блоку заголовка SOAP для свойств оповещения, а именно приоритету оповещения и времени истечения срока его действия;

b) содержимому SOAP тела для предупреждения самого себя о том, что является текстовым описанием предупреждения.

C.2.2.1 W3C SOAP сообщение-запрос

Приложение запрашивает последнее предупреждение, выполнив (с помощью соответствующего языка программирования, такого как Java) вызов метода без входных параметров, которые вернут предупреждение. Первоначальный SOAP отправитель создаст W3C SOAP сообщение без содержимого, представленного в XML как:

```
<env:Envelope xmlns:env=«gost.e-ecolog.ru                    >
              <env:Body>
              </env:Body>
</env:Envelope>
```

C.2.2.2 ASN.1 SOAP сообщение-запрос

Это W3C SOAP сообщение отображается в ASN.1 SOAP сообщение-запрос, состоящий из:

```
envelope Envelope ::= {
       header {}
       body-or-fault : body {} }
```

Тип **Envelope** описан в приложении A (см. также 6.1).

C.2.2.3 HTTP запрос

Затем ASN.1 SOAP сообщение кодируется, используя Basic Aligned PER, в последовательность октетов, что составляет содержимое HTTP запроса. Полем HTTP заголовка **Content-Type** является «**application/fastsoap**», а параметр **action** установлен как «**urn:alert**». Начальный SOAP узел заявляет, используя поле HTTP заголовка **Accept**, что и ASN.1 SOAP сообщения, и XML SOAP сообщения (в данном случае SOAP 1.1 сообщения [1]) поддерживаются.

```
POST /AlertPort HTTP/1.1
Content-Type: application/fastsoap; action=«urn:alert»
Accepts: application/fastsoap, application/text+xml
Content-Length: ....

... последовательность октетов ...
```

C.2.2.4 HTTP ответ

Затем первоначальный SOAP отправитель сменит роль и станет конечным SOAP получателем и будет ждать, пока не получит ответ на запрос. Полем HTTP заголовка **Content-Type** в ответ является «**application/fastsoap**».

```
HTTP/1.1 200 OK
Content-Type: application/fastsoap
Content-Length: ....

... последовательность октетов ...
```

C.2.2.5 ASN.1 SOAP сообщение-ответ

ASN.1 SOAP сообщение декодируется с помощью Basic Aligned PER для формирования значения ASN.1:

```
envelope Envelope ::= {
  header { {
     role «gost.e-ecolog.ru                          ,
     content : encoded-value {
        id : qName {
           uri «gost.e-ecolog.ru                          ,
          name «alertcontrol»},
        encoding {......}}}},
  body-or-fault : body {
     content : encoded-value {
        id : qnName {
           uri «gost.e-ecolog.ru                          ,
           name «alert»},
        encoding {......}}}}
```

C.2.2.6 W3C SOAP сообщение-ответ

C.2.2.6.1 ASN.1 SOAP сообщение отображается в W3C SOAP сообщение. W3C SOAP сообщение содержит блок заголовка W3C SOAP **alertcontrol** и элемент **alert** (информационный элемент) как дочерний **Body** EII:

```
<env:Envelope
        xmlns:env=«gost.e-ecolog.ru                                    >
    <env:Header>
        <n:altertcontrol
            xmlns:n=«gost.e-ecolog.ru
            env:role=«gost.e-ecolog.ru
env:encodingStyle=«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper»>
            ... Base64 content ...
            </n:altertcontrol>
    </env:Header>
    <env:Body>
        <m:alert
            xmlns:m=« gost.e-ecolog.ru
env:encodingStyle=«urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:
encoding-style:aper»>
            ... Base64 content ...
            </m:alert>
    </env:Body>
</env:Envelope>
```

C.2.2.6.2 Встроенное в ASN.1 абстрактное значение для блока заголовка W3C SOAP **alertcontrol** идентифицируется и обрабатывается, так как запрашивающий SOAP узел работает в роли «gost.e-ecolog.ru . Значение `Identifier` для блока заголовка W3C SOAP **alertcontrol** и встроенного в ASN.1 значения, декодированного с использованием Basic Aligned PER из содержимого Base64 с помощью ASN.1 типа `AlertControl`, связанного с `Identifier`, является следующим:

```
alertControlIdentifier Identifier ::= qName : {
        uri «gost.e-ecolog.ru                          ,
        name «altertcontrol» }
alertcontrol Alertcontrol ::= {
            role «gost.e-ecolog.ru                          ,
            priority 1,
            expires «2001-06-22T14:00:00-05:00» }
```

C.2.2.6.3 Встроенное в ASN.1 абстрактное значение для элемента **alert** (информационного элемента) определяется и обрабатывается так, как будто SOAP узел является конечным SOAP получателем. Значение `Identifier` для **alert** и встроенного в ASN.1 значения, декодированного с использованием Basic Aligned PER из Base64 содержимого с помощью ASN.1 типа `Alert`, связанного с `Identifier`, является следующим:

```
alertIdentifier Identifier ::= qName : {
      uri «gost.e-ecolog.ru                          ,
      name «alert» }
alert Alert ::= {
      msg «Pick up Mary at school at 2pm» }
```

C.3 Описания сервисов

C.3.1 Общие положения

C.3.1.1 Описания сервисов, выраженные в WSDL 1.1 [2], могут быть использованы без изменения для описания конечных точек ASN.1 SOAP. Это увеличивает область использования быстрых веб-сервисов, так как влияние на разработчиков веб-сервисов сведено к минимуму.

C.3.1.2 Интерфейс привязки WSDL 1.1 (см. приложение E) для SOAP 1.1 [1] может быть повторно использован для интерфейса привязки ASN.1 SOAP, тем самым обеспечивая, что WSDL документ является SOAP-ориентированным описанием сервиса (см. раздел 12 и приложение E), а WSDL 1.1 привязка соответствует разъяснениям и дополнениям, указанным в WS-I Basic Profile 1.0 [3] (см. приложение E).

C.3.2 Пример

C.3.2.1 Описание сервиса (выраженное в WSDL 1.1), показанное в C.3.3, описывает интерфейсные привязки ASN.1 SOAP для примера из C.2.2.

C.3.2.2 WSDL документ имеет два определения **xsd:schema**, содержащихся в **wsdl:type** (указывающем дочернее содержимое **Body** EII и блок заголовка W3C SOAP для единственного ответа). Эквивалентная ASN.1 схема получается путем применения МСЭ-Т X.694 | ISO/IEC 8825-5 к двум схемам (см. E.2 и 12.2).

C.3.2.3 ASN.1 SOAP HTTP привязка будет использована, так как значение атрибута **transport** на элементе **soapbind:binding** (интерфейсные привязки ASN.1 SOAP) равно «gost.e-ecolog.ru (см. E.4.2 и 12.4.2).

C.3.2.4 Поддержка быстрых веб-сервисов подробно описана для интерфейса привязки ASN.1 SOAP с помощью использования аннотации интерфейсной привязки ASN.1 SOAP (элемент **fast-service:binding**) в элементе **wsdl:binding** и после элемента **soapbind:binding** (см. E.4.5 и 12.4.2, e)).

C.3.2.5 Видом интерфейса привязки ASN.1 SOAP является документ (см. E.4.3 и 12.4.2, d)), поскольку интерфейсная привязка соответствует документ-литеральной привязке, как указано в WS-I Basic Profile 1.0.

C.3.2.6 Определение входного сообщения является пустым (без высокоуровневого **element declaration**, так как **soapbind:body** в **wsdl:input** в ссылках операционной привязки **AlertOperation**, по сути, без **wsdl:parts** (см. E.4.9.1 и 12.3.6, a))). Тем не менее SOAP действие URI существует, так как операционная привязка **AlertOperation** имеет атрибут **soapAction** (см. E.4.10 и 12.4.9). URI «**urn:alert**» будет размещен в параметре **action** MIME типа «**application/fastsoap**» (см. B.1) для поля HTTP заголовка **Content-Type** HTTP запроса (который содержит пустое ASN.1 SOAP сообщение).

C.3.2.7 Определение выходного сообщения имеет одно высокоуровневое **element declaration**, **alert:alert** (поскольку **soapbind:body** в **wsdl:output** в ссылках операционной привязки **AlertOperation**, по сути, один **wsdl:part** (см. E.4.9.1 и 12.3.6, a))).

C.3.2.8 Определение блока заголовка SOAP (блока заголовка W3C SOAP **alertcontrol**) описывается для выхода операционной привязки **AlertOperation** с высокоуровневым **element declaration alertcontrol:alertcontrol** (см. E.4.11 и 12.4.8, c)).

C.3.3 Описание сервиса, выраженное в WSDL 1.1

```
<definitions name=«Alert»
        xmlns=«gost.e-ecolog.ru
        xmlns:soap=«gost.e-ecolog.ru                                    »
        xmlns:soapbind=«gost.e-ecolog.ru          p.org/wsdl/soap/»
        xmlns:http=«gost.e-ecolog.ru
        xmlns:xsd=«gost.e-ecolog.ru          I/XMLSchema»
 xmlns:fast-service=«urn:ohn:joint-iso-itu-t:asn1:generic-applications :fast-web-services:description»
        xmlns:tns=«gost.e-ecolog.ru
        targetNamespace=«gost.e-ecolog.ru
        xmlns:alert=«gost.e-ecolog.ru
        xmlns:alertcontrol=«gost.e-ecolog.ru                          >

   <types>
<schema
            targetNamespace=«gost.e-ecolog.ru
            xmlns=«gost.e-ecolog.ru
            xmlns:soap=«gost.e-ecolog.ru
   elementFormDefault=«qualified»>
    <import namespace=« gost.e-ecolog.ru                                  >
    <element name=«alertcontrol»>
       <complexType>
          <sequence>
             <element name=«priority» type=«xsd:integer»/>
             <element name=«expires» type=«xsd:dateTime»/>
          </sequence>
          <xsd:attribute ref=«soap:role»/>
       </complexType>
    </element>
</schema>
<schema
       targetNamespace=«gost.e-ecolog.ru
       xmlns=«http://gost.e-ecolog.ru
       elementFormDefault=«qualified»>
    <element name=«alert»>
       <complexType>
          <sequence>
             <element name=«msg» type=«xsd:string»/>
          </sequence>
       </complexType>
    </element>
```

```
</schema>
   </types>

   <message name=«AlertRequest»>
   </message>

   <message name=«AlertResponse»>
<part name=«header » element=«alertcontrol:alertcontrol»/>
<part name=«body» element=«alert:alert»/>
   </message>

   <portType name=«AlertPortType»>
<operation name=«AlertOperation»>
    <input message=«tns:AlertRequest»/>
    <output message=«tns:AlertResponse»/>
</operation>
   </portType>

   <binding name=«AlertBinding» type=«tns:AlertPortType»>
<soapbind:binding
    transport=«gost.e-ecolog.ru
    style=«document»/>
<fast-service:binding/>

<operation name=«AlertOperation» soapAction=«urn:alert»>
    <input message=«tns:AlertRequest»>
        <soapbind:body use=«literal»/>
    </input>
    <output message=«tns:AlertResponse»>
        <soapbind:body use=«literal» parts=«body»/>
        <soapbind:header
                use=«literal»
                message=«tns:AlertResponse»
                part=«header »/>
    </output>
</operation>
   </binding>

   <service name=«AlertService»>
<port name=«AlertPort» binding=«tns:AlertBinding»>
    <soapbind:address location=«gost.e-ecolog.ru                    >
</port>
   </service>
   </definitions>
```

Приложение D
(справочное)

Общее предоставление сервисов с использованием быстрых веб-сервисов и веб-сервисов XML

В данном приложении описаны стратегии, которые могут быть применены клиентами быстрых веб-сервисов для того, чтобы взаимодействовать с SOAP узлами, не отличающимися быстротой. Стратегии используют функции ASN.1 HTTP привязки, описанные в разделе 10.

Стратегия применена успешно, если клиент быстрого веб-сервиса идентифицирует SOAP узел как быстрый и взаимодействие происходит путем обмена SOAP ASN.1 сообщениями. В противном случае стратегия применена неудачно, и взаимодействие происходит путем обмена XML SOAP сообщениями.

Примечание 1 — В настоящем приложении предполагается, что описание сервиса либо не используется, либо не содержит информацию, описывающую возможности быстрых веб-сервисов (или такая информация, если она присутствует, игнорируется), как указано в разделе 12.

Описаны три стратегии: одна оптимистическая (см. D.1) и две пессимистические (см. D.2).

Примечания

1 Важность и полезность некоторых из этих стратегий зависит от того, использует ли большинство веб-сервисов один запрос/ответ для соединения и выполняется ли кэширование информации о конкретном сервере.

2 После того как клиент удостоверился, поддерживает ли SOAP узел быстрые веб-сервисы, у него больше нет необходимости применять запрашивающие намеки (см. D.2.1) или ответные возможности (см. D.2.2). Однако кэширование возможностей клиента и SOAP узла следует использовать с осторожностью, так как реализация может измениться. Возможности SOAP узла могут гарантироваться только описанием сервиса или же тем, что было установлено в течение срока HTTP соединения. HTTP/1.1 имеет возможность «keep-alive» соединений, при которых несколько пар запрос/ответ могут быть отправлены через то же самое соединение.

D.1 Оптимистическая стратегия

D.1.1 При использовании данной стратегии клиент быстрого веб-сервиса оптимистично предполагает, что соответствующие SOAP узлы быстрые и имеют возможность обработки ASN.1 SOAP запрос-сообщений и ответов ASN.1 SOAP ответ-сообщениями.

D.1.2 Прием ASN.1 SOAP сообщения SOAP узлом может привести к двум возможным вариантам:

a) SOAP узел отвечает клиенту ошибкой HTTP с кодом 400-серии (см. RFC 2616, 10.4). Клиент быстрого веб-сервиса должен ожидать HTTP код состояния «415 Unsupported Media Type», но требуется также обработка и других кодов состояния, а именно «400 Bad Request».

Примечания

1 «415 Unsupported Media Type» будет появляться потому, что SOAP узел не поддерживает медиа тип HTTP для ASN.1 SOAP сообщения и, следовательно, не является быстрым.

2 HTTP предоставляет открытый механизм для поддержки HTTP кодов состояния определенных расширений. Соответствующие HTTP приложения должны обрабатывать любой нераспознанный код состояния 4xx как эквивалент кода состояния «400 Bad Request»;

b) SOAP узел отвечает ASN.1 SOAP ответ-сообщением.

D.1.3 Если происходит случай, описанный в D.1.2, то оптимистическая стратегия не удалась и клиент быстрого веб-сервиса должен повторно отправить семантически эквивалентное XML SOAP сообщение для взаимодействия либо приступить к пессимистической стратегии, описанной в D.2.

D.1.4 Если происходит случай, описанный в D.1.2, b), то оптимистическая стратегия удалась при первом запросе.

D.2 Пессимистическая стратегия

При использовании данной стратегии клиент быстрого веб-сервиса пессимистически предполагает, что соответствующие SOAP узлы не быстрые и не имеют возможности обработки ASN.1 SOAP запрос-сообщений и ответа ASN.1 SOAP ответ-сообщениями. Две пессимистические стратегии изложены в D.2.1 и D.2.2.

D.2.1 Пессимистическая стратегия с запросом подсказок

D.2.1.1 Клиент быстрого веб-сервиса посылает XML SOAP сообщение с запросом подсказок, соответствующим полю HTTP заголовка **Accept**, как указано в 10.1.4, ASN.1 SOAP HTTP привязке. Поле заголовка **Accept** будет содержать HTTP медиа типы для ASN.1 SOAP сообщения «**application/fastsoap**» и XML SOAP сообщение.

Примечание — Данная стратегия использует управляемое сервером согласование содержания (см. IETF RFC 2616, 12.1), которое является функцией HTTP/1.1. ASN.1 SOAP HTTP привязка поддерживает HTTP/1.1 и HTTP/1.0. W3C SOAP Часть 2, 7.1.2, рекомендует использовать для реализаций HTTP/1.1.

D.2.1.2 Прием XML SOAP сообщения SOAP узлом может привести к двум возможным исходам:

a) SOAP узел отвечает XML SOAP сообщением или

b) SOAP узел отвечает ASN.1 SOAP сообщением.

D.2.1.3 Если происходит случай, описанный в D.2.1.2, a), то пессимистическая стратегия не удалась, так как SOAP узел не является быстрым.

Примечание — Согласно пункту 10.2.2 ASN.1 SOAP HTTP привязка гарантирует, что SOAP узел, если способен, отвечает ASN.1 SOAP сообщением. Поэтому, если происходит неудача, можно гарантировать, что SOAP узел не поддерживает быстрые веб-сервисы.

D.2.1.4 Если происходит случай, описанный в D.2.1.2, b), то пессимистическая стратегия удалась при первом запросе.

D.2.2 Пессимистическая стратегия с возможностью быстрого ответа

D.2.2.1 Клиент быстрого веб-сервиса посылает XML SOAP сообщение без запроса подсказок в HTTP запросе.

D.2.2.2 Прием XML SOAP сообщения SOAP узлом может привести к двум возможным вариантам:

a) SOAP узел отвечает XML SOAP сообщением без указания возможности или

b) SOAP узел отвечает XML SOAP сообщением с указанием возможности, как указано в 10.2.3.

Примечание — Поддержка быстрых возможностей указывается полем HTTP заголовка **Fast-Enabled**.

D.2.2.3 Если происходит случай, описанный в D.2.2.2, a), то стратегия не удалась, так как SOAP узел не является быстрым. Клиент быстрого веб-сервиса должен повторить отправку семантически эквивалентного XML SOAP сообщения для взаимодействия.

Примечание — Согласно пункту 10.2.3 ASN.1 SOAP HTTP привязка гарантирует, что SOAP узел отвечает с полем HTTP заголовка **Fast-Enabled**, если узел является быстрым. Поэтому, если происходит неудача, можно гарантировать, что SOAP узел не поддерживает быстрые веб-сервисы.

D.2.2.4 Если происходит случай, описанный в D.2.2.2, b), и клиент быстрого веб-сервиса может обработать поле HTTP заголовка **Fast-Enabled**, то стратегия будет успешной при втором запросе.

Приложение Е
(справочное)

SOAP-ориентированное описание сервиса в WSDL 1.1

В данном приложении описано использование WSDL 1.1 [2] в сочетании с WS-I Basic Profile 1.0 [3] в качестве языка для написания SOAP-ориентированных описаний сервисов.

Примечание — WSDL 1.1 и WS-I Basic Profile 1.0 терминологии повторно используются в случае необходимости. Таким образом, W3C XML Information Set условия не используются при обращении к элементам XML и атрибутам, указанным в WSDL 1.1 и WS-I Basic Profile 1.0.

В данном приложении использованы следующие префиксы пространств имен:

soapbind	«**http://schemas.xmlsoap.org/wsdl/soap/**»
wsdl	«gost.e-ecolog.ru »
xsd	«gost.e-ecolog.ru »

Примечание — Выбор префикса не является семантически значимым.

Е.1 SOAP-ориентированные описания сервисов, выраженные в WSDL 1.1

Е.1.1 WSDL 1.1 документы, соответствующие профилю, указанному в WS-I Basic Profile 1.0, соответствуют требованиям SOAP-ориентированных описаний сервисов, указанных в разделе 12, и используют SOAP-ориентированные описания сервисов, указанные в разделе 13.

Примечание — WS-I Basic Profile 1.0 уточняет и дополняет WSDL 1.1 для обеспечения совместимости.

Е.1.2 Интерфейсная привязка (см. Е.4) для описания конкретных операций, которые должны быть выполнены путем обмена SOAP 1.1 [1] сообщениями, интерпретируется без изменения, а интерфейсная привязка ASN.1 SOAP для описания конкретных операций, которые должны быть выполнены путем обмена ASN.1 SOAP сообщениями, отображается из W3C SOAP сообщений (см. Е.4.6).

Примечание — Это гарантирует, что существующие документы WSDL 1.1 можно описать быстрым веб-сервисом без изменений. Быстрый веб-сервис и веб-сервис XML могут поддерживаться при использовании той же привязки и сетевого расположения (указанного URI) для приема входных сообщений и отправки выходных сообщений, которые являются SOAP 1.1 сообщениями и ASN.1 SOAP сообщениями (отображаются из W3C SOAP сообщений).

Е.2 Схема

Исходный набор схем (см. 12.2) есть множество объявленных схем XSD с использованием элементов **xsd:schema** в элементе **wsdl:types**.

Е.3 Абстрактный интерфейс и абстрактные операции

Е.3.1 Абстрактный интерфейс (см. 12.3.1) является элементом **wsdl:portType** в элементе **wsdl:definition**. Набор абстрактных интерфейсов является множеством всех элементов **wsdl:portType** в элементе **wsdl:definition**.

Е.3.2 Абстрактные операции (см. 12.3.4) абстрактного интерфейса являются элементом **wsdl:operation** в абстрактном интерфейсе. Набором абстрактных операций является множество всех элементов **wsdl:operation** в абстрактном интерфейсе.

Е.3.3 Названием операции (см. 12.3.4, a)) является значение атрибута **name** на абстрактные операции.

Е.3.4 Определением входного сообщения (см. 12.3.4, b)) является элемент **wsdl:input** в абстрактной операции. Элемент **wsdl:input** всегда присутствует в операциях на основе документов (см. 12.3.8) и в операциях на основе RPC (см. 12.3.9).

Примечание — WS-I Basic Profile 1.0 ограничивает наличие и порядок определения входного и выходного сообщений. Поддерживаются только такие операции, как односторонние операции (определение входного сообщения присутствует, а определение выходного сообщения отсутствует) и операции запрос-ответ (определение входного сообщения присутствует и описано первым, а определение выходного сообщения присутствует и описано вторым). (Операции запрос-ответ и уведомление не поддерживаются данным профилем).

Е.3.5 Определение выходного сообщения (см. 12.3.4, c)) является элементом **wsdl:output** (если представлено) в абстрактной операции.

Е.3.6 Определение сообщения об ошибке (см. 12.3.4, d)) является элементом **wsdl:fault** в абстрактной операции. Набор определений сообщений об ошибке является множеством всех элементов **wsdl:fault** в абстрактной операции.

E.3.6.1 Высокоуровневое **element declaration** определения сообщения об ошибке (см. 12.3.7) является глобальным **element declaration**, которое является значением атрибута **element** на единственном элементе **wsdl:part** в элементе **wsdl:message**, на который ссылается определение сообщения об ошибке (атрибут **message**).

Примечание — WSDL 1.1, 3.6 определяет следующие ограничения для определений сообщений об ошибке: элемент **wsdl:message** содержит только один элемент **wsdl:part**, а также элемент **wsdl:part**, ссылающийся на глобальное **element declaration** (с помощью атрибута **element**).

E.3.7 Форму определения входного или выходного сообщения (см. 12.3.6) задает привязка абстрактной операции (см. E.4.8).

Примечания

1 WS-I Basic Profile 1.0 (см. 5.3 и 5.3.1) устанавливает множество элементов **wsdl:part** элемента **wsdl:message**, на которое ссылается определение входного или выходного сообщения абстрактной операции с формой, соответствующей только той, что описана в 12.3.6, a) или b).

2 WS-I Basic Profile 1.0 ограничивает все абстрактные операции абстрактного интерфейса только лишь до абстрактных операций, описанных как документо-ориентированные (см. 12.3.8) или RPC-ориентированные (см. 12.3.9).

E.4 Интерфейсные и операционные привязки

E.4.1 Интерфейсная привязка (см. 12.4) представляет собой элемент **wsdl:binding** (в элементе **wsdl:definition**), который содержит элемент **soap:binding**. Набор интерфейсных привязок является множеством всех элементов **wsdl:binding** в элементе **wsdl:definition**.

E.4.2 URI транспорта (см. 12.4.2, c)) конкретного интерфейса есть значение атрибута **transport** на элементе **soap:binding**. Как указано в WS-I Basic Profile 1.0, 5.6.2 (требования R2702), только транспорт HTTP поддерживается и значением атрибута **transport** является «gost.e-ecolog.ru. Это значение определяет использование ASN.1 SOAP HTTP привязки (см. раздел 10) для интерфейса привязки ASN.1 SOAP (см. E.4.6).

E.4.3 Вид конкретного интерфейса — документ (см. 12.4.2, d)), если интерфейсная привязка соответствует точной привязке документа, как указано в WS-I Basic Profile 1.0, 5.3 и 5.3.1.

Примечание — WS-I Basic Profile 1.0 определяет документ-литеральную привязку как интерфейсную привязку с операционной привязкой, которые являются документ-литеральными операциями.

E.4.4 Вид конкретного интерфейса — RPC (см. 12.4.2, d)), если интерфейсная привязка точно соответствует rpc привязке, как указано в WS-I Basic Profile 1.0, 5.3 и 5.3.1.

Примечание — WS-I Basic Profile 1.0 определяет rpc-литеральную привязку как интерфейсную привязку с операционной привязкой, которые являются rpc-литеральными операциями.

E.4.5 Интерфейсная привязка опционально указывает, что конкретный интерфейс поддерживает быстрый веб-сервис (см. 12.4.2, e)) путем расширения WSDL 1.1, что называется аннотацией интерфейсной привязки ASN.1 SOAP. Аннотация есть EII, появляющийся как дочерний элемент **wsdl:binding** и после элемента **soapbind:binding**, со:

a) свойством **[local name]** «binding» и

b) свойством **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description**».

E.4.6 По умолчанию все конкретные интерфейсы поддерживают быстрые веб-сервисы и являются интерфейсными привязками ASN.1 SOAP (см. 12.4.7).

E.4.7 Интерфейсная привязка опционально указывает идентификатор объекта, присвоенный всем конкретным операциям (см. 12.4.2, a) и 12.4.3) путем расширения WSDL 1.1, который называется интерфейсной привязкой аннотации идентификатора объекта. Аннотация является All среди членов свойства **[attributes]** аннотации интерфейсной привязки ASN.1 SOAP (см. E.4.5) со:

a) свойством **[local name]** «object-identifier»;

b) свойством **[namespace name]** объекта «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description**»;

c) свойством **[normalized value]**, являющимся идентификатором объекта, закодированным как «XMLOIDValue», используя только «XMLNumberForm» (см. МСЭ-Т X.680 ISO/IEC 8824-1, п. 32).

E.4.8 Операционная привязка (см. 12.4.8) является элементом **wsdl:operation** в интерфейсной привязке. Набор операционных привязок является множеством всех элементов **wsdl:operation** в интерфейсной привязке.

E.4.9 Форма операционной привязки, как указано WS-I Basic Profile 1.0, 5.3 и 5.3.1, определяет форму соответствующих определений сообщений абстрактной операции (см. 12.3.6).

E.4.9.1 Если операционная привязка соответствует документ-литеральной операции, как указано WS-I Basic Profile 1.0, 5.3 и 5.3.1, то определение входного и выходного сообщений имеет вид, как указано в 12.3.6, a).

E.4.9.2 Для формы, указанной в 12.3.6, a), высокоуровневым **element declaration** является глобальное **element declaration**, которое является значением атрибута **element** элемента **wsdl:part**, на который явно или неявно

ссылается элемент **soap:body**. Никакого высокоуровневого **element declaration** не появляется, если нет элемента **wsdl:part**, на который ссылается элемент **soap:body**.

Примечание — Требования R2201, R2210, R2202, R2204, R2208 из WS-I Basic Profile 1.0 указывают, что есть ноль или один элемент **wsdl:part** и атрибут **element** присутствует (атрибут **type** отсутствует) на элементе (если имеется).

E.4.9.3 Если операционная привязка соответствует rpc-литеральной операции, как указано WS-I Basic Profile 1.0, 5.3 и 5.3.1, то определение входного и выходного сообщения имеет вид, как указано в 12.3.6, b).

E.4.9.4 Для формы, указанной в 12.3.6, b), неквалифицированное имя есть значение имени атрибута на ссылаемый элемент **wsdl:part** и связанный с высокоуровневым **complex type definition** или **simple type definition**, являющимся значением атрибута **type** на том же элементе **wsdl:part**. Список из нуля или более неквалифицированных имен соответствует именам, полученным (в том же порядке) из списка элементов **wsdl:part**, на которые явно или неявно ссылается (в порядке, предусмотренном WS-I Basic Profile 1.0, 5.4.1) элемент **soap:body**.

Примечание — Требования R2202, R2203, R2207, R2208 из WS-I Basic Profile 1.0 указывают, что существует список из нуля или более присутствующих элементов **wsdl:part** и атрибут **type** присутствует (атрибут **element** отсутствует) на элементе. Требование R2301 определяет порядок элементов **wsdl:part** в списке элементов, на который ссылается (явно или неявно) элемент **soap:body**.

E.4.10 SOAP действие URI (см. 12.4.8, a)) конкретной операции является значением атрибута **soapAction** на элементе **soap:operation** (если есть) в операционной привязке.

E.4.11 Определение блока заголовка SOAP (см. 12.4.8, b) и 12.4.11) заключается в следующем:

a) элемент **soap:header** либо в элементе **wsdl:input**, либо в элементе **wsdl:output** в операционной привязке,

b) элемент **soap:headerfault** в элементе **soap:header**.

E.4.11.1 Высокоуровневое **element declaration** определения блока заголовка SOAP является глобальным **element declaration**, которое является значением атрибута **element** на элементе **wsdl:part** в элементе **wsdl:message**, на что ссылается определение блока заголовка SOAP (соответственно атрибутами **part** и **message**).

Примечание — WS-I Basic Profile 1.0 указывает для определения блока заголовка SOAP, что атрибут **element** на элементе **wsdl:part** присутствует (а атрибут **type** отсутствует).

E.4.12 Идентификатор объекта может быть назначен как аннотация, которая является расширением WSDL 1.1, к высокоуровневому **element declaration** (см. 12.4.8, c)) для следующих определений:

a) определение входного или выходного сообщения как аннотация идентификатора объекта **element declaration** (см. E.4.13), что является атрибутом на определение входного или выходного сообщения соответственно;

b) определение сообщения об ошибке как аннотация идентификатора объекта **element declaration** (см. E.4.13), что является атрибутом на определение сообщения об ошибке, и

c) определение блока заголовка SOAP как аннотация идентификатора объекта **element declaration** (см. E.4.13), который является одним из членов свойства **[attributes]** аннотации блока заголовка SOAP (см. E.4.14).

E.4.13 Аннотация идентификатора объекта **element declaration** (расширение WSDL 1.1) является AII со:

a) свойством **[local name]** «object-identifier»;

b) свойством **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description**»;

c) свойством **[normalized value]**, которое является идентификатором объекта, закодированным как «XMLOIDValue», используя только «XMLNumberForm» (см. МСЭ-Т X.680 | ISO/IEC 8824-1, п. 32).

E.4.14 Аннотация блока заголовка SOAP (расширение WSDL 1.1) соответствует (при наличии) определению блока заголовка SOAP, которое присутствует в интерфейсной привязке. Аннотация — это EII, который является элементом в определении входного или выходного сообщения:

a) со свойством **[local name]** «header»;

b) со свойством **[namespace name]** объекта «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description**»;

c) с AII среди членов свойства **[attributes]** со:

- свойством **[local name]** «message»;
- свойством **[normalized value]**, равным значению атрибута **message** соответствующего определения блока заголовка SOAP;

d) с AII среди членов свойства **[attributes]** со:

- свойством **[local name]** «part»;
- свойством **[normalized value]**, равным значению атрибута **part** соответствующего определения блока заголовка SOAP.

E.4.15 По умолчанию высокоуровневые **element declaration** могут быть представлены в виде встроенных в ASN.1 значений (см. 12.4.8, d)).

E.4.16 Высокоуровневое **element declaration** может быть представлено в виде поддерева (см. 12.4.8, d)) путем включения аннотации, которая является расширением к WSDL 1.1 для следующих определений:

a) определение входного или выходного сообщения как аннотация поддерева **element declaration** (см. E.4.17), что является атрибутом определения входного или выходного сообщения соответственно;

b) определение сообщения об ошибке как аннотация поддерева **element declaration** (см. E.4.17), что является атрибутом определения сообщения об ошибке;

c) определение блока заголовка SOAP как аннотация поддерева **element declaration** (см. E.4.17), что является одним из членов свойства **[attributes]** аннотации блока заголовка SOAP (см. E.4.14).

E.4.17 Аннотация поддерева **element declaration** (расширение к WSDL 1.1) является All со:

a) свойством **[local name]** «subtree»;

b) свойством **[namespace name]** «**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description**»;

c) свойством **[normalized value]** «1» или «true».

E.4.17.1 Свойство **[normalized value]** аннотации поддерева элемента, которое является чем-то иным, кроме «1» или «true» (например, «0» или «false»), эквивалентно опусканию аннотации.

E.4.18 Высокоуровневое **element declaration** с аннотацией поддерева **element declaration** и аннотацией идентификатора объекта **element declaration** эквивалентно высокоуровневому **element declaration** с аннотацией только из аннотации поддерева **element declaration**.

Приложение F
(справочное)

Присвоение значений идентификаторов объектов

В настоящем стандарте присвоены следующие идентификаторы объектов и дескрипторы объектов:

```
{joint-iso-itu-t(2) asn1(1) generic-applications(10)
                    fast-web-services(1) modules(0) asn1soap(0)}
«ASN.1 SOAP Module»
```

Приложение ДА
(справочное)

Сведения о соответствии межгосударственных стандартов ссылочным международным стандартам (международным документам)

Таблица ДА.1

Обозначение и наименование международного стандарта	Степень соответствия	Обозначение и наименование межгосударственного стандарта
ISO/IEC 9834-1:2005 Информационные технологии. Процедуры для работы регистрационных органов по идентификации объекта. Часть 1. Общие процедуры и высшие разряды дерева идентификаторов объекта международного объекта	—	*
ISO/IEC 8824-1:2002 Информационные технологии. Нотация абстрактного синтаксиса версии 1 (ASN.1). Часть 1. Спецификация базовой нотации	—	*
ISO/IEC 8824-2:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 2. Спецификация информационных объектов	—	*
ISO/IEC 8824-3:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 3. Спецификация ограничений	—	*
ISO/IEC 8824-4:2002 Информационные технологии. Нотация абстрактного синтаксиса версии один (ASN.1). Часть 4. Спецификация для параметризации ASN.1	—	*
ISO/IEC 8825-1:2002 Информационные технологии. Правила кодирования ASN.1. Часть 1. Спецификация основных (BER), канонических (CER) и различительных правил кодирования (DER)	—	*
ISO/IEC 8825-2:2002 Информационные технологии. Правила кодирования ASN.1. Часть 2. Спецификация правил уплотненного кодирования (PER)	—	*
ISO/IEC 8825-3:2002 Информационные технологии. Правила кодирования ASN.1. Часть 3. Спецификация нотации контроля кодирования (ECN)	—	*
ISO/IEC 8825-4:2002 Информационные технологии. Правила кодирования ASN.1. Часть 4. Правила кодирования XML (XER)	—	*
ISO/IEC 8825-5:2004 Информационные технологии. Правила кодирования ASN.1. Часть 5. Определения схемы отображения W3C XML в ASN.1	—	*
ISO/IEC 24824-1:2005 Информационные технологии. Общие правила применения ASN.1: Быстрые команды. Часть 1	IDT	ГОСТ ISO/IEC 24824-1-2013 Информационные технологии. Общие правила применения ASN.1. Быстрые команды. Часть 1

* Соответствующий межгосударственный стандарт отсутствует. До его утверждения рекомендуется использовать перевод на русский язык данного международного стандарта. Перевод данного международного стандарта находится в Федеральном информационном фонде технических регламентов и стандартов.

Примечание — В настоящей таблице использовано следующее условное обозначение степени соответствия стандартов:

- IDT — идентичные стандарты.

Библиография

[1] W3C Note, Simple Object Access Protocol (SOAP) 1.1 (Рекомендация W3C, Простой протокол доступа к объекту (SOAP) 1.1) Don Box, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Nielsen, Satish Thatte, Dave Winer, W3C Note, 8 May 2000 (см. gost.e-ecolog.ru .)

[2] W3C Note, Web Services Description Language (WSDL) 1.1 (Рекомендация W3C, Язык описания веб-сервисов (WSDL) 1.1), Erik Christensen, Francisco Curbera, Greg Meredith, Sanjiva Weerawarana, W3C Note, 15 March 2001 (см. gost.e-ecolog.ru .)

[3] WS-I, WS-I Basic Profile Version 1.0 (Web Services Interoperability Organization, Организация по развитию возможности взаимодействия веб-сервисов, Базовый Профиль WS-I версии 1.0), Final Material, 16 April 2004 (см. gost.e-ecolog.ru .)

УДК 681.3:691.39:006.354 МКС 35.100.60 IDT

Ключевые слова: обработка данных, информационный обмен, сетевое взаимодействие, взаимосвязь открытых систем, ASN.1, кодирование, быстрый инфо-набор

Редактор *Н.Н. Кузьмина*
Технический редактор *В.Н. Прусакова*
Корректор *С.И. Фирсова*
Компьютерная верстка *Е.Е. Кругова*

Сдано в набор 10.09.2015. Подписано в печать 05.11.2015. Формат 60 ×84 1/8. Гарнитура Ариал
Усл. печ. л. 5,12. Уч.-изд. л. 4,70. Тираж 33 экз. Зак. 3493

Издано и отпечатано во ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4
gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru