

国际电信联盟

**ITU-T**

国际电信联盟  
电信标准化部门

**X.892**

(05/2005)

X系列：数据网、开放系统通信和安全性

OSI应用 — ASN.1的一般应用

---

**信息技术 — ASN.1的一般应用：快速Web服务**

ITU-T X.892建议书



ITU-T X系列建议书  
数据网、开放系统通信和安全性

公众数据网	
业务和设施	X.1-X.19
接口	X.20-X.49
传输、信令和交换	X.50-X.89
网络概貌	X.90-X.149
维护	X.150-X.179
管理安排	X.180-X.199
开放系统互连	
模型和记法	X.200-X.209
服务限定	X.210-X.219
连接式协议规范	X.220-X.229
无连接式协议规范	X.230-X.239
PICS书写形式	X.240-X.259
协议标识	X.260-X.269
安全协议	X.270-X.279
层管理对象	X.280-X.289
一致性测试	X.290-X.299
网间互通	
概述	X.300-X.349
卫星数据传输系统	X.350-X.369
以IP为基础的网络	X.370-X.379
报文处理系统	X.400-X.499
号码簿	X.500-X.599
OSI 组网和系统概貌	
组网	X.600-X.629
效率	X.630-X.639
业务质量	X.640-X.649
命名、寻址和登记	X.650-X.679
抽象句法记法1 (ASN.1)	X.680-X.699
OSI 管理	
系统管理框架和结构	X.700-X.709
管理通信服务和协议	X.710-X.719
管理信息的结构	X.720-X.729
管理功能	X.730-X.799
安全	X.800-X.849
OSI 应用	
托付、并发和恢复	X.850-X.859
事务处理	X.860-X.879
远程操作	X.880-X.889
<b>ASN.1的一般应用</b>	<b>X.890-X.899</b>
开放分布式处理	X.900-X.999
电信安全	X.1000-X.1999

欲了解更详细信息，请查阅ITU-T建议书目录。

## 信息技术 — ASN.1的一般应用：快速Web服务

### 摘 要

本建议书 | 国际标准规定了快速 Web 服务所需的消息。规定了承载与 W3C SOAP 消息相同语义的 ASN.1 SOAP 消息。ASN.1 SOAP 消息的交互提供了快速 Web 服务。

本规范引用了其他的 ASN.1 建议书 | 国际标准和 Web 服务规范《W3C SOAP 1.2》来完整地定义使用 Web 协议和 W3C SOAP 功能性的应用交互。这是对使用 W3C SOAP 来提供 Web 服务的一种扩展，而无需对 W3C SOAP 功能和服务描述语言进行任何改变。主要变动在于使用压缩的和易处理的 XML 数据二进制编码，而不是字符编码。

本建议书 | 国际标准（使用 ASN.1 分组编码规则（PER）编码的 ASN.1 SOAP 消息）的使用允许采用相对于字符编码的 XML 数据而言需要占用更少的网络带宽和处理能力的消息（因此可以提供更高的事务处理速率）来提供 Web 服务的应用。

本建议书 | 国际标准也规定了一种多用途互联网邮件扩展（MIME）的媒体类型，以标识采用分组编码规则编码的 ASN.1 SOAP 消息。还规定了一种多用途互联网邮件扩展的媒体类型来标识采用快速信息集文档（见 ITU-T X.891 建议书 | ISO/IEC 24824-1）编码的 W3C SOAP 信息集。这两种多用途互联网邮件扩展类型在本建议书 | 国际标准中都会使用。

### 来 源

ITU-T 第 17 研究组（2005-2008）按照 ITU-T A.8 建议书规定的程序，于 2005 年 5 月 14 日批准了 ITU-T X.892 建议书。相同的文本也做为 ISO/IEC 24824-2 出版。

## 前 言

国际电信联盟（ITU）是从事电信领域工作的联合国专门机构。ITU-T（国际电信联盟电信标准化部门）是国际电信联盟的常设机构，负责研究技术、操作和资费问题，并且为在世界范围内实现电信标准化，发表有关上述研究项目的建议书。

每四年一届的世界电信标准化全会（WTSA）确定 ITU-T 各研究组的研究课题，再由各研究组制定有关这些课题的建议书。

WTSA 第 1 号决议规定了批准建议书须遵循的程序。

属 ITU-T 研究范围的某些信息技术领域的必要标准，是与国际标准化组织（ISO）和国际电工技术委员会（IEC）合作制定的。

## 注

本建议书为简明扼要起见而使用的“主管部门”一词，既指电信主管部门，又指经认可的运营机构。

遵守本建议书的规定是以自愿为基础的，但建议书可能包含某些强制性条款（以确保例如互操作性或适用性等），只有满足所有强制性条款的规定，才能达到遵守建议书的目的。“应该”或“必须”等其它一些强制性用语及其否定形式被用于表达特定要求。使用此类用语不表示要求任何一方遵守本建议书。

## 知识产权

国际电联提请注意：本建议书的应用或实施可能涉及使用已申报的知识产权。国际电联对无论是其成员还是建议书制定程序之外的其它机构提出的有关已申报的知识产权的证据、有效性或适用性不表示意见。

至本建议书批准之日止，国际电联已经收到实施本建议书可能需要的受专利保护的知识产权的通知。但需要提醒实施者注意的是，这可能并非最新信息，因此特大力提倡他们通过下列网址查询电信标准化局（TSB）的专利数据库：<http://www.itu.int/ITU-T/ipr/>。

© 国际电联 2005

版权所有。未经国际电联事先书面许可，不得以任何手段复制本出版物的任何部分。

# 目 录

	页码
1 范围 .....	1
2 规范性参考文献 .....	1
2.1 等同的建议书   国际标准 .....	1
2.2 补充参考文献 .....	2
3 定义 .....	3
3.1 引入的定义 .....	3
3.2 补充定义 .....	3
4 缩写词 .....	4
5 符号 .....	4
6 ASN.1 SOAP 消息的处理 .....	5
7 Envelope 类型的组件映射到信息项 .....	7
7.1 概述 .....	7
7.2 Header 类型的映射 .....	7
7.3 Body 类型的映射 .....	7
7.4 Fault 类型的映射 .....	7
7.5 Content 类型的映射 .....	9
8 将 W3C SOAP 消息信息集映射到 Envelope 类型的抽象值 .....	10
8.1 概述 .....	10
8.2 Header 元素信息项的映射 .....	10
8.3 Body 元素信息项的映射 .....	11
8.4 Fault 元素信息项的映射 .....	11
8.5 内容元素信息项到 Content 类型取值的映射 .....	12
9 嵌入式 ASN.1 编码值的扩展 SOAP 处理 .....	13
9.1 概述 .....	13
9.2 嵌入式 ASN.1 编码值的 ASN.1 类型的识别 .....	13
9.3 从标识的嵌入式 ASN.1 编码值生成 ASN.1 值 .....	14
9.4 将 ASN.1 值（带标识符）插入 W3C SOAP 消息 .....	14
9.5 “ASN.1 类型不可识别”故障 .....	15
10 ASN.1 SOAP 的 HTTP 绑定 .....	16
10.1 HTTP 媒体类型 .....	16
10.2 响应 SOAP 节点的行为 .....	16
11 快速信息集 SOAP 消息和 SOAP HTTP 绑定 .....	16
12 支持 ASN.1 SOAP 接口绑定的面向 SOAP 的服务描述 .....	17
12.1 概述 .....	17
12.2 模式（Schemas） .....	17
12.3 抽象接口和抽象操作 .....	17
12.4 接口绑定和操作绑定 .....	18
12.5 RPC 模式 .....	19
13 使用带有 ASN.1 SOAP 接口绑定的面向 SOAP 的服务描述 .....	21
附件 A — 用于 ASN.1 SOAP 的 ASN.1 模块 .....	23
附件 B — 快速 Web 服务的多用途互联网邮件扩展媒体类型 .....	25
B.1 “application/fastsoap” 媒体类型 .....	25
B.2 “application/soap+fastinfoset” 媒体类型 .....	26
附件 C — 快速 Web 服务简介 .....	28
C.1 快速 Web 服务的优点 .....	28
C.2 ASN.1 SOAP 消息的概念上的和优化的处理 .....	29
C.3 服务描述 .....	32

	页码
附件 D — 使用快速 Web 服务和 XML Web 服务的公共服务提供.....	34
D.1    乐观策略.....	34
D.2    悲观策略.....	34
附件 E — WSDL 1.1 中面向 SOAP 的服务描述.....	36
E.1    WSDL 1.1 中表示的面向 SOAP 的服务描述.....	36
E.2    模式 (Schema).....	36
E.3    抽象接口和抽象操作.....	36
E.4    接口绑定和操作绑定.....	37
附件 F — 对象标识符值的分配.....	40
参考资料.....	41

## 引言

本建议书 | 国际标准规定了 ASN.1（见 ITU-T X.680 建议书 | ISO/IEC 8824-1）的使用、其压缩的编码规则（见 ITU-T X.691 建议书 | ISO/IEC 8825-2）和快速信息集（见 ITU-T X.891 建议书 | ISO/IEC 24824-1）以提供快速 Web 服务。（快速 Web 服务的一般介绍，参见附件 C）。

第 6 节描述了产生和处理采用 ASN.1 二进制编码的 SOAP 消息（被称为“ASN.1 SOAP 消息”）的体系结构模型和概念上的步骤。

第 7 节至第 9 节包含了处理 ASN.1 SOAP 消息的一般规定。第 7 节描述了 ASN.1 SOAP 消息到 W3C SOAP 消息的映射。第 8 节描述了 W3C SOAP 消息到 ASN.1 SOAP 消息的映射。第 9 节描述了用于处理 W3C SOAP 消息中嵌入式 ASN.1 编码值的扩展 SOAP 处理模型。

第 10 节描述了以 HTTP 作为传输协议来传送 ASN.1 SOAP 消息所需的 ASN.1 SOAP 的 HTTP 绑定。该绑定使用了第 B.1 节中所述的多用途互联网邮件扩展（MIME）媒体类型。

第 11 节描述了传送以快速信息集文档（快速信息集 SOAP 消息）编码的 W3C SOAP 消息所需的 W3C SOAP 的 HTTP 绑定。该绑定使用了第 B.2 节中所述的多用途互联网邮件扩展（MIME）媒体类型。

第 12 节描述了支持 ASN.1 SOAP 绑定接口和快速 Web 服务的面向 SOAP 的服务描述。

第 13 节描述了一种面向 SOAP 的服务描述如何影响与 W3C SOA 消息进行相互映射的 ASN.1 SOAP 信息的交互。

附件 A 是本建议书 | 国际标准的组成部分，其中包括了用于 ASN.1 SOAP 的全部 ASN.1 模块。

附件 B 是本建议书 | 国际标准的组成部分，其中包括了“**application/fastsoap**”和“**application/soap+fastinfoset**”的媒体类型。

附件 C 不是本建议书 | 国际标准的组成部分，其中提供了快速 Web 服务的介绍材料。

附件 D 不是本建议书 | 国际标准的组成部分，其中提供了使用 ASN.1 SOAP 的 HTTP 绑定的特性来进行快速 Web 服务与 XML Web 服务之间互操作的介绍材料。

附件 E 不是本建议书 | 国际标准的组成部分，其中说明了如何采用 WSDL 1.1[2]服务描述来表示 ASN.1 SOAP 的消息交互。



## 国际标准 ITU-T X.892建议书

### 信息技术 — ASN.1的一般应用: 快速Web服务

#### 1 范围

本建议书 | 国际标准规定了使用快速 Web 服务所需要的消息和编码以及描述这些服务的方法。

支撑这些服务的协议满足 SOAP 处理模型的需求（见《W3C SOAP Part 1》，第 2 节），并基于对如下信息的传送：

- a) 包含嵌入式 ASN.1 编码值和嵌入式快速信息集文档的 ASN.1 SOAP 消息；和
- b) 快速信息集 SOAP 消息。

本建议书 | 国际标准也规定：

- 为 ASN.1 SOAP 定义了 **Envelope** 类型（该类型的取值与一个 ASN.1 SOAP 消息相对应）的模块；
- ASN.1 SOAP 消息与 W3C SOAP 消息（定义为 XML 信息集的一个实例，见《W3C SOAP Part 1》，第 5 节）之间的概念映射；
- 一种对于 W3C SOAP 处理模型的扩展，用于处理嵌入式 ASN.1 编码值；
- ASN.1 SOAP 的 HTTP 绑定，这是对 W3C SOAP 的 HTTP 绑定（见《W3C SOAP Part 2》，第 7 节）的修改与扩展，以便传送 ASN.1 SOAP 消息；
- 对使用 W3C SOAP 的 HTTP 绑定（见《W3C SOAP Part 2》，第 7 节）来传送串行化为快速信息集文档（快速信息集 SOAP 消息）的 W3C SOAP 消息信息集的支持；
- 面向 SOAP 的服务描述，以定义对于快速 Web 服务的接口及其语义。

分配了两个多用途互联网邮件扩展（MIME）媒体类型名以标识：

- 使用基本对齐分组编码规则编码的 ASN.1 SOAP 消息；
- 快速信息集 SOAP 消息。

#### 2 规范性参考文献

下列建议书和国际标准的条款，在本建议书 | 国际标准的引用而构成本建议书 | 国际标准的条款。在出版时，所指出的版本是有效的。所有的建议书和国际标准均会得到修订，本建议书 | 国际标准的使用者应查证是否有可能使用下列建议书和国际标准最新版本。IEC 和 ISO 的各成员有目前有效的国际标准的目录。国际电联电信标准化局有目前有效的 ITU-T 建议书的清单。IETF 维护着一个 RFC 的列表，以及那些已经被新的 RFC 所废止的旧 RFC。

本建议书 | 国际标准中引用的文件自成一体时不具备建议书或国际标准的地位。

##### 2.1 等同的建议书 | 国际标准

- ITU-T Recommendation X.660 (2004) | ISO/IEC 9834-1:2005, *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.*
- ITU-T Recommendation X.680 (2002) | ISO/IEC 8824-1:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation.*

- ITU-T Recommendation X.681 (2002) | ISO/IEC 8824-2:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Information object specification.*
- ITU-T Recommendation X.682 (2002) | ISO/IEC 8824-3:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Constraint specification.* †
- ITU-T Recommendation X.683 (2002) | ISO/IEC 8824-4:2002, *Information technology – Abstract Syntax Notation One (ASN.1): Parameterization of ASN.1 specifications.* †
- ITU-T Recommendation X.690 (2002) | ISO/IEC 8825-1:2002, *Information technology – ASN.1 encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER), and Distinguished Encoding Rules (DER).* †
- ITU-T Recommendation X.691 (2002) | ISO/IEC 8825-2:2002, *Information technology – ASN.1 encoding rules: Specification of Packed Encoding Rules (PER).*
- ITU-T Recommendation X.692 (2002) | ISO/IEC 8825-3:2002, *Information technology – ASN.1 encoding rules: Specification of Encoding Control Notation (ECN).* †
- ITU-T Recommendation X.693 (2001) | ISO/IEC 8825-4:2002, *Information technology – ASN.1 encoding rules: XML Encoding Rules (XER) plus Amendment 1: XER Encoding Instructions and EXTENDED-XER.* †
- ITU-T Recommendation X.694 (2004) | ISO/IEC 8825-5:2004, *Information technology – ASN.1 encoding rules: Mapping W3C XML Schema Definitions into ASN.1.*
- ITU-T Recommendation X.891 (2005) | ISO/IEC 24824-1:2005, *Information technology – Generic Applications of ASN.1: Fast Infoset.*

注 — 完整的 ASN.1 建议书 | 国际标准集如上所列，因为它们都能应用于本建议书 | 国际标准的特定使用。对于其中没有直接在本建议书 | 国际标准的正文中引用的文献，其后标有†符号。

## 2.2 补充参考文献

- 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), <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>.
  - 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), <http://www.w3.org/TR/2003/REC-soap12-part2-20030624>.
  - 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), <http://www.w3.org/TR/2000/REC-xml-20040204/>.
  - 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), <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), <http://www.w3.org/TR/1999/REC-xml-names-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), <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), <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>.
- 注 — 当参考文献《W3C XML Schema》在本建议书 | 国际标准中被引用时，它是指《W3C XML Schema Part 1》和《W3C XML Schema Part 2》。
- IETF RFC 2045 (1996), *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies.*

– IETF RFC 2616 (1999), *Hypertext Transfer Protocol – HTTP/1.1*.

### 3 定义

本建议书 | 国际标准采用了以下定义。

#### 3.1 引入的定义

**3.1.1** 本建议书 | 国际标准使用了在 ITU-T X.680 建议书 | ISO/IEC 8824-1 中定义的如下术语：

- a) 抽象值；
- b) 模块；
- c) 对象标识符；
- d) 相对对象标识符；
- e) 类型。

**3.1.2** 本建议书 | 国际标准也使用了定义在《W3C XML Schema》中的如下术语：

- a) **complex type definition** (复杂类型定义)；
- b) **element declaration** (元素声明)；
- c) 模式；
- d) 模式组件；
- e) **simple type definition** (简单类型定义)。

**3.1.3** 本建议书 | 国际标准也使用了定义在《W3C XML Information Set》中的如下术语：

- a) 抽象信息项；
- b) 字符信息项；
- c) 元素信息项；
- d) 信息项；
- e) 名空间信息项；
- f) (一个信息项的) 特性。

**3.1.4** 本建议书 | 国际标准也使用了定义在《W3C SOAP Part 1》第 1.5.1 节中的如下术语：

- a) SOAP；
- b) SOAP 绑定；
- c) SOAP 消息交换方式 (MEP)；
- d) SOAP 节点。

**3.1.5** 本建议书 | 国际标准也使用了定义在 ITU-T X.891 建议书 | ISO/IEC 24824-1 中的如下术语：

- a) Base64；
- b) 快速信息集文档；
- c) XML 信息集。

#### 3.2 补充定义

**3.2.1 ASN.1 SOAP interface binding ASN.1 SOAP 接口绑定：** 一个服务描述 (见第 12.4 节) 的具体接口，描述了将通过 ASN.1 SOAP 消息交互来提供的快速 Web 服务的语义。

**3.2.2 ASN.1 SOAP endpoint ASN.1 SOAP 端点：** 在服务描述中所标识的一个快速 Web 服务的网络位置。

**3.2.3 ASN.1 SOAP header block ASN.1 SOAP 头部块：** 一个 **HeaderBlock** 类型的取值 (见附件 A)。

**3.2.4 ASN.1 SOAP HTTP binding ASN.1 SOAP 的 HTTP 绑定：** 为传送 ASN.1 SOAP 消息而从 SOAP 到 HTTP 的绑定。

- 3.2.5 ASN.1 SOAP message ASN.1 SOAP 消息：**一个从 W3C SOAP 消息映射而来的 **Envelope** 类型的取值（见第 8 节）。
- 3.2.6 embedded ASN.1 encoded value 嵌入式 ASN.1 编码的取值：**一个 ASN.1 类型的抽象值，并作为一个 Base64 编码的字符串被包含在一个 W3C SOAP 消息中。
- 3.2.7 embedded fast infoset document 嵌入式快速信息集文档：**指一个元素信息项，当被包含在一个 ASN.1 SOAP 消息中时，它被编码为一个快速信息集文档。
- 3.2.8 fast-enabled web service client 支持快速 web 服务的客户端：**指一个既可以使用 ASN.1 SOAP 消息又可以使用 XML SOAP 消息来发送请求并接收响应的 SOAP 节点。
- 3.2.9 fast infoset SOAP message 快速信息集 SOAP 消息：**串行化为一个快速信息集文档的一条 W3C SOAP 消息。
- 3.2.10 fast web services 快速 web 服务：**通过交互 ASN.1 SOAP 消息来提供的服务。
- 3.2.11 service description 服务描述：**用于描述对一个 Web 服务的接口及其语义的一组文档。
- 3.2.12 W3C SOAP header block W3C SOAP 头部块：**指定义在《W3C SOAP Part 1》第 1.5.2 节中的“SOAP 头部块”。
- 3.2.13 W3C SOAP message W3C SOAP 消息：**指定义在《W3C SOAP Part 1》第 1.5.2 节中的“SOAP 消息”。
- 3.2.14 W3C SOAP namespace W3C SOAP 名空间：**名为“<http://www.w3.org/2003/05/soap-envelope>”的名空间（见《W3C SOAP Part 1》第 1 部分，第 1.1 节）。
- 3.2.15 XML web services XML web 服务：**通过交互 XML SOAP 消息来提供的服务。
- 3.2.16 XML SOAP message XML SOAP 消息：**一个 W3C SOAP 消息或由任何以前或以后版本的 SOAP 所定义的消息，它串行化为一个 XML 文档。

## 4 缩写词

本建议书|国际标准使用下列缩写词：

AII	属性信息项（见《W3C XML Information Set》，第 2.3 节）
ASN.1	抽象句法符号 1
CII	字符信息项（见《W3C XML Information Set》，第 2.6 节）
EII	元素信息项（见《W3C XML Information Set》，第 2.2 节）
HTTP	超文本传送协议（见《IETF RFC 2616》）
MIME	多用途互联网邮件扩展
NII	名空间信息项（见《W3C XML Information Set》，第 2.11 节）
PER	ASN.1 的分组编码规则
RPC	远端方法调用
URI	一致性资源标识符
WSDL	Web 服务描述语言
XML	可扩展标记语言
XSD	W3C XML 模式

## 5 符号

- 5.1** 本建议书|国际标准使用在 ITU-T X.680 建议书|ISO/IEC 8824-1 中定义的 ASN.1 符号。
- 5.2** 在本建议书|国际标准中，**粗体 Courier** 字体用于表示 ASN.1 符号。
- 5.3** 对如下的符号，将使用**粗体 Arial** 字体来表示：
- XML 句法；
  - 元素信息项和属性信息项名称；
  - HTTP 头部域和 HTTP 头部域的参数。

5.4 信息项特性的名称将使用**粗体 Arial** 字体，并置于一对方括号之间（例如，**[children]**特性）。

5.5 多用途互联网邮件扩展媒体类型和 URI 将使用**粗体 Arial** 字体，并置于一对双引号之间（例如，URI “**http://www.w3.org/2003/05/soap-envelope**”）。

## 6 ASN.1 SOAP 消息的处理

6.1 ASN.1 SOAP 消息是指定义在 ASN.1 模块 **ASN1SOAP** 中的 **Envelope** 类型的抽象值（见附件 A）。**Envelope** 类型的抽象值在语义上与在《W3C SOAP Part 1》第 5 节中所述的 XML 信息集（被称为 W3C SOAP 消息信息集）的实例等价。

注 — **Envelope** 类型可对 W3C SOAP 消息信息集采用一种优化的二进制编码。

6.2 ASN.1 SOAP 消息可与 Web 服务描述一起使用或独立于任何 Web 服务描述而使用。使用 XML SOAP 消息的一个 Web 服务描述无需任何改变即可提供使用 ASN.1 SOAP 消息的快速 Web 服务描述（见附件 E）。

6.3 SOAP 处理模型、扩展模型和绑定模型（见《W3C SOAP Part 1》第 2、3、4 节）需由 SOAP 节点通过第 6.4 节中所描述的 **Envelope** 类型的组件与 W3C SOAP 消息信息集中的信息项之间的映射，应用于 **Envelope** 类型的抽象值。

6.4 这些 SOAP 模型到 **Envelope** 类型的抽象值的应用必须是下列概念步骤后的结果：

- a) **Envelope** 类型组件的抽象值（一个 ASN.1 SOAP 消息）映射为一个 W3C SOAP 消息信息集中的信息项，如第 7 节表 1 所示；
- b) 将 SOAP 模型应用于该信息集（见《W3C SOAP Part 1》，第 2、3、4 节），通常会产生一个新的 W3C SOAP 消息信息集，这个新的信息集符合《W3C SOAP Part 5》并受限于第 6.6 节所述的内容；和
- c) 该新 W3C SOAP 消息信息集的信息项又按第 8 节表 1 所述重新映射为一个 **Envelope** 类型组件的抽象值，而通常为该 **Envelope** 类型产生一个新的抽象值（一个新的 ASN.1 SOAP 消息）。

注 — 这三步只是概念上的。并不需要某实现产生一个 W3C SOAP 消息信息集的实际表示。W3C SOAP 消息信息集和 ASN.1 SOAP 消息都是抽象值，独立于用于在计算系统中进行的表示或在系统间进行传送的任何串行化或编码。

6.5 将 SOAP 模型应用于 W3C SOAP 消息信息集（见第 6.4 节 b）须包括对嵌入式 ASN.1 编码值的扩展处理，如第 9 节所述。

6.6 如下的约束适用于由第 6.4 节 b 所涉及的转换结果而得到的 W3C SOAP 消息信息集：

- a) 属性信息项不能出现在 **Body** 元素信息项和 **Detail** 元素信息项的**[attributes]**特性的成员中；和
- b) 至少有一个元素信息项必须出现在 **Body** 元素信息项和 **Detail** 元素信息项的**[children]**特性的成员中。

6.7 该 **Envelope** 类型的一个组件（在直到出现一个 **Content** 类型取值的任意层深度）须映射为一个信息项（或反向映射），如表 1 所示。表 1 的第 1 列列出了 **Envelope** 类型的组件。第 2 列给出了对《W3C SOAP Part 1》中描述那些在语义上等价的信息项的相关子节的引用。第 3 列列出了本建议书 | 国际标准中所描述的从组件映射到语义上等价的信息项的章节。第 4 列列出了本建议书 | 国际标准中描述从语义上等价的信息项映射到组件的章节。

表 1—Envelope 类型组件和 W3C SOAP 消息信息集中的信息项之间的映射

用于ASN.1 SOAP的ASN.1模块	《W3C SOAP Part 1》中的引用	映射自ASN.1	映射到ASN.1
<b>Envelope ::= SEQUENCE {</b>	5.1	第 7 节	第 8 节
<b>header Header,</b>	5.2	第 7.1.3 节	第 8.1.2 节
<b>body-or-fault CHOICE {</b> <b>body Body,</b> <b>fault Fault</b>	5.3, 5.4	第 7.1.4 和 7.1.5 节	第 8.1.3 和 8.1.4 节
<b>}</b> <b>}</b>			
<b>Header ::= SEQUENCE OF HeaderBlock</b>	5.2	第 7.2 节	第 8.2 节
<b>HeaderBlock: ::= SEQUENCE {</b>	5.2.1	第 7.2.2 节	第 8.2.2 节
<b>mustUnderstand BOOLEAN OPTIONAL,</b>	5.2.2	第 7.2.2.1 节	第 8.2.2.1 节
<b>relay BOOLEAN OPTIONAL,</b>	5.2.3	第 7.2.2.2 节	第 8.2.2.2 节
<b>role XSD.AnyURI</b> <b>DEFAULT ultimateReceiver,</b>	5.2.4	第 7.2.2.3 节	第 8.2.2.3 节
<b>content Content</b>		第 7.2 节	第 8.2 节
<b>}</b>			
<b>Body ::= SEQUENCE {</b>	5.3	第 7.3 节	第 8.3 节
<b>content Content OPTIONAL</b>	5.3	第 7.3.2 节	第 8.3.2 节
<b>}</b>			
<b>Fault ::= SEQUENCE {</b>	5.4	第 7.4 节	第 8.4 节
<b>code Code,</b>	5.4.1	第 7.4.1.2 节	第 8.4.1.2 节
<b>reason SEQUENCE SIZE(1..MAX) OF Text,</b>	5.4.2	第 7.4.1.3 节	第 8.4.1.3 节
<b>node XSD.AnyURI OPTIONAL,</b>	5.4.3	第 7.4.1.4 节	第 8.4.1.4 节
<b>role XSD.AnyURI OPTIONAL,</b>	5.4.4	第 7.4.1.5 节	第 8.4.1.5 节
<b>detail Content</b>	5.4.5	第 7.4.1.6 节	第 8.4.1.6 节
<b>}</b>			
<b>Code ::= SEQUENCE {</b>	5.4.1	第 7.4.2 节	第 8.4.2 节
<b>value Value,</b>	5.4.1.1	第 7.4.2.2 节	第 8.4.2.2 节
<b>subcodes SEQUENCE OF XSD.QName</b>	5.4.1.2, 5.4.1.3	第 7.4.2.3 和 7.4.2.4 节	第 8.4.2.3 和 8.4.2.4 节
<b>}</b>			
<b>Value ::= ENUMERATED {</b> <b>versionMismatch,</b> <b>mustUnderstand,</b> <b>dataEncodingUnknown,</b> <b>sender,</b> <b>receiver</b>	5.4.1.1, 5.4.8	第 7.4.3 节	第 8.4.3 节
<b>}</b>			
<b>Text ::= SEQUENCE {</b>	5.4.2.1	第 7.4.4 节	第 8.4.4 节
<b>lang XSD.Language,</b>		第 7.4.4.2 节	第 8.4.4.2 节
<b>text UTF8String</b>		第 7.4.4.3 节	第 8.4.4.3 节
<b>}</b>			
<b>Content ::= ....</b>	不适用	第 7.5 节	第 8.5 节

## 7 Envelope类型的组件映射到信息项

### 7.1 概述

7.1.1 一个 **Envelope** 元素信息项须自一个 **Envelope** 类型的取值生成。

7.1.2 一个名空间信息项的惟一 **[prefix]**特性，并带有一个 **[namespace name]**特性，该名空间特性等于 **Envelope** 元素信息项的 **[in-scope namespaces]**特性的成员中的 W3C SOAP 名空间的名称，在生成时其取值须由该 SOAP 节点来选择。

注 1 — 在《W3C SOAP Part 1》第 1.1 节中通常使用前缀“**env**”，但可以使用任何前缀。

注 2 — 所有定义在 SOAP 中的元素信息项和属性信息项都有一个 **[namespace name]**特性，并与《W3C SOAP 1》第 1.1 节中所述的 W3C SOAP 名空间的名字相同。

7.1.3 **header** 组件的取值须按第 7.2 节中所述进行映射。

7.1.4 如果 **body-or-fault** 组件的取值有 **body** 选项存在，则该选项须映射为 **Body** 元素信息项，如第 7.3 节所述。

7.1.5 如果 **body-or-fault** 组件的取值有 **fault** 选项存在，则须生成 **Body** 元素信息项并且该选项须映射为一个 **Fault** 元素信息项，如第 7.4 节所述。

注 — 一个包含故障信息的 W3C SOAP 消息只能有一个 **Fault** 元素信息项作为 **Body** 元素信息项的孩子（不能有其他的子元素信息项）。ASN.1 模式通过对 **body-or-fault** 的选择提供独立的 **body-or-fault** 选项来反映这些限制。

### 7.2 Header类型的映射

7.2.1 一个 **Header** 元素信息项须自一个 **Header** 类型的取值生成。如果该 **Header** 类型包含 **HeaderBlock** 一到多个出现次数，则 **HeaderBlock** 的每次出现须按顺序映射为该 **Header** 元素信息项的一个子元素信息项，如第 7.2.2 节所述。如果未出现 **HeaderBlock**，则不生成 **Header** 元素信息项。

7.2.2 一个 **content** 组件的取值须映射为一个 W3C SOAP 头部块，如第 7.5 节所述。在由第 7.5 节生成的元素信息项的 **[attributes]**特性的成员中所附加的属性信息项，须按第 7.2.2.1 节到第 7.2.2.3 节所述来生成。

7.2.2.1 如果 **mustUnderstand** 组件的取值存在且不为 **FALSE**，则该 **mustUnderstand** 属性信息项须自该取值生成，且 **mustUnderstand** 属性信息项的 **[normalized value]**特性须为“1”。否则不生成 **mustUnderstand** 属性信息项。

7.2.2.2 如果 **relay** 组件的取值存在且不为 **FALSE**，则 **relay** 属性信息项须自该取值生成，且该 **relay** 属性信息项的 **[normalized value]**特性须为“1”。否则不生成 **relay** 属性信息项。

7.2.2.3 如果 **role** 组件的取值不是 **ultimateReceiver**，则 **role** 属性信息项须生成自该取值生成，且该 **role** 属性信息项的 **[normalized value]**特性须为该 **role** 组件的字符串取值。否则不得生成 **role** 属性信息项。

### 7.3 Body类型的映射

7.3.1 一个 **Body** 元素信息项须自一个 **Body** 类型的取值生成。

7.3.2 **content** 组件（如果存在）的取值须按第 7.5 节所述进行映射。

### 7.4 Fault类型的映射

#### 7.4.1 概述

7.4.1.1 一个 **Fault** 元素信息项须自一个 **Fault** 类型的取值生成。

7.4.1.2 一个 **code** 组件的取值须按第 7.4.2 节所述进行映射。

7.4.1.3 **Reason** 元素信息项须自一个 **reason** 组件的取值生成。在 **sequence-of** 中每次出现的 **Text**，按顺序映射为 **Reason** 元素信息项的子 **Text** 元素信息项，如第 7.4.4 节所述。

注 — 建议所有 **sequence-of** 中出现的 **Text** 都有惟一的 **lang** 组件值（见《W3C SOAP Part 1》，第 5.4.2 节）。

7.4.1.4 **Node** 元素信息项须自一个 **node** 组件（若存在）的取值生成，且该 **Node** 元素信息项须将该 **node** 组件的字符串值中的字符作为其 **child** 字符信息项。

**7.4.1.5 Role** 元素信息项须自一个 **role** 组件（若存在）的取值生成，且该 **Role** 元素信息项须将该 **role** 组件的字符串值中的字符作为其 **child** 字符信息项。

**7.4.1.6 Detail** 元素信息项须自一个 **detail** 组件（若存在）的取值生成，如第 7.5 节所述。

**7.4.2 Code类型的映射**

**7.4.2.1 Code** 元素信息项须自一个 **Code** 类型的取值生成。

**7.4.2.2 value** 组件的取值应按第 7.4.3 节所述进行映射，以提供该 **Code** 元素信息项的第一个（若 **subcodes** 组件为空，则为惟一的一个）子元素信息项。

**7.4.2.3 subcodes** 组件的第一个 **XSD.QName**（如果有）须生成：

- a) 一个 **Subcode** 元素信息项，作为该 **Code**（代码）元素信息项的第二个子元素信息项；和
- b) 一个 **Value** 元素信息项（指一个子代码元素信息项的孩子，且自 **XSD.QName** 第一次出现时的取值生成，如第 7.4.2.5 节到第 7.4.2.6 节所述），作为 a) 中生成的 **Subcode** 元素信息项的第一个子元素信息项。

**7.4.2.4 subcodes** 组件接下来的每一个 **XSD.QName**（若存在）须生成：

- a) 一个 **Subcode** 元素信息项，作为生成自刚出现的 **XSD.QName** 取值的 **Subcode** 元素信息项的第二个孩子；和
- b) 一个 **Value** 元素信息项（指一个 **Subcode** 元素信息项的孩子，且自当前出现的 **XSD.QName** 的取值生成，如第 7.4.2.5 节至第 7.4.2.6 节所述），作为 a) 中生成的 **Subcode** 元素信息项的第一个子元素信息项。

注 — 当且仅当在 **subcodes** 中有后续的 **XSD.QName** 时，每个 **Subcode** 元素信息项都有第二个（**Subcode**）子元素信息项。

**7.4.2.5** 一个 **Value** 元素信息项（指一个 **Subcode** 元素信息项的孩子）须自 **XSD.QName**（且其 **uri** 组件存在）的一次出现生成，并带有：

- a) 在其[**in-scope namespaces**]特性之中的一个名空间信息项，其[**namespace name**]特性与 **uri** 组件，并且与一个由该 SOAP 节点选出的[**prefix**]特性的取值相等；和
- b) 一组字符信息项序列须为如下几项的级联：
  - 1) a) 中的[**prefix**]特性；
  - 2) 一个冒号（“.”）；和
  - 3) **name** 组件的字符串值。

**7.4.2.6 Value** 元素信息项（**Subcode** 元素信息项的孩子）须自 **XSD.QName**（且其 **uri** 组件不存在）的一次出现的取值生成，并且该 **XSD.QName** 带有一组子字符信息项的序列，它们是该 **name** 组件的取值。

**7.4.3 Value类型的映射**

**Value** 元素信息项（**Code** 元素信息项的孩子）须自一个带有一组字符信息项序列的 **Value** 类型的取值生成，其中这列字符信息项须自由如下各项级联所组成的字符串中的字符的枚举值生成：

- a) 如第 7.1.2 节所述的[**prefix**]特性；
- b) 一个冒号（“.”）；和
- c) 一个如表 2 所示的本地名。

表 2—value类型到本地名的映射

value 的枚举值	本地名
versionMismatch	VersionMismatch
mustUnderstand	MustUnderstand
dataEncodingUnknown	DataEncodingUnknown
sender	Sender
receiver	Receiver

**7.4.4 Text类型的映射**

**7.4.4.1 Text** 元素信息项须自一个 **Text** 类型的取值生成。

- 7.4.4.2 属性信息项须自 **lang** (语言) 组件生成, 并带有
- 一个取值为 “**lang**” 的[**local name**] ([本地名]) 特性; 和
  - 一个取值为 “**http://www.w3.org/XML/1998/namespace**” 的[**namespace name**]特性; 和
  - 一个取值为 “**xml**” 的[**prefix**]特性;
  - 一个等于 **lang** 组件的取值的[**normalized value**]特性。
- 7.4.4.3 **Text** 元素信息项的一组 **child** 字符信息项的序列须为 **Text** 组件的字符串值。

## 7.5 Content类型的映射

### 7.5.1 概述

7.5.1.1 内容元素信息项须自一个第 7.5.2、7.5.3 或 7.5.4 节中的 **Content** 类型的取值生成, 分别用于由快速信息集文档、ASN.1 编码值和 “不可理解” 的 ASN.1 SOAP 头部块 (见第 7.5.4 节) 到 XML 信息集的映射。

7.5.1.2 如果 **Content** 类型的 **fast-info-set-document** 选项存在, 则须应用第 7.5.2 节。

7.5.1.3 如果 **Content** 类型的 **encoded-value** 选项存在且 **encoded-value.id** 的取值不为 **notUnderstoodIdentifier**, 则须应用第 7.5.3 节。

7.5.1.4 如果 **Content** 类型的 **encoded-value** 选项存在且 **encoded-value.id** 的取值为 **notUnderstoodIdentifier**), 则须应用第 7.5.4 节。

### 7.5.2 快速信息集文档内容

7.5.2.1 **fast-info-set-document** 组件的八比特组将是一个如 ITU-T X.891 建议书 | ISO/IEC 24824-1 所述的快速信息集文档。

7.5.2.2 内容元素信息项须由如下的应用来生成:

- 对 **fast-info-set-content** 的八比特组进行解码以生成一个根元素信息项的 XML 信息集 (如 ITU-T X.891 建议书 | ISO/IEC 24824-1 所述); 和
- 将第 7.5.2.3 节应用于该根元素信息项以生成内容元素信息项。

7.5.2.3 如下的属性信息项 (在根元素信息项的[**attributes**]特性的成员中), 如果自从快速信息集的取值到一个根元素信息项的映射生成, 则须从该元素信息项的[**attributes**]特性中删除:

- role** 属性信息项;
- mustUnderstand** 属性信息项;
- relay** 属性信息项。

注 — **role**、**mustUnderstand** 和 **relay** 属性信息项分别映射自 **HeaderBlock** 类型的 **role**、**mustUnderstand** 和 **relay** 组件 (见第 7.2.2 节)。从根元素信息项的[**attributes**]特性中删除这些属性信息项保证了只有 **HeaderBlock** 的组件将被一个 SOAP 节点用于处理 W3C SOAP 头部块。

### 7.5.3 嵌入式ASN.1编码值内容

7.5.3.1 在内容元素信息项的[**attributes**]特性的成员中的一个 **encodingStyle** 属性信息项 (见《W3C SOAP Part 1》, 第 5.1.1 节), 在生成时须带有一个如下的[**normalized value**]特性:

“ **urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper**”。

7.5.3.2 如果 **encoded-value.id** 组件有 **qName** 选项存在, 则该内容元素信息项的[**namespace name**]和[**local name**]特性须从 **qName** 来进行设置。

7.5.3.3 如果 **encoded-value.id** 组件有 **roid** 相对对象标识符选项存在, 则该内容元素信息项在生成时须带有如下特性:

- 一个取值为 “**roid**” 的[**local name**]特性;
- 一个取值为 “**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**” 的[**namespace name**]特性;
- [**attributes**]特性的成员中的一个 **roid** 属性信息项, 如第 7.5.3.4 节所述。

7.5.3.4 一个属性信息项, 在该元素信息项的[**attributes**]特性的成员中, 须自一个 **Content** 类型的取值 (如果 **encoded-value.id** 组件有 **roid** 选项存在) 生成, 并带有:

- 一个取值为 “**roid**” 的[**local name**]特性;

- b) 一个取值为 “**urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope**” 的[**namespace name**]特性;
- c) 一个取值为 “true” 的[**specified**]特性;
- d) 一个 [**normalized value**] 特性，须为只使用 “XMLNumberForm” 编码为 “XMLRelativeOIDValue” 的 **roid** 组件的取值（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

**7.5.3.5** 内容元素信息项的一组子字符信息项的序列须自一个八比特组串的 Base64 编码（如 IETF RFC 2045 第 6.8 节所述）生成，且该八比特组串为 **encoded-value.encoding** 组件的取值。

**7.5.3.6** **schema-identifier** 组件，如果存在，须被忽略而不进行映射。

## 7.5.4 不可理解的W3C SOAP头部块内容

**7.5.4.1** **notUnderstoodIdentifier** 将标识 ASN.1 类型 **NotUnderstood**，其取值使用基本对齐分组编码规则编码为一个 **encoded-value.encoding** 组件取值的八比特组串。

**7.5.4.2** **NotUnderstood** 类型的取值须通过使用基本对齐分组编码规则解码 **encoded-value.encoding** 组件的八比特组串来生成。

**7.5.4.3** 一个 **NotUnderstood** 元素信息项（见《W3C SOAP Part 1》，第 5.4.8.1 节）须作为内容元素信息项而生成，并带有：

- a) 一个名空间信息项，在其[**in-scope namespaces**]特性成员之中，[**namespace name**]特性等于 **NotUnderstood.uri** 组件的取值，以及一个由 SOAP 节点选择的唯一的[**prefix**]特性；及
- b) 一个 **qname** 属性信息项（见《W3C SOAP Part 1》，第 5.4.8.2 节），带有一个 [**normalized value**]特性，它由一个 a) 中的 [**prefix**]特性、一个冒号（“:”）和 **NotUnderstood.name** 组件的字符串值的级联组成。

## 8 将W3C SOAP消息信息集映射到Envelope类型的抽象值

### 8.1 概述

**8.1.1** **Envelope** 类型的取值须自一个 **Envelope** 元素信息项生成。

**8.1.2** **Header** 元素信息项（如果存在）须映射到 **header** 组件，如第 8.2 节所述。

**8.1.3** 如果一个 **Body** 元素信息项不包含一个 **Fault** 元素信息项作为其唯一的子元素信息项，则须生成一个带有 **body** 选项存在的 **body-or-fault** 组件的取值，并且该 **Body** 元素信息项须映射为 **body** 选项，如第 8.3 节所述。

**8.1.4** 如果一个 **Body** 元素信息项包含一个 **Fault** 元素信息项作为其唯一的子元素信息项，则须生成一个带有 **fault** 选项存在的 **body-or-fault** 组件的取值，且该 **Fault** 元素信息项须映射为 **fault** 选项，如第 8.4 节所述。

### 8.2 Header元素信息项的映射

**8.2.1** 一个 **Header** 类型的取值须自一个 **Header** 元素信息项生成，且每个子元素信息项（一个 W3C SOAP 头部块）须依次映射为 **sequence-of** 中 **Content** 的一次出现，如第 8.2.2 节所述。

**8.2.2** 一个 **HeaderBlock** 类型的取值须自一个 W3C SOAP 头部块生成，且该 W3C SOAP 头部须映射为一个内容组件的取值，如第 8.5 节所述。**HeaderBlock** 类型的附加组件须按第 8.2.2.1 节至第 8.2.2.3 节所述生成。

**8.2.2.1** 一个 **mustUnderstand** 组件的取值须自 **mustUnderstand** 属性信息项（如果存在）生成，如果该 **mustUnderstand** 属性信息项的[**normalized value**]特性为 “1”，则该组件的取值须为 **TRUE**。否则该组件不得存在。

**8.2.2.2** 一个 **relay** 组件的取值须自 **relay** 属性信息项（如果存在）生成，如果该 **relay** 属性信息项的[**normalized value**]特性为 “1”，则该组件取值须为 **TRUE**。否则该组件不得存在。

**8.2.2.3** 一个 **role** 组件的取值须自 **role** 属性信息项（如果存在）生成，且为该 **role** 属性信息项的[**normalized value**]特性。

### 8.3 Body元素信息项的映射

8.3.1 一个 **Body** 类型的取值须自一个 **Body** 元素信息项生成。

8.3.2 **Body** 元素信息项的子元素信息项（如果存在）须映射为一个 **content** 组件的取值，如第 8.5 节所述。

### 8.4 Fault元素信息项的映射

#### 8.4.1 概述

8.4.1.1 一个 **Fault** 类型的取值须自一个 **Fault** 元素信息项生成。

8.4.1.2 一个 **Code** 元素信息项须映射为一个 **code** 组件的取值，如第 8.4.2 节所述。

8.4.1.3 一个 **reason** 组件的取值须自 **Reason** 元素信息项生成。每个子 **Text** 元素信息项须依次映射为 **sequence-of** 中 **Text** 的一次出现，如第 8.4.4 节所述。

8.4.1.4 一个 **node** 组件的取值须自 **Node** 元素信息项（如果存在）生成，且将该 **Node** 元素信息项的子字符信息项的序列作为其字符串的取值。

8.4.1.5 一个 **role** 组件的取值须自 **Role** 元素信息项（如果存在）生成，且将该 **Role** 元素信息项的子字符信息项的序列作为其字符串的取值。

8.4.1.6 一个 **detail** 组件的取值须自 **Detail** 元素信息项（如果存在）生成，且子元素信息项须按第 8.5 节所述的进行映射。

#### 8.4.2 Code元素信息项的映射

8.4.2.1 一个 **Code** 类型的取值须自一个 **Code** 元素信息项生成。

8.4.2.2 **Value** 元素信息项（一个 **Code** 元素信息项的孩子）须映射为一个 **value** 组件的取值，如第 8.4.3 节所述。

8.4.2.3 第一个 **Subcode** 元素信息项（如果存在）须生成一个 **XSD.QName** 类型的取值作为 **subcodes** 组件的第一项。该取值须自第一个子 **Value** 元素信息项生成，如第 8.4.2.5 节和第 8.4.2.6 节所述。

8.4.2.4 每个 **Subcode** 元素信息项的第二个子 **Subcode** 元素信息项（如果存在）须生成一个 **XSD.QName** 类型的取值作为 **subcodes** 组件的下一项。该取值须自第二个子 **Subcode**（子代码）元素信息项的第一个子 **Value**（值）元素信息项生成，如第 8.4.2.5 节或第 8.4.2.6 节所述。

8.4.2.5 一个 **XSD.QName** 类型的取值须自一个 **Value**（值）元素信息项（一个 **Subcode** 元素信息项的孩子）生成，具有一组子字符信息项的序列，由一个前缀（如 P）、一个冒号（“:”）和一个本地名的级联所组成，并带有：

- a) 一个取值为名空间信息项的[**namespace name**]特性的 **uri** 组件的值，在 **Value** 元素信息项（一个 **Subcode** 元素信息项的孩子）的[**in-scope namespaces**]特性成员中并带有 [**prefix**]特性 P；
- b) 一个取值为本地名的 **name** 组件的值。

8.4.2.6 一个 **XSD.QName** 类型的取值须自一个 **Value** 元素信息项（一个 **Subcode** 元素信息项的孩子）生成，有一组不包括冒号（“:”）的子字符信息项的序列，且 **name** 组件的取值为子字符信息项的序列的字符串值。

#### 8.4.3 Value元素信息项（一个Code元素信息项的孩子）的映射

一个 **value** 类型的取值须自一个 **Value** 元素信息项（一个 **Code** 元素信息项的孩子）生成并带有一个本地名，如表 2 所述，且须为由如下各项级联所组成的子字符信息项的序列的子串：

- a) 一个[**prefix**]特性，如第 7.1.2 节所述；
- b) 一个冒号（“:”）；和
- c) 一个本地名。

#### 8.4.4 Text元素信息项的映射

8.4.4.1 一个 **Text** 类型的取值须自一个 **Text** 元素信息项生成。

8.4.4.2 一个 **lang** 组件的取值须自一个[**local name**]（[本地名]）特性为“**lang**”且[**namespace name**]特性为“**http://www.w3.org/XML/1998/namespace**”的属性信息项生成，并且取值为该属性信息项的[**normalized value**]特性。

8.4.4.3 一个 **Text** 组件的取值须自 **Text** 元素信息项生成，且为 **Text** 元素信息项的子字符信息项的序列。

## 8.5 内容元素信息项到Content类型取值的映射

### 8.5.1 概述

8.5.1.1 一个 **Content** 类型的取值须自一个内容元素信息项生成，并按第 8.5.2 节、第 8.5.3 节或第 8.5.4 节所述，进行从 XML 信息集分别到快速信息集文档、ASN.1 编码值和不可理解的 ASN.1 SOAP 头部块的映射。

8.5.1.2 如果如下情况发生，则须应用第 8.5.2 节：

- a) **encodingStyle** 属性信息项（见《W3C SOAP Part 1》，第 5.1.1 节）不在该内容元素信息项的[**attributes**]特性的成员之中，并且该内容元素信息项不是一个 **NotUnderstood** 元素信息项（见《W3C SOAP Part 1》，第 5.4.8.1 节）；或
- b) **encodingStyle** 属性信息项在该内容元素信息项的[**attributes**]特性的成员之中，并且该 **encodingStyle** 属性信息项有一个不等于“**urn:ohm:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper**”的[**normalized value**]特性（见第 7.5.3.1 节）。

8.5.1.3 如果 **encodingStyle** 属性信息项（见《W3C SOAP Part 1》，第 5.1.1 节）是在该内容元素信息项的[**attributes**]特性的成员之中，并且 **encodingStyle** 属性信息项有一个取值为“**urn:ohm:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper**”的[**normalized value**]特性，则应用第 8.5.3 节。

8.5.1.4 如果该内容元素信息项是一个 **NotUnderstood** 元素信息项（见《W3C SOAP Part 1》，第 5.4.8.1 节），则须应用第 8.5.4 节。

注 — 一个 **encodingStyle** 属性信息项不能在一个 **NotUnderstood** 元素信息项的[**attributes**]特性的成员之中（见《W3C SOAP Part 1》，第 5.4.8.1 节）。

### 8.5.2 嵌入式快速信息集文档

8.5.2.1 须生成一个带有 **fast-infoset-document** 选项存在的 **Content** 类型的取值。

8.5.2.2 **fast-infoset-document** 组件的八比特组将是一个快速信息集文档，且由如下应用生成：

- a) 将第 8.5.2.3 节应用于该内容元素信息项来生成一个 XML 信息集的根元素信息项；
- b) 将该 XML 信息集编码为一个快速信息集文档（如 ITU-T X.891 建议书 | ISO/IEC 24824-1 所述）。

8.5.2.3 如下的属性信息项，如果存在于该内容元素信息项的[**attributes**]特性的成员之中，则须从该元素信息项的[**children**]特性中删除。

- a) **role** 属性信息项；
- b) **mustUnderstand** 属性信息项；
- c) **relay** 属性信息项。

注 — 从该内容元素信息项的[**attributes**]特性中删除这些属性信息项保证了只有 **HeaderBlock** 的组件将被 SOAP 节点用于处理 W3C SOAP 头部块。

### 8.5.3 嵌入式的ASN.1编码值

8.5.3.1 须生成一个带有 **encoded-value** 选项存在的 **Content** 类型的取值。

8.5.3.2 如果 **roid** 属性信息项（见第 7.5.3.4 节）不在该内容元素信息项的[**attributes**]特性的成员之中，则：

- a) 须生成一个带有 **qName** 选项存在的 **encoded-value.id**；且
- b) 其取值须从该内容元素信息项的[**local name**]特性和[**namespace name**]特性进行设置。

8.5.3.3 如果 **roid** 属性信息项（见第 7.5.3.4 节）在该内容元素信息项的[**attributes**]特性的成员中，则：

- a) 须生成一个带有 **roid** 选项存在的 **encoded-value.id**；且
- b) 其取值须从一个只使用“XMLNumberForm”编码为一个“XMLRelativeOIDValue”的 **roid** 属性信息项的[**normalized value**]特性进行设置（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

注 — 当对 ASN.1 SOAP 消息的大小有限制时，可以使用相对对象标识符，而不是使用带有限制符的名字。

8.5.3.4 一个 **encoded-value.encoding** 组件的取值须自内容元素信息项的子字符信息项的序列生成，且须为八比特组串，其中的内容元素信息项为一个八比特组串的 Base64 编码。

8.5.3.5 **schema-identifier** 组件不得映射，应被忽略。

## 8.5.4 不可理解的W3C SOAP头部块的内容

8.5.4.1 须生成一个带有 **qName** 选项存在的 **encoded-value.id**；且其取值须按照该 **notUnderstood** 元素信息项的[**local name**]（[本地名]）特性和[**namespace name**]特性进行设置。

8.5.4.2 一个 **NotUnderstood** 类型的取值须自 **NotUnderstood** 元素信息项生成，且带有 **qname** 属性信息项的[**normalized value**]特性，它由一个前缀（如：P）、一个冒号（“:”）和一个本地名的级联所组成，并带有：

- a) 一个取值为名空间信息项的[**namespace name**]特性的 **uri** 组件的值，在 **NotUnderstood** 元素信息项的[**in-scope namespaces**]特性的成员中，并带有[**prefix**]特性 P；和
- b) 一个取值为本地名的 **name** 组件的值。

8.5.4.3 **NotUnderstood** 类型的取值须使用基本对齐分组编码规则编码为一个取值为 **encoded-value.encoding** 组件的八比特组串。

## 9 嵌入式ASN.1编码值的扩展SOAP处理

### 9.1 概述

9.1.1 下面子节所描述的扩展处理拓展了对 W3C SOAP 消息的处理（如《W3C SOAP Part 1》所规定的），允许由一个 SOAP 节点对已经从 ASN.1 SOAP 消息映射来的内容元素信息项进行附加的转换。

注一 需要扩展处理是由于内容元素信息项将包括一个嵌入式编码值，作为子字符信息项的序列，而这个嵌入式编码值对于 SOAP 节点而言是不透明的，除非进行更进一步的处理来从子字符信息项的序列生成一个 ASN.1 值。

9.1.2 内容元素信息项须为 **Body** 元素信息项、**Header** 元素信息项（W3C SOAP 头部块）、**Detail** 元素信息项的一个子元素信息项。

注一 内容元素信息项一般按如下步骤进行处理：

- a) 一个最终的 SOAP 接收者处理一个 **Body** 元素信息项的孩子或详细元素信息项的孩子和任何目标 W3C SOAP 头部块；
- b) 一个 SOAP 中介处理任何目标 W3C SOAP 头部块；
- c) 一个 SOAP 节点把目标 W3C SOAP 头部块作为 a) 或 b) 的结果来处理；以及
- d) 一个 SOAP 节点（比如一个活跃的中介）通过未在 W3C SOAP 头部块中描述的附加处理方式来处理信息项。

9.1.3 内容（孩子）元素信息项将会在[**attributes**]特性的成员中有一个带有[**normalized value**]特性为“**urn:ohm:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper**”的 **encodingStyle** 属性信息项，如第 7.5.3.1 节所述。

9.1.4 对于一个内容元素信息项的扩展处理的应用须为如下概念步骤的结果：

- a) 嵌入式 ASN.1 编码值的 ASN.1 类型按第 9.2 节所规定的进行识别；
- b) 从所识别的嵌入式 ASN.1 编码值生成 ASN.1 值，并按第 9.3 节所规定的给定 ASN.1 类型；
- c) 生成的 ASN.1 值由 SOAP 节点进行处理，通常产生一个或多个带有标识符的结果 ASN.1 值；以及
- d) 带有标识符的结果 ASN.1 值将作为如第 9.4 节所述的嵌入式 ASN.1 编码值被插入到新的 W3C SOAP 消息信息集中（见第 6.4 节 b）。

注一 这 4 个步骤只是概念上的。并不要求一个实现从一个给定的嵌入式 ASN.1 值中实际生成一个 ASN.1 值，这是因为并不要求一个实现生成一个 W3C SOAP 消息信息集 的表示。

### 9.2 嵌入式ASN.1编码值的ASN.1类型的识别

9.2.1 为了识别的目的，一个 **Identifier** 类型的取值须自第 9.2.2 节和第 9.2.3 节中的内容元素信息项生成，且该取值须标识一个嵌入式 ASN.1 编码值的 ASN.1 类型。

**9.2.1.1** 若 ASN.1 类型无法从 **Identifier** 的取值进行识别，则处理的 SOAP 节点须产生故障，如第 9.5 节所述。

**9.2.1.2** 处理的 SOAP 节点用来获得和管理 **Identifier** 值和标识的 ASN.1 类型的集合的方式不在本建议书 | 国际标准中描述。

注 — 一个 SOAP 节点可以从一个服务描述（见第 13.8 节）中获得一（部分）组标识的 ASN.1 类型的集合。

**9.2.2** 如果该内容元素信息项带有一个取值为“roid”的 **[local name]** 特性和一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope”的 **[namespace name]** 特性，且 **roid** 属性信息项（见第 7.5.3.4 节）在 **[attributes]** 特性的成员中，则：

- a) 须生成一个带有 **roid** 选项存在的 **Identifier** 类型的取值；且
- b) 其取值须按照只使用“XMLNumberForm”编码为一个“XMLRelativeOIDValue”的 **roid** 属性信息项的 **[normalized value]** 特性进行设置（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

**9.2.3** 如果 **roid** 属性信息项（见第 7.5.3.4 节）不在该内容元素信息项的 **[attributes]** 特性的成员中，则：

- a) 须生成一个带有 **qName** 选项存在的 **Identifier** 类型的取值；且
- b) 其取值须按照该内容元素信息项的 **[local name]** 特性和 **[namespace name]** 特性进行设置。

### 9.3 从标识的嵌入式ASN.1编码值生成ASN.1值

一个 ASN.1 值须自内容元素信息项的子字符信息项（嵌入式 ASN.1 编码值）生成。这些子字符信息项是八比特组串的 Base64 编码（如 IETF RFC 2045 第 6.8 节所述），包含按基本对齐分组编码规则编码的 ASN.1 值，且其 ASN.1 类型已被识别，如第 9.2 节所述。

## 9.4 将ASN.1值（带标识符）插入W3C SOAP消息

### 9.4.1 概述

**9.4.1.1** 一个带有标识符的 ASN.1 值，其标识符是一个 **Identifier** 类型的值及可能的附加值，须作为一个生成的内容元素信息项的嵌入式 ASN.1 编码值插入到一个 W3C SOAP 消息中，如下面的子节所述。

注 — **Identifier** 类型的取值和附加值可以从服务描述中获得或由一个 SOAP 节点应用来提供。

**9.4.1.2** 如果作为一个生成的内容元素信息项的嵌入式 ASN.1 编码值将要插入的 ASN.1 值是 **Header** 元素信息项的孩子（一个 W3C SOAP 头部块），则须应用第 9.4.2 节。

**9.4.1.3** 如果作为一个生成的内容元素信息项的嵌入式 ASN.1 编码值将要插入的 ASN.1 值是 **Body** 元素信息项的孩子，则须应用第 9.4.3 节。

**9.4.1.4** 如果作为一个生成的内容元素信息项的嵌入式 ASN.1 编码值将要插入的 ASN.1 值是 **Detail** 元素信息项的孩子，则须应用第 9.4.4 节。

### 9.4.2 作为Header元素信息项的一个子元素信息项插入

**9.4.2.1** 一个内容元素信息项须自 ASN.1 值和 **Identifier** 值生成，如第 9.4.5 节所述。该内容元素信息项须为一个 W3C SOAP 头部块，该头部块是该 **Header** 元素信息项的一个孩子。附加值（如果存在）的结果应为将属性信息项插入到该内容元素信息项的 **[attributes]** 特性的成员中，如下面的 3 个子节所述。

注 — SOAP 头部块元素信息项插入的顺序取决于处理的 SOAP 节点。

**9.4.2.2** 与一个 **role** 属性信息项的语义相对应的附加 URI（如果存在）须生成该 **role** 属性信息项，且其 **[normalized value]** 特性须为该 URI 的字符串值。

**9.4.2.3** 与一个 **mustUnderstand** 属性信息项的语义相对应的附加布尔值（如果存在），如果该布尔值为 **TRUE**，则须生成该 **mustUnderstand** 属性信息项，且其 **[normalized value]** 特性须为“1”。

**9.4.2.4** 与一个 **relay** 属性信息项的语义相对应的附加布尔值（如果存在），如果该布尔值为 **TRUE**，则须生成该 **relay** 属性信息项，且其 **[normalized value]** 特性须为“1”。

### 9.4.3 作为Body元素信息项的子元素信息项插入

一个内容元素信息项须自 ASN.1 值和 **Identifier** 值生成，如第 9.4.5 节所述。该内容元素信息项须为 **Body** 元素信息项的惟一孩子，并要替换该 **Body** 元素信息项的任何现存的子元素信息项。

#### 9.4.4 作为Detail元素信息项的子元素信息项插入

一个内容元素信息项须自 ASN.1 值和 **Identifier** 值生成，如第 9.4.5 节所述。该内容元素信息项须为 **Detail** 元素信息项的惟一孩子，并要替换该 **Detail** 元素信息项的任何现存的子元素信息项。

#### 9.4.5 从ASN.1值和Identifier值生成内容元素信息项

**9.4.5.1** 一个内容元素信息项须自 ASN.1 值（将作为一个嵌入式 ASN.1 编码值插入）和 **Identifier** 值生成，如下面的 4 个子节所述。

**9.4.5.2** 如果 **Identifier** 类型的值带有 **qName** 选项存在，则该内容元素信息项的[**namespace name**]和 [**local name**]特性须按照 **qName** 来进行设置。

**9.4.5.3** 如果 **Identifier** 类型的值带有 **roid** 选项存在，则该内容元素信息项的[**local name**]特性须为“roid”且[**namespace name**]特性须为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope”。在该内容元素信息项的[**attributes**]特性的成员中的属性信息项须在生成时带有：

- a) 一个取值为“roid”的[**local name**]（[本地名]）特性；
- b) 一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope”的[**namespace name**]特性；
- c) 一个取值为“true”的[**specified**]特性；
- d) 一个[**normalized value**]特性，其取值为只使用“XMLNumberForm”编码为一个“XMLRelativeOIDValue”的 **roid** 组件的值（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

**9.4.5.4** 一个内容元素信息项的子字符信息项（嵌入式 ASN.1 编码值）的序列须生成自八比特组串的 Base64 编码（如 IETF RFC 2045，第 6.8 节所述），以组成由基本对齐分组编码规则编码的 ASN.1 值。

**9.4.5.5** 在元素信息项的[**attributes**]特性成员中的一个 **encodingStyle** 属性信息项（见《W3C SOAP Part 1》第 5.1.1 节），须在生成时带有一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper”的[**normalized value**]特性。

#### 9.5 “ASN.1类型不可识别”故障

**9.5.1** 如果一个 SOAP 节点无法从按照第 9.2 节生成出的 **Identifier** 类型的取值中识别出嵌入式 ASN.1 编码值的 ASN.1 类型，则该 SOAP 节点须产生错误。下面的子节描述了当一个 W3C SOAP 消息信息集是一条故障时的故障信息的信息集。

**9.5.2** 一个 **Value** 元素信息项（一个 **Code** 元素信息项的孩子）须在生成时带有一个由如下字符串的级联所组成的子字符信息项的序列：

- a) 如第 7.1.2 节所述的[**prefix**]特性；和
- b) 一个冒号（“:”）；和
- c) 字符串“Sender”。

**9.5.3** 一个名空间信息项，带有一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope”的[**namespace name**]特性，且该特性在 **Envelope** 元素信息项（或一个处在直到（含）按第 9.5.4 节生成的 **Value** 元素信息项的任意层深度的子元素信息项）的[**in-scope namespaces**]特性的成员中，则该名空间信息项的惟一[**prefix**]特性在生成时其值由该 SOAP 节点进行选择。

**9.5.4** 一个 **Subcode** 元素信息项（一个 **Code** 元素信息项的孩子）须在生成时带有一个单子的子 **Value** 元素信息项。

**9.5.4.1** 该单子的子 **Value** 元素信息项须带有一组由如下字符串级联的子字符信息项的序列：

- a) 如第 9.5.3 节所述的[**prefix**]特性；
- b) 一个冒号（“:”）；和
- c) 字符串“NotIdentified”。

注 — 这样一种不可识别故障（一个 W3C SOAP 消息）可作为一个 XML 文档做如下串行化表示：

```
<env:Envelope xmlns:env=" http://www.w3.org/2003/05/soap-envelope ">
  <env:Body>
    <env:Fault>
      <env:Code>
        <env:Value>env:Sender</env:Value>
      </env:Code>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

```

                <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的HTTP绑定

ASN.1 SOAP 的 HTTP 绑定是对 SOAP HTTP 绑定的修改和扩展（见《W3C SOAP Part 2》，第 7 节），并且为传送采用基本对齐分组编码规则编码的 ASN.1 SOAP 消息提供了从 W3C SOAP 到 HTTP 的绑定。ASN.1 SOAP 的 HTTP 绑定符合 SOAP 协议绑定框架（见《W3C SOAP Part 1》，第 4 节）。附件 D 提供了使用 ASN.1 SOAP 的 HTTP 绑定的特性来进行快速 Web 服务和 XML Web 服务之间互操作的介绍材料。

### 10.1 HTTP媒体类型

**10.1.1** 为了实现本建议书 | 国际标准的目的，《W3C SOAP Part 2》第 7.1.4 节须进行如下 3 个子节所述的修改。

**10.1.2** 一个 ASN.1 SOAP 的 HTTP 绑定的实现必须能使用“**application/fastsoap**”媒体类型发送和接收 ASN.1 SOAP 消息，对其的正确使用及参数说明见第 B.1 节。

**10.1.3** 一个实现也可以使用其他媒体类型发送请求和响应，其中这样的媒体类型可用于标识 W3C SOAP 消息。

注 — 除了这些因素之外，这些 W3C SOAP 消息可以是 XML SOAP 消息或快速信息集 SOAP 消息。

**10.1.4** 当发送请求时，一个绑定提供一个 HTTP 头部域 **Accept**（见 IETF RFC 2616，第 14.1 节）。该头部：

- a) 须指明至少有接受“**application/fastsoap**”媒体类型的能力；以及
- b) 可以补充说明愿意接受其他媒体类型来传送 W3C SOAP 消息。

注 — 一个实现可发送一个 XML SOAP 消息请求，并带有一个 HTTP 头部域 **Accept**，其取值为“**Accept: application/fastsoap, application/soap+xml**”，以指明接受 ASN.1 SOAP 消息的能力以及表明愿意接受 XML SOAP 消息（见第 D.2 节中所述的悲观策略）。

### 10.2 响应SOAP节点的行为

**10.2.1** 《W3C SOAP Part 2》第 7.5.2 节，须扩展如下两个子节。

**10.2.2** 一个响应 SOAP 节点，接收到一个 HTTP 头部域 **Accept** 并带有与“**application/fastsoap**”媒体类型平等优先权的一个或多个附加媒体类型，则须将“**application/fastsoap**”媒体类型解释为具有最高优先权（见 IETF RFC 2616，第 14.1 节），且须以该媒体类型来做出响应。

**10.2.3** 如果一个响应节点不能从 HTTP 请求中确定发送者能够处理由“**application/fastsoap**”媒体类型标识的内容，则该响应 SOAP 节点须增加带有一个空域值的 HTTP 头部域 **Fast-Enabled**，以表示它支持快速 Web 服务。

## 11 快速信息集SOAP消息和SOAP HTTP绑定

本节说明了一个支持在 HTTP 上传送快速信息集 SOAP 消息的 SOAP HTTP 绑定（见《W3C SOAP Part 2》，第 7 节）（被称为快速信息集 SOAP HTTP 绑定）。

**11.1** 一个快速信息集 SOAP HTTP 绑定的实现须能够使用“**application/soap+fastinfoset**”媒体类型（其正确的使用和参数如第 B.2 节所述）发送和接收快速信息集 SOAP 消息。

注 — SOAP HTTP 绑定支持使用其他媒体类型发送请求和响应，只要这些媒体类型表示传送 W3C SOAP 消息信息集。

**11.2** 一个绑定在发送请求时可提供一个 HTTP 头部域 **Accept**（见 IETF RFC 2616，第 14.1 节）。该头部：

- a) 须指明至少有接受“**application/soap+fastinfoset**”媒体类型的能力。

b) 可以补充表示愿意接受其他媒体类型以传送 W3C SOAP 消息。

注 — 一个实现可以发送一个 XML SOAP 消息请求，并带有一个取值为 “**Accept: application/soap+fastinfoset, application/soap+xml**” 的 HTTP 头部域 **Accept**，以指明接受快速信息集 SOAP 消息的能力以及表示愿意接受 XML SOAP 消息。

## 12 支持ASN.1 SOAP接口绑定的面向SOAP的服务描述

### 12.1 概述

**12.1.1** 本节描述了支持 ASN.1 SOAP 接口绑定（见第 12.4.7 节）的面向 SOAP 的服务描述。

**12.1.2** 面向 SOAP 的服务描述是用于描述接口和一个 Web 服务语义的一组文档，其中该 Web 服务将通过 SOAP 消息交互来提供。

**12.1.3** 面向 SOAP 的服务描述须满足本节所述的需求，但不同的是，对于那些文档的形式或写这些文档所使用的语言（自然的或正式的）并没有限制。

注 — 附件 E 描述了使用 WSDL 1.1[2]作为编写面向 SOAP 的服务描述的一种语言。

**12.1.4** 面向 SOAP 的服务描述须指明：

- a) 一组模式（见第 12.2 节）；
- b) 一组抽象接口，其中每一个都包含一组抽象操作（见第 12.3 节）；和
- c) 一组接口绑定（ASN.1 SOAP 接口绑定的集合），其中每一个都包含一组操作绑定（见第 12.4 节）。

### 12.2 模式 (Schemas)

**12.2.1** 对一个给定的 Web 服务的面向 SOAP 的服务描述可包含一个或多个内容数据类型的定义，这些数据将在提供该 Web 服务的 SOAP 消息中被承载。这包含了将在消息体、头部块和故障中所要承载的内容数据。

**12.2.2** 内容数据（如果有）须由一个或多个 XSD 模式来定义。每个模式或者通过提供其名空间 URI 引入，或者嵌入在服务描述中。

注 — 这里的“XSD 模式”不是一个“模式文档”，而是一个其 XML 表示中包括一个或多个“xsd:schema”元素信息项的抽象模式（一组模式组件 — 见《W3C XML schema》）。通常，一个模式通过包含其 XML 表示而嵌入在服务描述中。

**12.2.3** 引入或嵌入在服务描述中的所有 XSD 模式集被称为原始模式集。

**12.2.4** 原始模式集中内容数据的每个类型可通过一个顶级 **element declaration** 模式组件或一个顶级 **complex type definition** 或 **simple type definition** 模式组件来描述。

### 12.3 抽象接口和抽象操作

**12.3.1** 抽象接口通过提供一组抽象操作的信息来进行描述，并隐式地包含了如下的模式信息（派生自服务描述中的其他信息）：

- a) RPC 模式（见第 12.3.2 节）；和
- b) ASN.1 模式集（见第 12.3.3 节）。

**12.3.2** RPC 模式是支持 RPC 方式具体接口的一个特别构建的 XSD 模式，并且须按第 12.5 节所述生成。原始模式集以及增加的 RPC 模式合称为抽象接口的完整模式集。

**12.3.3** ASN.1 模式集是抽象接口的完整模式集的 ASN.1 映射。每个完整模式集中的 XSD 模式须独立映射到 ASN.1，如 ITU-T X.694 建议书 | ISO/IEC 8825-5 中所述。由 X.694 映射所生成的 ASN.1 模块形成了抽象接口的 ASN.1 模式集。

**12.3.4** 抽象接口通过提供如下信息来描述：

- a) 操作名（一个带有限制符的名字）；
- b) （可选）一个输入消息的定义；

- c) (可选) 一个输出消息的定义; 和
- d) 一组零或多个故障消息的定义。

**12.3.5** 如果同时存在输入和输出消息的定义, 则它们出现的顺序(例如, 一个请求和响应)不在本建议书|国际标准中描述, 但须在服务描述中指明。

**12.3.6** 输入消息定义和输出消息定义须有如下形式之一:

- a) 属于原始模式集(见第 12.2.4 节)中模式之一的零或一个顶级 **element declaration**; 或
- b) 由零或多个不同的不带限制符的名字组成的列表, 每个都与属于原始模式集(见第 12.2.4 节)中模式之一的一个顶级 **complex type definition** 或 **simple type definition** 相关联。

注一 在一些服务描述语言(例如, WSDL 1.1[2])中, 输入或输出消息定义的形式将通过操作绑定来描述, 作为对由抽象操作所提供信息的约束(本建议书|国际标准中未描述)。

**12.3.7** 每个故障消息定义须描述属于原始模式集(见第 12.2.4 节)中模式之一的一个顶级 **element declaration**。

**12.3.8** 如果对于每个操作, 输入消息定义(如果存在)有第 12.3.6 节中的形式 a) 且输出消息定义(如果存在)有形式 a), 则该接口被称为基于文档的。

**12.3.9** 如果对于每个操作, 输入消息定义存在并有第 12.3.6 节中的形式 b) 且输出消息定义(如果存在)有形式 b), 则该接口被称为基于 RPC 的。

**12.3.10** 否则, 该接口既不是基于文档的也不是基于 RPC 的, 且包含这些抽象接口的服务描述也不是如本建议书|国际标准中所描述的面向 SOAP 的服务描述。

## 12.4 接口绑定和操作绑定

**12.4.1** 一个接口绑定将一个抽象接口与附加信息关联起来, 其结果是形成了一个具体接口的完整规范。

**12.4.2** 接口绑定通过提供如下的信息来描述:

- a) (可选) 分配给该具体操作的一个对象标识符;
- b) 一组操作绑定;
- c) 一个传输的 URI;
- d) 该具体接口的方式(文档方式或 RPC 方式); 和
- e) 该具体接口是否支持快速 Web 服务。

**12.4.3** 分配给该具体操作(如果有)的对象标识符须根据 ITU-T X.660 建议书|ISO/IEC 9834-1 进行分配, 且必须惟一标识该具体操作。

注一 基于相同抽象操作的两个具体操作将会有不同的对象标识符。

**12.4.4** 一个操作绑定将一个抽象操作与附加信息关联起来, 其结果是形成了一个将通过 W3C SOAP 消息交互来执行的具体操作的完整规范。

**12.4.5** 一个传输是一个用于从一个 SOAP 节点到另一个 SOAP 节点传送 SOAP 消息所使用的协议, 且需作为一个 URI 来指明。

注一 在 SOAP 的上下文中, 传输被称为绑定。公共的例子是 XML SOAP 的 HTTP 绑定(见《W3C SOAP Part 2》, 第 7 节)和 ASN.1 SOAP 的 HTTP 绑定(见第 10 节)。

**12.4.6** 如果抽象接口是基于文档的, 则具体接口的方式必须为文档方式。如果抽象接口是基于 RPC 的, 则具体接口的方式必须为 RPC 方式。

**12.4.7** 一个支持快速 Web 服务的接口绑定被称为是一个 ASN.1 SOAP 接口绑定, 且具体操作可通过 ASN.1 SOAP 消息交互来执行。

**12.4.8** 一个操作绑定通过提供如下信息来描述:

- a) (可选) 一个 SOAP 动作的 URI;
- b) 零或多个 SOAP 头部块定义, 其中每一个都包含一个顶级 **element declaration**;
- c) 分配给顶级 **element declaration** 的零或多个对象标识符; 和

d) 对每一个顶级 **element declaration**，都提供一个指示符以指明它是否会被表示为一棵子树或一个嵌入式 ASN.1 值。

**12.4.9** 一个 SOAP 动作 URI 是指，当该 ASN.1 SOAP 的 HTTP 绑定（见第 10 节）被指明为传输时，为一个 HTTP 请求而放置在（若存在）“**application/fastsoap**”多用途互联网邮件扩展类型（见第 B.1 节）的 **action**（动作）参数中的一个 URI。

**12.4.10** 在头部块定义中每个顶级 **element declaration** 须属于原始模式集中的一个模式（见第 12.2.4 节）。

**12.4.11** 如果为该具体操作分配了一个对象标识符，可以为在一个输入消息定义、输出消息定义、故障消息定义、头部块定义中所描述的，或为一个 RPC 方式的具体操作所生成的一个或多个顶级 **element declaration** 分配一个唯一的对象标识符。如果该具体操作没有对象标识符，则对那些 **element declaration** 分配对象标识符是不允许的。

**12.4.12** 分配给一个 **element declaration** 的每个对象标识符须根据 ITU-T X.660 建议书|ISO/IEC 9834-1 进行分配，且须惟一标识该 **element declaration**。每个这种对象标识符须与该具体操作的对象标识符相同，而在其右侧增加一个或多个附加的对象标识符组件。

注 — 这将允许使用相对对象标识符来标识 SOAP 消息中一个嵌入式 ASN.1 编码值的类型（见第 9.2.2 节）。每个这样的相对对象标识符将只包括附加对象标识符组件，而对象标识符前边的组件将不会被传送。

**12.4.13** 带有一个分配给它的对象标识符的任何 **element declaration** 须为对第 12.4.8 节 d) 的一个附加指示，以表示该 **element declaration** 将编码为一个嵌入式 ASN.1 编码值而不是一棵子树。

注 — 使用嵌入式 ASN.1 编码值不需要对象标识符。在一个顶级 **element declaration** 缺少对象标识符的情况下，该 **element declaration** 的带有限制符的名字将用于标识 SOAP 消息中嵌入式 ASN.1 编码值的类型（见第 9.2.3 节）。

## 12.5 RPC 模式

**12.5.1** 一个 RPC 模式是指支持 RPC 方式的具体接口的特别构建的一个 XSD 模式（见第 12.4.6 节）。RPC 模式是隐式生成的而不是引入的或嵌入在服务描述中的。

注 — RPC 模式是隐式存在于所有面向 SOAP 的服务描述中的，但在服务描述中没有 RPC 方式的接口，则它为空白。

**12.5.2** 对一个给定的抽象接口的 RPC 模式（即绑定到一个 RPC 方式的具体接口）须按如下进行构建。

**12.5.3** 对基于 RPC 的抽象接口中描述的每个抽象操作，一个 **element declaration** 模式组件带有如下特性：

- **name**: 该操作的本地名；
- **target namespace**: 该操作的名空间的名称；
- **type definition**: 一个 **complex type definition** 模式组件，如第 12.5.4 节所述；
- **scope**: **global**,

且其余特性或者 **absent** 或者设为 **false** 或空（合适时），须增加到该 RPC 模式中。

**12.5.4** **type definition** 特性中的 **complex type definition** 模式组件须有如下特性：

- **name**: 无；
- **target namespace**: **absent**；
- **base type definition**: **ur-type**；
- **derivation method**: **restriction**；
- **content type**: **element-only** 和一个 **particle** 模式组件，如第 12.5.5 节所述，

且其余特性或者 **absent** 或者设为 **false** 或空（合适时），须增加到该 RPC 模式中。

**12.5.5** **content type** 特性中的 **particle** 模式组件须有如下特性：

- **min occurs**: 1；
- **max occurs**: 1；
- **term**: 一个 **model group** 模式组件，如第 12.5.6 节所述，

且须增加到该 RPC 模式中。

**12.5.6** 在 **term** 特性中的 **model group** 模式组件须有如下特性：

- **compositor: sequence;**
- **particles:** 零到多个 **particle** 模式组件的一个列表，如第 12.5.7 节所述（见第 12.3.6 节 b），

且须增加到该 RPC 模式中。

**12.5.7** **particles** 特性里的粒子列表中的每个 **particle** 须有如下特性：

- **min occurs: 1;**
- **max occurs: 1;**
- **term:** 一个 **element declaration** 模式组件，如第 12.5.8 节所述，

且须增加到该 RPC 模式中。

**12.5.8** **term** 特性中的 **element declaration** 模式组件须有如下特性：

- **name:** 在抽象操作的输入消息定义中不带限制符的名字之一；
- **target namespace: absent;**
- **type definition:** 与抽象操作（见第 12.3.6 节 b）的输入消息定义中不带限制符的名字相关的 **complex type definition** 或 **simple type definition** 的模式组件；
- **scope: complex type definition** 模式组件，如第 12.5.4 节所述，

且须增加到该 RPC 模式中。

**12.5.9** 对于在基于 RPC 的抽象接口中描述的带有一个输出消息定义的一个抽象操作，一个 **element declaration** 模式组件带有如下特性：

- **name:** 该操作的本地名，并带有后缀“Response”；
- **target namespace:** 该操作的名空间的名称；
- **type definition:** 一个 **complex type definition** 模式组件，如第 12.5.10 节所述；
- **scope: global,**

且其余特性或者 **absent** 或者设为 **false** 或空（合适时），须增加到该 RPC 模式中。

**12.5.10** 在 **type definition** 特性中的 **complex type definition** 模式组件须有如下特性：

- **name:** 无；
- **target namespace: absent;**
- **base type definition: ur-type;**
- **derivation method: restriction;**
- **content type: element-only** 和一个 **particle** 模式组件，如第 12.5.11 节所述，

且其余特性或者 **absent** 或者设为 **false** 或空（合适时），须增加到该 RPC 模式中。

**12.5.11** **content type** 特性中的 **particle** 模式组件须有如下特性：

- **min occurs: 1;**
- **max occurs: 1;**
- **term:** 一个 **model group** 模式组件，如第 12.5.15 节所述，

且须增加到该 RPC 模式中。

**12.5.12** 在 **term** 特性中的 **model group** 模式组件须有如下特性：

- **compositor: sequence;**
- **particles:** 零到多个 **particle** 模式组件的一个列表，如第 12.5.13 节所述（见第 12.3.6 节 b），

且须增加到该 RPC 模式中。

**12.5.13** **particles** 特性里粒子列表中的每个 **particle** 须有如下特性：

- **min occurs: 1;**

- **max occurs:** 1;
- **term:** 一个 **element declaration** 模式组件, 如第 12.5.14 节所述,

且须增加到该 RPC 模式中。

**12.5.14 term** 特性中的 **element declaration** 模式组件须有如下特性:

- **name:** 在抽象操作的输出消息定义中不带限制符的名字之一;
- **target namespace: absent;**
- **type definition:** 与抽象操作 (见第 12.3.6 节 b) 的输出消息定义中不带限制符的名字相关联的 **complex type definition** 或 **simple type definition** 的模式组件;
- **scope: complex type definition** 模式组件, 如第 12.5.4 节所述,

且须增加到该 RPC 模式中。

**12.5.15** 在第 12.5.8 节和第 12.5.14 节里 **type definition** 特性中的 **complex type definition** 或 **simple type definition** 模式组件须为原始模式集中 XSD 模式之一的一个模式组件的复本。这个模式组件, 连同出现在其特性之一 (除非以前增加过) 中或出现在任意深度的一个特性中的特性的任何模式组件, 须被增加到该 RPC 模式中 (除非以前增加过)。

## 13 使用带有ASN.1 SOAP接口绑定的面向SOAP的服务描述

**13.1** 包含针对一个给定的快速 Web 服务的 ASN.1 SOAP 接口绑定的一个面向 SOAP 的服务描述, 影响在提供快速 Web 服务过程中所有映射到和映射自 W3C SOAP 消息 (由 ASN.1 SOAP 接口绑定所描述) 的 ASN.1 SOAP 消息的形式与内容。

**13.2** 每个 W3C SOAP 消息须为服务描述中所述的某个具体接口的一个具体操作 (见第 12.4 节) 的一个输入或输出消息。那些输入消息须从客户端 SOAP 节点流向服务 SOAP 节点, 而那些输出消息则按照相反的方向流动。对于 W3C SOAP 消息为一个 RPC 方式的具体接口的具体操作的输出消息的情况, 只有在具体操作带有一个输入消息定义时才被允许。

**13.3** 一个 W3C SOAP 消息 (一个给定的具体操作的输入或输出消息) 中的任何头部块 (**Header** 元素信息项的孩子) 或故障 (**Detail** 元素信息项的孩子), 须为一个嵌入式元素信息项, 并分别对于该操作的头部块和故障, 符合顶级 **element declarations** 其中之一 (分别见第 12.3.7 节和第 12.4.10 节)。

**13.4** 一个 W3C SOAP 消息 (一个给定的具体操作的输入或输出消息) 的消息体 (**Body** 元素信息项的孩子), 须为符合下列 **element declaration** 的嵌入式元素信息项:

- a) 如果该具体操作是一个 RPC 方式的具体接口的一个成员 (见第 12.4.6 节), 分别为那个操作的输入或输出消息所隐式生成 (在 RPC 模式中, 见第 12.5 节) 的顶级 **element declaration** (分别见第 12.5.3 节和第 12.5.9 节); 或
- b) 如果该具体操作是一个文档方式的具体接口的一个成员 (见第 12.4.6 节), 分别在该操作的输入或输出消息定义中描述的顶级 **element declaration** (见第 12.3.6 节 a)。

**13.5** 由一个 ASN.1 SOAP 接口绑定所描述的一个嵌入式元素信息项须在一个 W3C SOAP 消息 (映射到一个 ASN.1 SOAP 消息) 中做如下表示:

- a) 作为一个元素信息项, 它是一棵子树; 或  
注 1 — 这样的项将映射到一个 ASN.1 SOAP 消息 (嵌入式快速信息集文档) 的组件。
- b) 作为一个嵌入式 ASN.1 编码值, 它自该元素信息项生成。  
注 2 — 服务描述指明该嵌入式元素信息项是被表示为一棵子树还是一个嵌入式 ASN.1 编码值 (见第 12.4.8 节 d)。

**13.6** 一个嵌入式 ASN.1 编码值的生成需要如下信息:

- a) 一个 ASN.1 类型;
- b) 该 ASN.1 类型的标识; 和
- c) 所标识的 ASN.1 类型的一个 ASN.1 取值。

**13.7** ASN.1 类型须为抽象接口 (见第 12.3.3 节) 的 ASN.1 模式集中的一个成员, 映射自顶级 **element declaration**, 且该嵌入式元素信息项符合该顶级 **element declaration**。

**13.8** ASN.1 类型的标识须为 **Identifier** 类型的一个取值。

**13.8.1** 如果一个对象标识符分配给了该顶级 **element declaration**（见第 12.4.12 节），则须应用 **Identifier** 值的 **roid** 选项，且该 **roid** 值须根据相对对象标识符（是对于已分配的对象标识符附加的对象标识符组件）来进行设置（见第 12.4.12 节）。

**13.8.2** 否则，须应用 **Identifier** 值的 **qName** 选项，且 **qName** 值需根据该顶级 **element declaration** 的带有限制符的名字进行设置。

**13.9** 如果给定了 ASN.1 类型，则需要生成 **Identifier** 值、ASN.1 值和嵌入式 ASN.1 编码值，并插入到 W3C SOAP 消息中，如第 9.4 节中所述。

注 — 第 9.4 子节描述了将一个 ASN.1 值（带有一个标识符）插入到一个 W3C SOAP 消息。ASN.1 值将被表示为一个字符信息项的序列，它们是该 ASN.1 值在基本对齐分组编码规则编码后再基于 Base64 的编码。这种表示法将被映射到一个 ASN.1 SOAP 消息的组件，它是一个八比特组串，且其取值为相应 ASN.1 值的基本对齐分组编码规则编码。

## 附 件 A

## 用于ASN.1 SOAP的ASN.1 模块

(本附件是本建议书 | 国际标准的组成部分)

用于 ASN.1 SOAP 的 ASN.1 模块如下所示。该模式重用了一些在 XSD 模块中定义的类型，如 ITU-T X.694 建议书 | ISO/IEC 8825-5 所述，以及在 **FastInfoset**（快速信息集）模块中定义的 **Document** 类型，如 ITU-T X.891 建议书 | ISO/IEC 24824-1 所述。

```

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 ::=
  "http://www.w3.org/2003/05/soap-envelope/role/UltimateReceiver"
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 "http://www.w3.org/2003/05/soap-envelope",  
    name "NotUnderstood"}  
END
```

## 附 件 B

### 快速Web服务的多用途互联网邮件扩展媒体类型

(本附件是本建议书 | 国际标准的组成部分)

本附件为与快速 Web 服务一起使用定义了两个多用途互联网邮件扩展 (MIME) 媒体类型:

- a) “**application/fastsoap**” 媒体类型描述了使用基本对齐分组编码规则对 ASN.1 SOAP 消息, 特别是 ASN.1 **Envelope** 类型的取值, 进行的编码 (见 B.1);
- b) “**application/soap+fastinfoset**” 媒体类型描述了串行化为快速信息集文档的快速信息集 W3C SOAP 消息 (见 B.2)。

多用途互联网邮件扩展媒体类型通过使用 IETF 多用途互联网邮件扩展注册模板来描述, 且已经依据 IETF 程序进行了注册。

#### B.1 “application/fastsoap” 媒体类型

多用途互联网邮件扩展媒体类型名:

application

多用途互联网邮件扩展子类型名:

Fastsoap (快速 SOAP)

必选参数:

无。

可选参数:

“action”: 本参数须用于标识 ASN.1 SOAP 消息的目的, 如《W3C SOAP 1.2》中为“application/soap+xml”的多用途互联网邮件扩展媒体类型 (见《W3C SOAP 1.2 Part 2》, 附录 A) 的“action”参数所做的描述。“action”参数的取值须为一个绝对 URI 引用, 如 IETF RFC 2396 中所述。对 URI 的特征不能有限制或者它是可解析的。

编码考虑:

这种媒体类型用于标识 ITU-T X.892 建议书 | ISO/IEC 24824-2 中的 ASN1SOAP 模块所述的 ASN.1 信封类型的取值的内容, 并使用 ITU-T X.691 建议书 | ISO/IEC 8825-2 中所述的基本对齐分组编码规则来进行编码。

对本多用途互联网邮件扩展媒体类型的使用, 如果它用于那些不提供 8 比特二进制透明性的传输时, 将需要补充规范。(为了快速 Web 服务, ITU-T X.892 建议书 | ISO/IEC 24824-2, 本媒体类型始终使用基本对齐分组编码规则来进行编码并使用 HTTP 作为其传输机制, 且不需要其他的规范。)

安全考虑:

因为 ASN.1 SOAP 消息可承载在语义上独立于任何多用途互联网邮件扩展包装 (或使用多用途互联网邮件扩展包装的上下文) 的应用定义的数据, 我们不能期望只基于多用途互联网邮件扩展包装的语义就能理解 ASN.1 SOAP 消息的语义。因此, 只要使用“application/fastsoap”媒体类型, 就强烈建议使用 ASN.1 SOAP 消息的上下文的安全实现是完全可理解的。安全实现可能涉及特定的 ASN.1 SOAP 到一个下层协议的绑定以及在 ASN.1 SOAP 消息中承载的数据的应用定义的语义。

互操作性考虑:

目前没有已知的互操作性问题。

发布的规范:

ITU-T X.892 建议书 | ISO/IEC 24824-2

使用该媒体类型的应用:

目前没有已知的应用使用该媒体类型。

## ISO/IEC 24824-2: 2005(C)

附加信息:

文件扩展:

ASN.1 SOAP 消息并没有必要或被期望作为文件来存储。

为获取更多信息的联系人和电子邮件地址:

ITU-T ASN.1 报告人 (通过 [tsbmail@itu.int](mailto:tsbmail@itu.int) 联系)

ISO/IEC JTC1/SC6 ASN.1 报告人 (通过 [ittf@iso.org](mailto:ittf@iso.org) 联系)

预期的使用:

COMMON

作者/修改管理员:

依照 ITU-T A.23 建议书《与国际标准化组织 (ISO) 和国际电工技术委员会 (IEC) 就信息技术进行合作》附件 A 和《ISO/IEC JTC1 指示》附件 K 的 ITU-T | ISO/IEC 联合选举程序。

## B.2 “application/soap+fastinfoset” 媒体类型

多用途互联网邮件扩展媒体类型名:

application

多用途互联网邮件扩展子类型名:

soap+fastinfoset

必选参数:

无。

可选参数:

“action” : 本参数须用于标识 W3C SOAP 消息信息集的目的, 如《W3C SOAP 1.2》中为“application/soap+xml”的多用途互联网邮件扩展媒体类型 (见《W3C SOAP 1.2 Part 2》, 附录 A) 的“action”参数所做的描述。“action”参数的取值须为一个绝对 URI 引用, 如 IETF RFC 2396 中所述。对 URI 的特征不能有限制或者它是可解析的。

编码考虑:

这种媒体类型用于标识串行化为快速信息集文档的 W3C SOAP 消息信息集, 如 ITU-T X.892 建议书 | ISO/IEC 24824-2 中所述。

对本多用途互联网邮件扩展媒体类型的使用, 如果它用于那些不提供 8 比特二进制透明性的传输时, 将需要补充规范。(为了快速 Web 服务, ITU-T X.892 建议书 | ISO/IEC 24824-2, 本媒体类型始终使用 HTTP 作为其传输机制, 且不需要其他的规范。)

安全考虑:

因为 W3C SOAP 消息信息集可承载在语义上独立于任何多用途互联网邮件扩展包装 (或使用多用途互联网邮件扩展包装的上下文) 的应用定义的数据, 我们不能期望只基于多用途互联网邮件扩展包装的语义就能理解 W3C SOAP 消息信息集的语义。因此, 只要使用“application/soap+fastinfoset”媒体类型, 就强烈建议使用 W3C SOAP 消息信息集的上下文的安全实现是完全可理解的。安全实现可能涉及特定的 SOAP 到一个下层协议的绑定以及在 W3C SOAP 消息信息集中承载的数据的应用定义的语义。

互操作性考虑:

目前没有已知的互操作性问题。

发布的规范:

ITU-T X.892 建议书 | ISO/IEC 24824-2

使用本媒体类型的应用:

目前没有已知的应用使用该媒体类型。

附加信息:

**魔术号码:**

对于一个快速信息集文档的标识的细节，参考“application/fastinfoset”媒体类型的魔术号码一节。

对于串行化为快速信息集文档的 W3C SOAP 消息信息集的标识需要对快速信息集文档进行解析，且元素信息项的特性，相应于元素树的根，符合《W3C SOAP 1.2》第 5.1 节所述的 SOAP 信封元素信息项的特性。

**文件扩展:**

串行化为快速信息集文档的 W3C SOAP 消息信息集并没有必要或被期望作为文件来存储。

**为获取更多信息的联系人和电子邮件地址:**

ITU-T ASN.1 报告人 (通过 [tsbmail@itu.int](mailto:tsbmail@itu.int) 联系)

ISO/IEC JTC1/SC6 ASN.1 报告人 (通过 [ittf@iso.org](mailto:ittf@iso.org) 联系)

**预期的使用:**

COMMON

**作者/修改管理员:**

依照 ITU-T A.23 建议书《与国际标准化组织 (ISO) 和国际电工技术委员会 (IEC) 就信息技术进行合作》附件 A 和《ISO/IEC JTC1 指示》附件 K 的 ITU-T | ISO/IEC 联合选举程序。

## 附 件 C

### 快速Web服务简介

(本附件是本建议书 | 国际标准的组成部分)

本附件提供了快速 Web 服务的介绍资料。描述了使用快速 Web 服务的一些好处。突出了对 SOAP 消息在概念上的处理和优化的处理之间的区别，并提供了一个例子。这个例子基于一个简单交互，其中客户端发送了一个请求消息并接收了响应消息。讨论了服务描述的使用，并提供了一个服务描述（在《WSDL 1.1》中一见[2]）的举例，其中描述了该消息举例中所提供的服务。

#### C.1 快速Web服务的优点

快速 Web 服务规范基于对 SOAP 消息的 ASN.1 定义的使用和它们的内容以及基于对那些消息的二进制编码的使用。这提供了快速 Web 服务的主要优点（快速计算机处理和低消息带宽），但对 XML SOAP 的许多进一步的优化将在下面进行讨论。

##### C.1.1 ASN.1工具

ASN.1 工具可用于开发 ASN.1 SOAP 处理器，但是 XML SOAP 处理器主要是通过手工编写的，而用于 SOAP 的 W3C XML 模式只是被用做指南，因为 XML 绑定工具不大可能帮助开发优化的 XML SOAP 处理器。ASN.1 的方法允许对使用工具或手工开发 SOAP 处理器进行选择，而不会引入严重的性能问题，并且可能缩短投入市场的时间。

##### C.1.2 优化的特性

ASN.1 SOAP 为 SOAP 节点提供了许多优化的特性（不只是通过使用 ASN.1 和分组编码规则所提供的压缩及高效的处理 — 见 ITU-T X.691 建议书 | ISO/IEC 8825-2）：

- a) 一个 ASN.1 SOAP 消息的消息主体是与一个 ASN.1 SOAP 消息的故障的编码明确分离的。这使得故障易于辨识和管理。
- b) 用于 W3C SOAP 消息的递归故障子码（见《W3C SOAP Part 1》，第 5.4.6 节）被平滑为用于 ASN.1 SOAP 消息的故障子码的序列。这将使解码器在解码前就知道有多少故障子码。
- c) ASN.1 相对对象标识符可替代带有限制符的名字来使用。在编码时，服务描述消息可通过相对标识符来注释，一般来说，这种标识符要比带有限制符的名字压缩很多，其结果是减小了消息的大小。
- d) 所有与属性相关的 ASN.1 SOAP 头部块组件的缺省值都被指明。
- e) 为 W3C SOAP 指定的故障代码将使用枚举类型而不是带有限制符的名字。

##### C.1.3 压缩消息和高效处理

使用 ASN.1 压缩编码规则来进行编码的 ASN.1 SOAP 消息通常比使用字符编码的 XML 数据所提供的 Web 服务要求的处理能力更少（因此提供了更高的事务处理速率）且要求的网络带宽更少。这在很多方面都是有利的：

- a) 受限制的设备，如移动电话、智能卡或者甚至是无线频率识别（RFID）设备，它们的处理能力存储器 and 电池寿命都是有限的。  
注 1 — 对于电池技术没有等价的摩尔定律（电池的寿命不能每 18 个月就延长一倍）。
- b) 带宽限制系统 — 如无线网络。  
注 2 — 无线网络的无线电频率，比如移动电话 GSM（全球移动通信系统）网络，可能会固定 10 年。对于无线电频率没有等价的摩尔定律（带宽不能每 18 个月就增长一倍）。
- c) 高吞吐量的事务处理系统，如需要在每秒钟内处理许多来自多个客户端的 SOAP 消息的系统。

## C.1.4 SOAP中介的高效处理

SOAP 中介有潜力比初始的 SOAP 发送者和最终的 SOAP 接收者处理更多的 SOAP 消息。处理 ASN.1 SOAP 消息的 SOAP 中介可以轻松地识别用于处理（包括解码）的 ASN.1 SOAP 头部块而跳过（包括复制）其他的 SOAP 头部块（目的地为其他 SOAP 中介或 SOAP 最终接收者）和 SOAP 体。（这是因为 SOAP 头部块和 SOAP 体编码为以长度作为前缀的八比特组的序列。）

注 — ASN.1 SOAP 中介还能有效地管理 **fault**，因为一个 **fault** 将始终发生在消息末尾（在 SOAP 头部块的后面）且如果头部块存在，则保证会在字节分界处开始。因而一个中介并非必须去解码 **fault**，除非该中介执行的是没有在 SOAP 处理模型中指明的处理。

## C.2 ASN.1 SOAP消息的概念上的和优化的处理

### C.2.1 概述

**C.2.1.1** 从 ASN.1 SOAP 消息到 W3C SOAP 消息的概念上的映射及反向映射保证了 W3C SOAP 处理模型可以应用到 ASN.1 SOAP 消息上。下面的 6 个子节突出强调了初始的 SOAP 发送者、SOAP 中介和最终的 SOAP 接收者处理消息所需要的概念上的步骤以及 SOAP 中介所需要的优化步骤。

**C.2.1.2** 一个实现 ASN.1 SOAP 的 HTTP 绑定的初始的 SOAP 发送者（见《W3C SOAP Part 1》，第 1.5.3 节）将按如下步骤生成 ASN.1 SOAP 消息：

- a) 创建一个新的 W3C SOAP 消息并将新的嵌入式 ASN.1 抽象值插入到该 W3C SOAP 消息中；
- b) 将该 W3C SOAP 消息映射为一个 ASN.1 SOAP 消息；和
- c) 使用基本对齐分组编码规则，将该 ASN.1 SOAP 消息编码为一个 HTTP 请求的内容的八比特组序列。

**C.2.1.3** 如果该初始的 SOAP 发送者使用 SOAP 请求—应答消息交互模式（见《W3C SOAP Part 2》，第 6.2 节），则该 SOAP 发送者（见《W3C SOAP Part 1》，第 1.5.3 节）将等待响应且改变角色而成为一个最终的 SOAP 接收者（见《W3C SOAP Part 1》，第 1.5.3 节）。

**C.2.1.4** 一个实现了 ASN.1 SOAP 的 HTTP 绑定的 SOAP 中介（见《W3C SOAP Part 1》，第 1.5.3 节）将按如下的步骤处理 ASN.1 SOAP 消息：

- a) 使用基本对齐分组编码规则，对一个从 HTTP 请求或响应内容中获得的八比特组序列进行解码，以获取一个输入的 ASN.1 SOAP 消息；
- b) 对该输入的 ASN.1 SOAP 消息进行映射以获得一个输入的 W3C SOAP 消息；
- c) 识别和处理在该输入的 W3C SOAP 消息中的嵌入式 ASN.1 抽象值；
- d) 修改该输入的 W3C SOAP 消息，使其成为一个输出的 W3C SOAP 消息并将新的嵌入式 ASN.1 抽象值插入到该输出的 W3C SOAP 消息中；
- e) 将该输出的 W3C SOAP 消息映射为一个输出的 ASN.1 SOAP 消息；
- f) 使用基本对齐分组编码规则，将该输出的 ASN.1 SOAP 消息编码为一个 HTTP 响应或请求的内容的八比特组序列。

**C.2.1.5** 一个实现 ASN.1 SOAP 的 HTTP 绑定的最终的 SOAP 接收者将按如下的步骤处理 ASN.1 SOAP 消息：

- a) 使用基本对齐分组编码规则，对一个从 HTTP 请求或响应内容中获得的八比特组序列进行解码，以获取一个 ASN.1 SOAP 消息；
- b) 对该 ASN.1 SOAP 消息进行映射以获得一个输入的 W3C SOAP 消息；
- c) 识别和处理在该 W3C SOAP 消息中的嵌入式 ASN.1 抽象值；

**C.2.1.6** 如果该最终的 SOAP 接收者使用 SOAP 请求—应答消息交互模式，则该 SOAP 节点将改变角色而成为一个初始的 SOAP 发送者并会发送一个 ASN.1 SOAP 消息做为响应。

**C.2.1.7** 映射到和映射自 W3C SOAP 消息及处理嵌入式 ASN.1 值（在一个 W3C SOAP 消息中的识别和处理以及插入到一个 W3C SOAP 消息）的概念上的步骤在第 6 节到第 9 节中进行了描述。然而，一个 SOAP 节点可以选择通过跳过概念上的步骤来进行优化处理，只要其结果与执行这些概念步骤之后的结果相同即可（见第 6.4 节）。例如第 C.2.1.4 节中 b 到 e 的步骤为概念步骤，则 SOAP 中介可选择通过按如下步骤处理 ASN.1 SOAP 消息来进行优化：

- a) 使用基本对齐分组编码规则，对一个从 HTTP 请求内容中获得的八比特组序列进行解码，以获取一个输入的 ASN.1 SOAP 消息；
- b) 识别和处理在该输入的 ASN.1 SOAP 消息中的嵌入式 ASN.1 抽象值；
- c) 修改该输入的 ASN.1 SOAP 消息，使其成为一个输出的 ASN.1 SOAP 消息（或创建一个新的输出 ASN.1 SOAP 消息），并将新的嵌入式 ASN.1 抽象值插入到该输出的 ASN.1 SOAP 消息中；
- d) 使用基本对齐分组编码规则，将该输出的 ASN.1 SOAP 消息编码为一个 HTTP 响应的内容的八比特组序列。

## C.2.2 举例

如下子节所举的例子是从一个应用发送一个 ASN.1 SOAP 消息请求和接收响应的角度给出的。快速 Web 服务在第 C.3.2 节中使用 WSDL 1.1 进行了描述，并以《W3C SOAP Part 1》第 1.4 节中的样例 W3C SOAP 消息为基础。该服务是一个重要的服务，在该服务中一个应用可以请求关于那些对于该应用（或该应用的使用者）很重要的信息的最新警报。该请求应用将发送一个空的 ASN.1 SOAP 消息（不含应用定义的内容）并作为响应接收一个 ASN.1 SOAP 消息，其中为该警报带有两部分应用定义的内容（第 C.3.2 节中使用 WSDL 1.1 进行了描述），分别对应于：

- a) 一个为警报特性的 SOAP 头部块，即警报的优先级以及它的过期时间；和
- b) 一个为警报自身的 SOAP 体内容，那是一个对该警报的文字描述。

### C.2.2.1 W3C SOAP 消息请求

应用通过（使用一些适当的编程语言，如 Java）执行一个将会返回警报的不带参数的方法调用来请求最新的警报。初始的 SOAP 发送者将一个不带内容的 W3C SOAP 消息，其在 XML 中的表示为：

```
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Body>
  </env:Body>
</env:Envelope>
```

### C.2.2.2 ASN.1 SOAP 消息请求

这一 W3C SOAP 消息映射到一个包含如下信息的 ASN.1 SOAP 消息请求：

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

其中 **Envelope** 类型在附件 A 中定义（也参见第 6.1 节）。

### C.2.2.3 HTTP 请求

ASN.1 SOAP 消息然后使用基本对齐分组编码规则编码为一个 HTTP 请求的内容的八比特组序列。HTTP 头部域 **Content-Type** 为 “**application/fastsoap**” 且 **action** 参数设为 “**urn:alert**”。初始 SOAP 节点使用 HTTP 头部域 **Accept** 来声明 ASN.1 SOAP 消息和 XML SOAP 消息（这种情况为 SOAP 1.1 消息 [1]）都是支持的。

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

... sequence of octets ...
```

### C.2.2.4 HTTP 响应

初始的 SOAP 发送者然后将改变角色而成为一个最终的 SOAP 接受者并一直等待，直到它收到了对该请求的响应为止。在响应中的 HTTP 头部域 **Content-Type** 为 “**application/fastsoap**”。

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

... sequence of octets ...
```

### C.2.2.5 ASN.1 SOAP 消息响应

ASN.1 SOAP 消息使用基本对齐分组编码规则进行解码，以产生该 ASN.1 值：

```
envelope Envelope ::= {
  header { {
    role "http://example.org/alertrole",
    content : encoded-value {
      id : qName {
        uri "http://example.org/alertcontrol",
        name "alertcontrol"},
      encoding {.....}}}},
  body-or-fault : body {
    content : encoded-value {
      id : qName {
        uri "http://example.org/alert",
        name "alert"},
      encoding {.....}}}}
```

### C.2.2.6 W3C SOAP消息响应

C.2.2.6.1 ASN.1 SOAP 消息映射为一个 W3C SOAP 消息。该 W3C SOAP 消息包括一个 **alertcontrol** W3C SOAP 头部块和一个 **alert** 元素（信息项）作为一个 **Body** 元素信息项的孩子：

```
<env:Envelope
  xmlns:env="http://www.w3.org/2003/05/soap-envelope">
  <env:Header>
    <n:alertcontrol
      xmlns:n="http://example.org/alertcontrol"
      env:role="http://example.org/alertrole"
env:encodingStyle="urn:oh:n:joint-iso-itu-t:asn1:generic-applications:fast-web-services:soap-envelope:encoding-style:aper">
    ... Base64 content ...
    </n:alertcontrol>
  </env:Header>
  <env:Body>
    <m:alert
      xmlns:m=" http://example.org/alert"
env:encodingStyle="urn:oh:n: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 **alertcontrol** W3C SOAP 头部块的嵌入式 ASN.1 抽象值被识别和处理，开始于请求的 SOAP 节点作为 `http://example.org/alertrole` 的角色来运行。**alertcontrol** W3C SOAP 头部块的 **Identifier** 值和使用基本对齐分组编码规则解码自使用与该 **Identifier** 相关的 ASN.1 类型 **AlertControl** 的 Base64 编码内容的嵌入式 ASN.1 值，如下所示：

```
alertControlIdentifier Identifier ::= qName : {
  uri "http://example.org/alertcontrol",
  name "alertcontrol" }
alertcontrol Alertcontrol ::= {
  role "http://example.org/alertrole",
  priority 1,
  expires "2001-06-22T14:00:00-05:00" }
```

C.2.2.6.3 **alert** 元素（信息项）的嵌入式 ASN.1 抽象值被识别和处理，因为该 SOAP 节点是一个最终的 SOAP 接收者。**alert** 的 **Identifier** 值和使用基本对齐分组编码规则解码自使用与该 **Identifier** 相关的 ASN.1 类型 **Alert** 的 Base64 编码内容的嵌入式 ASN.1 值，如下所示：

```
alertIdentifier Identifier ::= qName : {
  uri "http://example.org/alert",
  name "alert" }
alert Alert ::= {
  msg "Pick up Mary at school at 2pm" }
```

## C.3 服务描述

### C.3.1 概述

**C.3.1.1** 在 WSDL 1.1[2]中所表达的服务描述可以不加修改地用来描述 ASN.1 SOAP 端点。这增加了快速 Web 服务的范围及使用，因为对 Web 服务开发者的影响已经最小化了。

**C.3.1.2** 用于 SOAP 1.1[1]的一个 WSDL 1.1 绑定接口（见附件 E）可被重用于一个 ASN.1 SOAP 绑定接口，WSDL 文档是一个面向 SOAP 的服务描述（见第 12 节和附件 E），并且 WSDL 1.1 绑定符合由《WS-I Basic Profile 1.0》[3]（见附件 E）所描述的解释和改进。

### C.3.2 举例

**C.3.2.1** 在第 C.3.3 节中所示的服务描述（在 WSDL 1.1 中表达）描述了一个 ASN.1 SOAP 接口绑定的第 C.2.2 节的举例。

**C.3.2.2** WSDL 文档有两个包含在 **wSDL:types**（描述一个 **Body** 元素信息项子内容，且只有 W3C SOAP 头部块的回复）中的 **xsd:schema** 定义。等价的 ASN.1 模式通过将 ITU-T X.694 建议书 | ISO/IEC 8825-5 应用到两个模式（见第 E.2 节和第 12.2 节）来获得。

**C.3.2.3** ASN.1 SOAP 的 HTTP 绑定将被使用，因为 **soapbind:binding** 元素（ASN.1 SOAP 接口绑定）的 **transport** 属性的值等于“**http://schemas.xmlsoap.org/soap/http/**”（见第 E.4.2 节和第 12.4.2 节）。

**C.3.2.4** 对快速 Web 服务的支持，通过使用在 **wSDL:binding** 元素里和在 **soapbind:binding** 元素后（见第 E.4.5 节和第 12.4.2 节 e）的 ASN.1 SOAP 接口绑定注释（一个 **fast-service:binding** 元素），为 ASN.1 SOAP 绑定接口所明确指明。

**C.3.2.5** ASN.1 SOAP 绑定的方式是文档方式的（见第 E.4.3 节和第 12.4.2 节 d），因为该接口绑定符合《WS-I Basic Profile 1.0》中所述的文档 — 文字绑定。

**C.3.2.6** 输入消息定义为空（无顶级 **element declaration**，因为 **AlertOperation** 操作绑定中的 **wSDL:input** 中的 **soapbind:body** 并没有隐式地引用 **wSDL:parts**（见第 E.4.9.1 节和第 12.3.6 节 a））。然而，一个 SOAP 动作 URI 存在，因为 **AlertOperation** 操作绑定有一个 **soapAction** 属性（见第 E.4.10 节和第 12.4.9 节）。URI “**urn:alert**” 将被放在“**application/fastsoap**”多用途互联网邮件扩展类型（见第 B.1 节）的 **action** 参数中，用于（包含空 ASN.1 SOAP 消息的）HTTP 请求的 HTTP 头部域 **Content-Type**。

**C.3.2.7** 输出消息有一个顶级 **element declaration**，**alert:alert**（因为 **AlertOperation** 操作绑定中的 **wSDL:output** 中的 **soapbind:body** 并没有隐式地引用 **wSDL:parts**（见第 E.4.9.1 节和第 12.3.6 节 a））。

**C.3.2.8** 为 **AlertOperation** 操作绑定的输出指明了 SOAP 头部块定义（**alertcontrol** W3C SOAP 头部块），并带有一个顶级 **element declaration alertcontrol:alertcontrol**（见第 E.4.11 节和第 12.4.8 节 c））。

### C.3.3 WSDL 1.1中表示的服务描述

```
<definitions name="Alert"
  xmlns="http://schemas.xmlsoap.org/wSDL/"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:soapbind="http://schemas.xmlsoap.org/wSDL/soap/"
  xmlns:http="http://schemas.xmlsoap.org/wSDL/http/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:fast-service="urn:oh:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description"
  xmlns:tns="http://example.org/alert/service"
  targetNamespace="http://example.org/alert/service"
  xmlns:alert="http://example.org/alert"
  xmlns:alertcontrol="http://example.org/alertcontrol">
  <types>
    <schema
      targetNamespace="http://example.org/alertcontrol"
      xmlns="http://www.w3.org/2001/XMLSchema"
      xmlns:soap="http://schemas.xmlsoap.org/wSDL/soap/"
      elementFormDefault="qualified">
      <import namespace="http://schemas.xmlsoap.org/wSDL/soap/" />
      <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>
  </types>
</definitions>
```

```

    </complexType>
  </element>
</schema>
<schema
  targetNamespace="http://example.org/alert"
  xmlns="http://www.w3.org/2001/XMLSchema"
  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="http://schemas.xmlsoap.org/soap/http"
    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="http://example.org/AlertPort"/>
  </port>
</service>
</definitions>

```

## 附 件 D

### 使用快速Web服务和XML Web服务的公共服务提供

(本附件不是本建议书 | 国际标准的组成部分)

本附件描述了可以被支持快速 Web 服务的客户端所应用以便与并不知道是否支持快速 (Web 服务) 的 SOAP 节点进行交互的策略。这些策略使用了在第 10 节中所描述的 ASN.1 HTTP 绑定的特性。

如果一个支持快速 Web 服务的客户端可识别出一个 SOAP 节点是支持快速 (Web 服务) 的, 则所应用的策略的结果是成功的, 且它们会通过交换 ASN.1 SOAP 消息进行交互; 否则其结果就是失败的, 它们将通过交换 XML SOAP 消息来进行交互。

注 1 — 本附件假设一个服务描述或者不被使用, 或者不包含描述快速 Web 服务能力信息 (或者这类信息 (如果存在) 被忽略掉), 如第 12 节中所述。

本附件描述了三种策略, 一种乐观的 (见第 D.1 节) 和两种悲观的 (见第 D.2 节)。

注 2 — 这些策略的其中一部分的重要性和有效性受到多数 Web 服务对于一个连接是否使用一个单一请求/响应以及对一个特定服务的信息是否进行缓存的影响。

注 3 — 当客户机确定某个 SOAP 节点支持快速 Web 服务的能力之后, 它就可以不必再应用请求提示 (见第 D.2.1 节) 或响应能力 (见第 D.2.2 节)。然而, 对客户端和 SOAP 节点都进行缓存的能力要谨慎使用, 因为两种实现都可能会改变。一个 SOAP 节点的能力只能通过服务描述或通过在一个 HTTP 连接的生命周期里已经确定的部分来保证。HTTP/1.1 有使连接“保持活跃”的能力以便多个请求/响应对可以在同一个连接上进行发送。

#### D.1 乐观策略

**D.1.1** 当使用该策略时, 一个支持快速 Web 服务的客户端乐观地假设相关联的 SOAP 节点是支持快速 (Web 服务) 的, 而且能够处理 ASN.1 SOAP 消息请求并以 ASN.1 SOAP 消息响应作为应答。

**D.1.2** 一个 SOAP 节点接收到一个 ASN.1 SOAP 消息后可能会有两种结果:

a) 该 SOAP 节点给出一个 400 系列分类的 HTTP 客户机错误状态码 (见 RFC 2616, 第 10.4 节) 作为应答。支持快速 Web 服务的客户端应期待一个“415 Unsupported Media Type” HTTP 状态码, 但也要能处理其他的状态码, 即“400 Bad Request”; 或

注 1 — “415 Unsupported Media Type” 将发生是因为 SOAP 节点不支持 ASN.1 SOAP 消息的 HTTP 媒体类型, 因此它是不支持快速 (Web 服务) 的。

注 2 — HTTP 提供一种末端开放机制来支持由 HTTP 扩展定义的状态码。一个符合规范的 HTTP 应用必须将任何不可识别的状态码作为与“400 Bad Request”状态码等价来对待。

b) 该 SOAP 节点以一个 ASN.1 SOAP 消息响应作为应答。

**D.1.3** 如果出现第 D.1.2 节 a 中所描述的情况, 则乐观策略就失败了, 且该支持快速 Web 服务的客户端只能再发送一个语义上等价的 XML SOAP 消息来交互, 或者采用如第 D.2 节所述的悲观策略。

**D.1.4** 如果出现第 D.1.2 节 b 中所描述的情况, 则乐观策略在第一次请求时就成功了。

#### D.2 悲观策略

当使用该策略时, 一个支持快速 Web 服务的客户端悲观地假设相关的 SOAP 节点可能是不支持快速 (Web 服务) 的, 不能处理 ASN.1 SOAP 消息请求并返回 ASN.1 SOAP 消息响应。在第 D.2.1 节和第 D.2.2 节中描述了两种形式的悲观策略。

##### D.2.1 带有请求提示的悲观策略

**D.2.1.1** 支持快速 Web 服务的客户端发送一个 XML SOAP 消息, 其中带有与第 10.1.4 节“ASN.1 SOAP 的 HTTP 绑定”中所述的 HTTP 头部域 **Accept** 相对应的请求提示。该头部域 **Accept** 将包含该 ASN.1 SOAP 消息的 HTTP 媒体类型, “application/fastsoap” 和一个 XML SOAP 消息。

注 — 本策略利用服务器驱动、内容协商 (见 IETF RFC 2616, 第 12.1 节), 这是 HTTP/1.1 的一个特性。

ASN.1 SOAP 的 HTTP 绑定支持 HTTP/1.1 和 HTTP/1.0。《W3C SOAP Part 2》第 7.1.2 节推荐实现使用 HTTP/1.1。

**D.2.1.2** 一个 SOAP 节点接收到一个 XML SOAP 消息后可能会有两种结果:

- a) 该 SOAP 节点以一个 XML SOAP 消息来响应;
- b) 该 SOAP 节点以一个 ASN.1 SOAP 消息来响应。

**D.2.1.3** 如果第 D.2.1.2 节 a 中所描述的情况出现, 则该悲观策略就失败了, 因为该 SOAP 节点是不支持快速 (Web 服务) 的。

注 — 第 10.2.2 子节 (ASN.1 SOAP 的 HTTP 绑定) 保证了一个 SOAP 节点如果有这种能力, 将以一个 ASN.1 SOAP 消息作为响应。因此, 如果出现了失败, 能够确定该 SOAP 节点不支持快速 Web 服务。

**D.2.1.4** 如果第 D.2.1.2 节 b 中所描述的情况出现, 则该悲观策略在第一次响应时就成功了。

## **D.2.2 带有响应支持快速 (Web 服务) 能力的悲观策略**

**D.2.2.1** 支持快速 Web 服务的客户机发送一个在 HTTP 请求中不带有请求提示的 XML SOAP 消息。

**D.2.2.2** 一个 SOAP 节点接收到一个 XML SOAP 消息后可能会有两种结果:

- a) 该 SOAP 节点以一个不带有能力指示的 XML SOAP 消息来响应; 或
- b) 该 SOAP 节点以一个带有定义了如第 10.2.3 节所述的能力的 XML SOAP 消息来响应。

注 — 支持快速 (Web 服务) 的能力由一个 HTTP 头部域 **Fast-Enabled** 来声明。

**D.2.2.3** 如果出现第 D.2.2.2 节 a 中所描述的情况, 则该策略就失败了, 因为该 SOAP 是不支持快速 (Web 服务) 的。该支持快速 Web 服务的客户端只能再重发一个语义上等价的 XML SOAP 消息来进行交互。

注 — 第 10.2.3 子节 (ASN.1 SOAP 的 HTTP 绑定) 保证了如果一个节点是支持快速 (Web 服务) 的, 该 SOAP 节点将以一个 HTTP 头部域 **Fast-Enabled** 来响应。因此, 如果发生了失败, 可以确定该 SOAP 节点不支持快速 Web 服务。

**D.2.2.4** 如果出现第 D.2.2.2 节 b 中所描述的情况, 且该支持快速 Web 服务的客户端可以处理这个 HTTP 头部域 **Fast-Enabled**, 则该策略将在第二次请求时成功。

## 附件 E

## WSDL 1.1中面向SOAP的服务描述

(本附件不是本建议书 | 国际标准的组成部分)

本附件描述了将《WSDL 1.1》[2]与《WS-I Basic Profile 1.0》[3]一起作为编写面向 SOAP 的服务描述的一种语言来使用。

注 1 — 《WSDL 1.1》和《WS-I Basic Profile 1.0》中的术语在合适的时候被重用；因而，当涉及《WSDL 1.1》或《WS-I Basic Profile 1.0》中所述的 XML 元素和属性时，并不使用 W3C XML 信息集的术语。

本非正式的附件为名空间使用如下的名空间前缀：

soapbind	"http://schemas.xmlsoap.org/wsdl/soap/"
wsdl	"http://schemas.xmlsoap.org/wsdl/"
xsd	"http://www.w3.org/2001/XMLSchema"

注 2 — 对前缀的选择在语义上并不重要。

## E.1 WSDL 1.1中表示的面向SOAP的服务描述

**E.1.1** 符合《WS-I Basic Profile 1.0》中所述的概要的 WSDL 1.1 文档满足第 12 节所述的面向 SOAP 的服务描述和使用第 13 节所述的这种面向 SOAP 的服务描述的需求。

注 — 《WS-I Basic Profile 1.0》解释和修正了《WSDL 1.1》以提高互操作性。

**E.1.2** 一个为了描述将通过交互 SOAP 1.1[1]消息而被执行的具体操作的接口绑定（见第 E.4 节），被不加修改地翻译为一个为了描述将通过交互映射自 W3C SOAP 消息的 ASN.1 SOAP 消息而被执行的具体操作的 ASN.1 SOAP 接口绑定（见第 E.4.6 节）。

注 — 这保证了现在的 WSDL 1.1 文档不加修改就能描述快速 Web 服务。可以使用同一个绑定和网络地址（由一个 URI 所指定）用于接收输入消息和发送输出消息，这些消息是 SOAP 1.1 消息和（映射自 W3C SOAP 消息的）ASN.1 消息，以此来支持一个快速 Web 服务和一个 XML Web 服务。

## E.2 模式（Schema）

原始的模式集（见第 12.2 节）是在 **wsdl:types** 元素中使用 **xsd:schema** 元素声明的 XSD 模式的集合。

## E.3 抽象接口和抽象操作

**E.3.1** 一个抽象接口（见第 12.3.1 节）是在 **wsdl:definition** 元素中的一个 **wsdl:portType** 元素。抽象接口的集合是 **wsdl:definition** 元素中的所有 **wsdl:portType** 元素的集合。

**E.3.2** 一个抽象接口的一个抽象操作（见第 12.3.4 节）是该抽象接口中的一个 **wsdl:operation** 元素。抽象操作的集合是该抽象接口中所有 **wsdl:operation** 元素的集合。

**E.3.3** 操作的名字（见第 12.3.4 节 a）是该抽象操作的 **name** 属性的取值。

**E.3.4** 一个输入消息定义（见第 12.3.4 节 b）是该抽象操作中的一个 **wsdl:input** 元素。对基于文档（见第 12.3.8 节）和基于 RPC（见第 12.3.9 节）的操作而言，**wsdl:input** 元素始终存在。

注 — 《WS-I Basic Profile 1.0》限定了输入和输出消息定义的存在和顺序，即只支持单向操作（输入消息定义存在而输出消息定义不存在）和请求—应答操作（输入消息定义存在并被第一个描述，输出消息定义存在并被第二个描述）（在本概要中不支持要求—应答操作和通知操作）。

**E.3.5** 一个输出消息定义（见第 12.3.4 节 c）是该抽象操作中的一个 **wsdl:output** 元素（如果存在）。

**E.3.6** 故障消息定义（见第 12.3.4 节 d）是该抽象操作中的一个 **wsdl:fault** 元素。故障消息定义的集合是该抽象操作中所有 **wsdl:fault** 元素的集合。

**E.3.6.1** 故障消息定义的顶级 **element declaration**（见第 12.3.7 节）是全局 **element declaration**，它是 **wsdl:message** 元素的惟一 **wsdl:part** 元素的 **element** 属性的取值，被故障消息定义所引用（通过 **message** 属性）。

注 — 《WSDL 1.1》第 3.6 节为故障消息定义描述了如下的限制：**wsdl:message** 元素只包含一个 **wsdl:part** 元素；且该 **wsdl:part** 元素引用一个全局的 **element declaration**（使用 **element** 属性）。

**E.3.7** 输入消息定义或输出消息定义（见第 12.3.6 节）的形式由该抽象操作的绑定来描述（见第 E.4.8 节）。

注 1 — 《WS-I Basic Profile 1.0》（见第 5.3 节和第 5.3.1 节）取了由一个抽象接口的输入消息定义或输出消息定义所引用的 **wsdl:message** 元素的 **wsdl:part** 元素集合的子集，以便其形式只对应于第 12.3.6 节 a 或 b 中所述的情况。

注 2 — 《WS-I Basic Profile 1.0》将该抽象接口的所有抽象操作限制到了只有基于文档方式（见第 12.3.8 节）和基于 RPC 方式（见第 12.3.9 节）的抽象操作。

## E.4 接口绑定和操作绑定

**E.4.1** 一个接口绑定（见第 12.4 节）是一个包含 **soap:binding** 元素的 **wsdl:binding** 元素（在 **wsdl:definition** 元素中）。接口绑定的集合是在 **wsdl:definition** 中所有 **wsdl:binding** 元素的集合。

**E.4.2** 一个具体接口的传输（见第 12.4.2 节 c）的 URI 是 **soap:binding** 元素的 **transport** 属性的取值。如《WS-I Basic Profile 1.0》第 5.6.2 节中所述（需求 R2702），只支持 HTTP 传输并且 **transport** 属性的取值为“<http://schemas.xmlsoap.org/soap/http>”。该值为一个 ASN.1 SOAP 绑定接口（见第 E.4.6 节）指明了使用 ASN.1 SOAP 的 HTTP 绑定（见第 10 节）。

**E.4.3** 如果一个接口绑定符合《WS-I Basic Profile 1.0》中第 5.3 节和第 5.3.1 节中所述的文档—文字绑定，则该具体接口的方式就是文档方式的（见第 12.4.2 节 d）。

注 — 《WS-I Basic Profile 1.0》指明了文档—文字绑定是一个其操作绑定全部为文档—文字操作的接口绑定。

**E.4.4** 如果一个接口绑定符合《WS-I Basic Profile 1.0》中第 5.3 节和第 5.3.1 节中所述的 rpc-文字绑定，则该具体接口的方式就是文档方式的（见第 12.4.2 节 d）。

注 — 《WS-I Basic Profile 1.0》指明了 rpc-文字绑定是一个其操作绑定全部为 rpc-文字操作的接口绑定。

**E.4.5** 接口绑定可以可选地通过一种被称为是 ASN.1 SOAP 接口绑定注释的 WSDL 1.1 扩展手段来指明该具体接口支持快速 Web 服务（见第 12.4.2 节 e）。该注释是一个元素信息项，作为 **wsdl:binding** 元素的一个孩子在 **soapbind:binding** 元素之后出现，并带有：

- a) 一个取值为“binding”的[local name]特性；和
- b) 一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description”的[namespace name]（[名空间名称]）特性。

**E.4.6** 默认的，所有具体接口都支持快速 Web 服务并且都是 ASN.1 SOAP 接口绑定（见第 12.4.7 节）。

**E.4.7** 接口绑定可以可选地通过一种被称为是接口绑定对象标识符注释的 WSDL 1.1 扩展手段来指明分配给所有具体操作的一个对象标识符（见第 12.4.2 节 a 和第 12.4.3 节）。该注释是该 ASN.1 SOAP 接口绑定的[attributes]特性成员中的一个属性信息项（见第 E.4.5），并带有：

- a) 一个取值为“object-identifier”的[local name]（[本地名]）特性；
- b) 一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description”的[namespace name]特性；和
- c) 一个 [normalized value] 的特性，它是只使用“XMLNumberForm”编码成一个“XMLObjectIDValue”的对象标识符（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

**E.4.8** 一个操作绑定（见第 12.4.8 节）是该接口绑定中的一个 **wsdl:operation** 元素。操作绑定的集合是该接口绑定中所有 **wsdl:operation** 元素的集合。

**E.4.9** 如《WS-I Basic Profile 1.0》第 5.3 节和第 5.3.1 节所述的操作绑定的形式，指明了该接口抽象操作的相应消息定义形式（见第 12.3.6 节）。

**E.4.9.1** 如果该操作绑定符合一个如《WS-I Basic Profile 1.0》第 5.3 节和第 5.3.1 节所述的文档—文字操作，则输入消息定义和输出消息定义都是如第 12.3.6 节 a 中所述的形式。

**E.4.9.2** 对于第 12.3.6 节 a 中所述的形式，顶级 **element declaration** 是一个全局 **element declaration**，并且是显示地或隐式地通过 **soap:body** 元素引用的 **wsdl:part** 元素的 **element** 属性的取值。如果没有 **wsdl:part** 元素通过 **soap:body** 元素被引用，则没有顶级 **element declaration** 出现。

注 — 《WS-I Basic Profile 1.0》的需求 R2201、R2210、R2202、R2204、R2208 指明了有零或一个 **wsdl:part** 元素存在，并且 **element** 属性存在于（**type** 属性不存在）该元素中（如果存在）。

**E.4.9.3** 如果该操作绑定符合如《WS-I Basic Profile 1.0》第 5.3 节和第 5.3.1 节所述的一个 rpc-文字操作，则输入消息定义和输出消息定义都是如第 12.3.6 节 b 所述的形式。

**E.4.9.4** 对于第 12.3.6 节 b 中所述的形式，一个不带限制符的名字是一个引用的 **wsdl:part** 元素的 **name** 属性的取值，且相关联的顶级 **complex type definition** 或 **simple type definition** 是同一个 **wsdl:part** 元素的 **type** 属性的取值。零或多个不带限制符的名字的列表对应于从那些显示或隐式地由 **soap:body** 元素所引用（按《WS-I Basic Profile 1.0》第 5.4.1 节的顺序）的 **wsdl:part** 元素的列表中所获取（按照同样的顺序）的名字。

注 — 《WS-I Basic Profile 1.0》的需求 R2202、R2203、R2207、R2208 指明了有零或多个 **wsdl:part** 元素存在，并且 **type** 属性存在（**element**（元素）属性不存在）于一个元素中。需求 R2301 指明了 **wsdl:part** 元素在由 **soap:body** 元素（显示或隐式）所引用的元素列表中的顺序。

**E.4.10** 一个具体操作的一个 SOAP 动作 URI（见第 12.4.8 节 a）是操作绑定中 **soap:operation** 元素（如果存在）的 **soapAction**（SOAP 动作）属性的取值。

**E.4.11** 一个 SOAP 头部块定义（见第 12.4.8 节 b 和第 12.4.11 节）如下所述：

- a) 一个 **soap:header** 元素在操作绑定的 **wsdl:input** 元素或者在 **wsdl:output** 元素中；和
- b) 一个在 **soap:header** 元素中的 **soap:headerfault** 元素。

**E.4.11.1** 一个 SOAP 头部块定义的顶级 **element declaration** 是全局 **element declaration** 且为 **wsdl:message** 元素中的 **wsdl:part** 的 **element** 属性的取值，其中的元素都被 SOAP 头部块定义（分别通过 **part** 和 **message** 属性）所引用。

注 — 《WS-I Basic Profile 1.0》为 SOAP 头部块定义指明了 **wsdl:part** 元素的 **element** 属性存在（且 **type** 属性不存在）。

**E.4.12** 对象标识符可以作为一种对 WSDL 1.1 进行扩展的注释，为如下定义分配给一个顶级 **element declaration**（见第 12.4.8 节 c）：

- a) 一个输入消息定义或输出消息定义，作为一个 **element declaration** 对象标识符注释（见第 E.4.13 节），且它分别是输入消息定义或输出消息定义的一个属性；
- b) 一个故障消息定义，作为一个 **element declaration** 对象标识符注释（见第 E.4.13 节），且它是故障消息定义的一个属性；和
- c) 一个 SOAP 头部块定义；作为一个 **element declaration** 对象标识符注释（见第 E.4.13 节），且它是在该 SOAP 头部块注释的[**attributes**]特性的成员之中（见第 E.4.14 节）。

**E.4.13** 一个 **element declaration** 对象标识符注释（对 WSDL 1.1 的一种扩展）是一个属性信息项，并带有：

- a) 一个取值为“object-identifier”的[**local name**]特性；和
- b) 一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description”的[**namespace name**]特性；
- c) 一个[**normalized value**]特性，且为只使用“XMLNumberForm”作为“XMLOIDValue”编码的一个对象标识符（见 ITU-T X.680 建议书 | ISO/IEC 8824-1，第 32 节）。

**E.4.14** 一个 SOAP 头部块注释（对 WSDL 1.1 的一种扩展对应（如果存在）于存在于该接口绑定中的一个 SOAP 头部块定义。该注释是一个元素信息项，它是输入消息定义或输出消息定义中的元素，并带有：

- a) 一个取值为“header”的[**local name**]特性；
- b) 一个取值为“urn:ohn:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description”的[**namespace name**]特性；
- c) 在[**attributes**]特性的成员中的一个属性信息项，并带有：
  - 一个取值为“message”的[**local name**]特性；和
  - 一个等于在相应的 SOAP 头部块定义中 **message** 属性值的[**normalized value**]特性；
- d) [**attributes**]特性的成员中的一个属性信息项，并带有：
  - 一个取值为“part”的[**local name**]特性；和
  - 一个等于在相应的 SOAP 头部块定义中 **part** 属性值的[**normalized value**]特性。

**E.4.15** 默认的，顶级 **element declarations** 作为嵌入式 ASN.1 值来进行表示（见第 12.4.8 节 d）。

**E.4.16** 通过为如下定义包含一个对 WSDL 1.1 进行扩展的注释，一个顶级 **element declaration** 可以被表示为一棵子树（见第 12.4.8 节 d）：

- a) 一个输入消息定义或输出消息定义，作为一个 **element declaration** 子树注释（见第 E.4.17 节），分别是该输入消息定义或输出消息定义的一个属性；
- b) 一个故障消息定义，作为一个 **element declaration** 子树注释（见第 E.4.17 节），是该故障消息定义的一个属性；和
- c) 一个 SOAP 头部块定义，作为一个 **element declaration** 子树注释（见第 E.4.17 节），是在该 SOAP 头部块注释中的[attributes]特性的成员中（见第 E.4.14 节）。

**E.4.17** 一个 **element declaration** 子树注释（对 WSDL 1.1 的一种扩展）是一个属性信息项，并带有：

- a) 一个取值为“subtree”的[local name]；
- b) 一个取值为“urn:ohj:joint-iso-itu-t:asn1:generic-applications:fast-web-services:description”的[namespace name]特性；
- c) 一个取值为“1”或“true”的[normalized value]特性。

**E.4.17.1** 一个元素子树注释的[normalized value]特性的取值不是“1”或“true”（例如，“0”或“false”），则等价于忽略该注释。

**E.4.18** 通过一个 **element declaration** 子树注释和一个 **element declaration** 对象标识符进行注释的一个顶级 **element declaration** 等价于只通过一个 **element declaration** 子树注释进行注释的一个顶级 **element declaration**。

## 附 件 F

### 对象标识符值的分配

(本附件不是本建议书 | 国际标准的组成部分)

在本建议书 | 国际标准中分配了如下的对象标识符和对象描述符:

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

## 参 考 资 料

- [1] W3C Note, *Simple Object Access Protocol (SOAP) 1.1*, Don Box, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Nielsen, Satish Thatte, Dave Winer, *W3C Note*, 8 May 2000. (See <http://www.w3.org/TR/2000/NOTE-SOAP-20000508>.)
- [2] W3C Note, *Web Services Description Language (WSDL) 1.1*, Erik Christensen, Francisco Curbera, Greg Meredith, Sanjiva Weerawarana, *W3C Note*, 15 March 2001. (See <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>.)
- [3] WS-I, *WS-I Basic Profile Version 1.0, Final Material*, 16 April 2004. (See <http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html>.)





## ITU-T 系列建议书

A系列	ITU-T工作的组织
D系列	一般资费原则
E系列	综合网络运行、电话业务、业务运行和人为因素
F系列	非话电信业务
G系列	传输系统和媒质、数字系统和网络
H系列	视听及多媒体系统
I系列	综合业务数字网
J系列	有线网络和电视、声音节目及其它多媒体信号的传输
K系列	干扰的防护
L系列	电缆和外部设备其它组件的结构、安装和保护
M系列	电信管理，包括TMN和网络维护
N系列	维护：国际声音节目和电视传输电路
O系列	测量设备的技术规程
P系列	电话传输质量、电话设施及本地线路网络
Q系列	交换和信令
R系列	电报传输
S系列	电报业务终端设备
T系列	远程信息处理业务的终端设备
U系列	电报交换
V系列	电话网上的数据通信
X系列	数据网、开放系统通信和安全性
Y系列	全球信息基础设施、互联网协议问题和下一代网络
Z系列	用于电信系统的语言和一般软件问题