INTERNATIONAL STANDARD

ISO/IEC 29363

First edition
2008-06-15

Corrected version
2008-10-01

Information technology — Web Services Interoperability — WS-I Simple SOAP Binding Profile Version 1.0

Technologies de l'information — Interopérabilité des services du Web — Profil de liaison SOAP simple WS-I, version 1.0

Reference number
ISO/IEC 29363:2008(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 29363 was prepared by the Web Services Interoperability Organization (WS-I) and was adopted, under the PAS procedure, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

In this corrected version of ISO/IEC 29363:2008, the full stop in the last paragraph of 2.3 has been moved from before “Appendix B” to the end of the sentence.

Information technology — Web Services Interoperability — WS-I Simple SOAP Binding Profile Version 1.0

1 Scope and introduction

1.1 Scope

This International Standard defines the WS-I Simple SOAP Binding Profile 1.0 (hereafter, "Profile"), consisting of a set of non-proprietary Web services specifications, along with clarifications to and amplifications of those specifications which promote interoperability.

Section 1 introduces the Profile, and explains its relationships to other profiles.

Section 2, "Profile Conformance", explains what it means to be conformant to the Profile.

Each subsequent section addresses a component of the Profile, and consists of two parts: an overview detailing the component specifications and their extensibility points, followed by subsections that address individual parts of the component specifications. Note that there is no relationship between the section numbers in this International Standard and those in the referenced specifications.

1.2 Relationships to Other Profiles

This Profile is derived from those Basic Profile 1.0 requirements related to the serialization of the envelope and its representation in the message, incorporating any errata to date. These requirements have been factored out of the Basic Profile 1.1 to enable other Profiles to be composable with it.

A combined claim of conformance to both the Basic Profile 1.1 and the Simple SOAP Binding Profile 1.0 is roughly equivalent to a claim of conformance to the Basic Profile 1.0.

This Profile composed with the Basic Profile 1.1 supersedes the Basic Profile 1.0.

1.3 Notational Conventions

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119.

Normative statements of requirements in the Profile (i.e., those impacting conformance, as outlined in "Conformance Requirements") are presented in the following manner:

Rnnnn*Statement text here.*

where "nnnn" is replaced by a number that is unique among the requirements in the Profile, thereby forming a unique requirement identifier.

Requirement identifiers can be considered to be namespace qualified, in such a way as to be compatible with QNames from Namespaces in XML. If there is no explicit namespace prefix on a requirement's identifier (e.g., "R9999" as opposed to "bp10:R9999"), it should be interpreted as being in the namespace identified by the conformance URI of the document section it occurs in. If it is qualified, the prefix should be interpreted according to the namespace mappings in effect, as documented below.

Some requirements clarify the referenced specification(s), but do not place additional constraints upon implementations. For convenience, clarifications are annotated in the following manner: c

Some requirements are derived from ongoing standardization work on the referenced specification(s). For convenience, such forward-derived statements are annotated in the following manner: xxxx, where "xxxx" is an identifier for the specification (e.g., "WSDL20" for WSDL Version 2.0). Note that because such work was not complete when this document was publiished, the specification that the requirement is derived from may change; this information is included only as a convenience to implementers.

This specification uses a number of namespace prefixes throughout; their associated URIs are listed below. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

- **wsdl** - "http://schemas.xmlsoap.org/wsdl/"
- **soapbind** - "http://schemas.xmlsoap.org/wsdl/soap/"
- **uddi** - "urn:uddi-org:api_v2"

1.4 Profile Identification and Versioning

This document is identified by a name (in this case, Simple SOAP Binding Profile) and a version number (here, 1.0). Together, they identify a particular *profile instance*.

Version numbers are composed of a major and minor portion, in the form "major.minor". They can be used to determine the precedence of a profile instance; a higher version number (considering both the major and minor components) indicates that an instance is more recent, and therefore supersedes earlier instances.

Instances of profiles with the same name (e.g., "Example Profile 1.1" and "Example Profile 5.0") address interoperability problems in the same general scope (although some developments may require the exact scope of a profile to change between instances).

One can also use this information to determine whether two instances of a profile are backwards-compatible; that is, whether one can assume that conformance to an earlier profile instance implies conformance to a later one. Profile instances with the same name and major version number (e.g., "Example Profile 1.0" and "Example Profile 1.1") MAY be considered compatible. Note that this does not imply anything about compatibility in the other direction; that is, one cannot assume that conformance with a later profile instance implies conformance to an earlier one.

2 Profile Conformance

Conformance to the Profile is defined by adherence to the set of *requirements* defined for a specific *target*, within the *scope* of the Profile. This section explains these terms and describes how conformance is defined and used.

2.1 Conformance Requirements

Requirements state the criteria for conformance to the Profile. They typically refer to an existing specification and embody refinements, amplifications, interpretations and clarifications to it in order to improve interoperability. All requirements in the Profile are considered normative, and those in the specifications it references that are in-scope (see "Conformance Scope") should likewise be considered normative. When requirements in the Profile and its referenced specifications contradict each other, the Profile's requirements take precedence for purposes of Profile conformance.

Requirement levels, using RFC2119 language (e.g., MUST, MAY, SHOULD) indicate the nature of the requirement and its impact on conformance. Each requirement is individually identified (e.g., R9999) for convenience.

For example;

> R9999 *WIDGETs SHOULD be round in shape.*

This requirement is identified by "R9999", applies to the target WIDGET (see below), and places a conditional requirement upon widgets; i.e., although this requirement must be met to maintain conformance in most cases, there are some situations where there may be valid reasons for it not being met (which are explained in the requirement itself, or in its accompanying text).

Each requirement statement contains exactly one requirement level keyword (e.g., "MUST") and one conformance target keyword (e.g., "MESSAGE"). Additional text may be included to illuminate a requirement or group of requirements (e.g.,

rationale and examples); however, prose surrounding requirement statements must not be considered in determining conformance.

Definitions of terms in the Profile are considered authoritative for the purposes of determining conformance.

None of the requirements in the Profile, regardless of their conformance level, should be interpreted as limiting the ability of an otherwise conforming implementation to apply security countermeasures in response to a real or perceived threat (e.g., a denial of service attack).

2.2 Conformance Targets

Conformance targets identify what artifacts (e.g., SOAP message, WSDL description, UDDI registry data) or parties (e.g., SOAP processor, end user) requirements apply to.

This allows for the definition of conformance in different contexts, to assure unambiguous interpretation of the applicability of requirements, and to allow conformance testing of artifacts (e.g., SOAP messages and WSDL descriptions) and the behavior of various parties to a Web service (e.g., clients and service instances).

Requirements' conformance targets are physical artifacts wherever possible, to simplify testing and avoid ambiguity.

The following conformance targets are used in the Profile:

- **ENVELOPE** - the serialization of the soap:Envelope element and its content (from ISO/IEC 29361)
- **MESSAGE** - protocol elements that transport the ENVELOPE (e.g., SOAP/HTTP messages) (from ISO/IEC 29361)
- **DESCRIPTION** - descriptions of types, messages, interfaces and their concrete protocol and data format bindings, and the network access points associated with Web services (e.g., WSDL descriptions) (from ISO/IEC 29361)
- **INSTANCE** - software that implements a wsdl:port or a uddi:bindingTemplate (from ISO/IEC 29361)
- **RECEIVER** - software that consumes a message according to the protocol(s) associated with it (e.g., SOAP processors) (from ISO/IEC 29361)

2.3 Conformance Scope

The scope of the Profile delineates the technologies that it addresses; in other words, the Profile only attempts to improve interoperability within its own scope. Generally, the Profile's scope is bounded by the specifications referenced by it.

The Profile's scope is further refined by extensibility points. Referenced specifications often provide extension mechanisms and unspecified or open-ended configuration parameters; when identified in the Profile as an extensibility point, such a mechanism or parameter is outside the scope of the Profile, and its use or non-use is not relevant to conformance.

Note that the Profile may still place requirements on the use of an extensibility point. Also, specific uses of extensibility points may be further restricted by other profiles, to improve interoperability when used in conjunction with the Profile.

Because the use of extensibility points may impair interoperability, their use should be negotiated or documented in some fashion by the parties to a Web service; for example, this could take the form of an out-of-band agreement.

The Profile's scope is defined by the referenced specifications in Appendix A, as refined by the extensibility points in Appendix B.

2.4 Claiming Conformance

Claims of conformance to the Profile can be made using the following mechanisms, as described in Conformance Claim Attachment Mechanisms, when the applicable Profile requirements associated with the listed targets have been met:

- **WSDL 1.1 Claim Attachment Mechanism for Web Services Instances** - MESSAGE DESCRIPTION INSTANCE RECEIVER
- **WSDL 1.1 Claim Attachment Mechanism for Description Constructs** - DESCRIPTION
- **UDDI Claim Attachment Mechanism for Web Services Instances** - MESSAGE DESCRIPTION INSTANCE RECEIVER

The conformance claim URI for this Profile is "http://wsi.org/Profiles/SimpleSoapBinding/1.0".

3 Messaging

This section of the Profile incorporates the following specifications by reference, and defines extensibility points within them:

- Simple Object Access Protocol (SOAP) 1.1
- Extensible Markup Language (XML) 1.0 (Second Edition)
- Namespaces in XML 1.0
- RFC2616: Hypertext Transfer Protocol -- HTTP/1.1

3.1 Message Serialization

SOAP 1.1 defines an XML structure for transmitting messages, the envelope. The Profile mandates the use of that structure, and places the following constraints on its use:

3.1.1 XML Envelope Serialization

R9700 *A MESSAGE MUST serialize the envelope as the exclusive payload of the HTTP entity-body.*

R9701 *A MESSAGE MUST serialize the envelope as XML 1.0.*

R9702 *A MESSAGE MUST have a "Content-Type" HTTP header field.*

R9703 *A MESSAGE's "Content-Type" HTTP header field MUST have a field-value whose media type is "text/xml".*

3.1.2 XML Namespace declarations

Although published errata NE05 (see http://www.w3.org/XML/xml-names-19990114-errata) allows this namespace declaration to appear, some older processors considered such a declaration to be an error. This requirement ensures that conformant artifacts have the broadest interoperability possible.

R9704 *An ENVELOPE SHOULD NOT contain the namespace declaration xmlns:xml="http://www.w3.org/XML/1998/namespace".*c

3.1.3 Unicode BOMs

XML 1.0 allows UTF-8 encoding to include a BOM; therefore, receivers of envelopes must be prepared to accept them. The BOM is mandatory for XML encoded as UTF-16.

R4001 *A RECEIVER MUST accept envelopes that include the Unicode Byte Order Mark (BOM).*c

3.1.4 XML Declarations

Presence or absence of an XML declaration does not affect interoperability. Certain implementations might always precede their XML serialization with the XML declaration.

R1010 *A RECEIVER MUST accept messages with envelopes that contain an XML Declaration.* c

3.1.5 Character Encodings

The Profile requires XML processors to support the "UTF-8" and "UTF-16" character encodings, in order to aid interoperability.

As a consequence of this, in conjunction with SOAP 1.1's requirement to use the "text/xml" media type (which has a default character encoding of "us-ascii") on envelopes, the "charset" parameter must always be present on the envelope's content-type. A further consequence of this is that the encoding pseudo-attribute of XML declaration within the message is always ignored, in accordance with the requirements of both XML 1.0 and RFC3023, "XML Media Types".

The "charset" parameter of Content-Type HTTP header field must be used to determine the correct character encoding of the message, in absence of a

"charset" parameter, the default value for charset (which is "us-ascii") must be used.

R1012 *A MESSAGE MUST serialize the envelope using either UTF-8 or UTF-16 character encoding.*

R1018 *A MESSAGE's "Content-Type" HTTP header field-value MUST indicate the correct character encoding, using the "charset" parameter.* c

R1019 *A RECEIVER MUST ignore the encoding pseudo-attribute of the envelope's XML declaration in a message.*

4 Description

This section of the Profile incorporates the following specifications by reference, and defines extensibility points within them:

- WSDL 1.1, Section 3

WSDL 1.1 defines a SOAP binding extension for describing messages serialized as SOAP envelopes. The Profile mandates the use of that structure, and places the following constraints on its use:

4.1 Bindings

4.1.1 SOAP Binding Extensions

The Profile limits the choice of WSDL bindings to the well defined and most commonly used WSDL SOAP binding. WSDL 1.1 defined binding extensions for HTTP GET/POST and MIME, or any other attachments technology, are not permitted by the Profile.

R9802 *A* `wsdl:binding` *element in a DESCRIPTION MUST only use the WSDL SOAP Binding as defined in WSDL 1.1 Section 3.*

R9800 *In a DESCRIPTION WSDL binding extension elements and attributes which cause messages on the wire to be non-conformant to the Profile MUST NOT be used.* c

R9801 *In a DESCRIPTION the WSDL MIME and HTTP GET/POST and DIME binding extensions MUST NOT appear in the SOAP binding.* c

Note that this places a requirement on the construction of conformant `wsdl:binding` elements. It does not place a requirement on descriptions as a whole; in particular, it does not preclude WSDL documents from containing non-conformant `wsdl:binding` elements.

4.1.2 Unbound portType Element Contents

WSDL 1.1 is not explicit about whether it is permissible for a `wsdl:binding` to leave the binding for portions of the content defined by a `wsdl:portType` unspecified.

R2209 *A* `wsdl:binding` *in a DESCRIPTION SHOULD bind every* `wsdl:part` *of a* `wsdl:message` *in the* `wsdl:portType` *to which it refers to one of* `soapbind:body`, `soapbind:header`, `soapbind:fault` *or* `soapbind:headerfault`.

A portType defines an abstract contract with a named set of operations and associated abstract messages. Although not disallowed, it is expected that every part of the abstract input, output and fault messages specified in a portType is bound to `soapbind:body` or `soapbind:header` (and so forth) as appropriate when using the SOAP binding as defined in WSDL 1.1 Section 3. Un-bound wsdl:parts should be ignored.

Appendix A: Referenced Specifications

The following specifications' requirements are incorporated into the Profile by reference, except where superseded by the Profile:

- Simple Object Access Protocol (SOAP) 1.1
- Extensible Markup Language (XML) 1.0 (Second Edition)
- Namespaces in XML 1.0
- RFC2616: Hypertext Transfer Protocol -- HTTP/1.1
- WSDL 1.1, Section 3

Appendix B: Extensibility Points

This section identifies extensibility points, as defined in "Scope of the Profile," for the Profile's component specifications.

These mechanisms are out of the scope of the Profile; their use may affect interoperability, and may require private agreement between the parties to a Web service.

None.

Appendix C: Normative References

In addition to all of the profiled specifications listed in Appendix A, the following specifications are referenced:

- RFC2119, http://ietf.org/rfc/rfc2119, Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, March 1997.
- WS-I Basic Profile 1.0, http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html, K. Ballinger et al., April 2004.
- Namespaces in XML 1.0 (Second Edition), http://www.w3.org/TR/2006/REC-xml-names-20060816, T. Bray et al., August 2006.
- WS-I Conformance Claim Attachment Mechanisms Version 1.0, http://www.ws-i.org/Profiles/ConformanceClaims-1.0-2004-11-15.html, M. Nottingham et al., November 2004.

Appendix D: Acknowledgements

This document is the work of the WS-I Basic Profile Working Group, whose members have included:

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

SHIMAYA (NTT), David Shoaf (Hewlett-Packard), Yasser Shohoud (Microsoft Corp.), David Smiley (Ascential Software), Seumas Soltysik (IONA Technologies), Joseph Stanko (Plumtree Software Inc.), Andrew Stone (Accenture), Julie Surer (MITRE Corp.), YASUO TAKEMOTO (NTT), Nobuyoshi Tanaka (NEC Corporation), Jorgen Thelin (Microsoft Corp.), Sameer Vaidya (Talking Blocks, Inc.), William Vambenepe (Hewlett-Packard), Claus von Riegen (SAP AG), Rick Weil (Eastman Kodak Company), Scott Werden (WRQ Inc.), Ajamu Wesley (IBM), Ian White (Micro Focus), Dave Wilkinson (Vignette), Mark Wood (Eastman Kodak Company), Prasad Yendluri (webMethods Inc.), and Brandon Zhu (NetManage Inc).

ICS 35.100.05

Price based on 13 pages