



Web Services Standards Profile

Version 1.0 — 31/01/2006

For Comment

National E-Health Transition Authority Ltd

Level 25
56 Pitt Street
Sydney, NSW, 2000
Australia.

www.nehta.gov.au

Disclaimer

NEHTA makes the information and other material (“Information”) in this discussion document available in good faith but without any representation or warranty as to its accuracy or completeness. NEHTA cannot accept any responsibility for the consequences of any use of the Information. As the Information is of a general nature only, it is up to any person using or relying on the Information to ensure that it is accurate, complete and suitable for the circumstances of its use.

Document Control

This document is maintained in electronic form. The current revision of this document is located on the NEHTA website www.nehta.gov.au and is uncontrolled in printed form. It is the responsibility of the user to verify that this copy is of the latest revision.

Trademarks

Company, product, and service names mentioned herein may be trademarks or service marks; such marks are the property of their respective owners.

Copyright © 2006, NEHTA.

This document contains information which is protected by copyright. All Rights Reserved. No part of this work may be reproduced or used in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems—without the permission of NEHTA. All copies of this document must include the copyright and other information contained on this page.

1	Introduction.....	1
1.1	Background.....	1
1.2	Purpose.....	1
1.3	Scope	1
1.4	Feedback.....	1
2	Service oriented architecture.....	2
2.1	Recommendation	2
2.2	Benefits of SOA	2
3	Web services.....	3
3.1	Recommendation	3
3.2	Benefits of Web services	3
3.3	Overview of Web services	3
3.4	Standards.....	4
3.4.1	HTTP 1.1.....	5
3.4.2	SOAP 1.2	6
3.4.3	MTOM and XOP	6
3.4.4	WS-Addressing	7
3.4.5	WSDL 2.0.....	7
3.4.6	WS-Security	7
3.4.7	WS-ReliableMessaging	8
4	References.....	9

This page intentionally left blank.

1 Introduction

1.1 Background

NEHTA's Secure Messaging initiative provides specifications for secure messaging in Australia's e-health environment. This document should be read in conjunction with *Towards a Secure Messaging Environment: An E-Health Transition Strategy*, available from www.nehta.gov.au.

1.2 Purpose

This document identifies the Web Services specifications that NEHTA recommends for application-to-application secure messaging.

1.3 Scope

This document only covers the specifications for application-to-application type messaging. It does not cover person-to-person type messaging.

1.4 Feedback

Feedback on this document is being sought, particularly from health and messaging software vendors, public and private health institutions and government health jurisdictions. Feedback will be considered by NEHTA when finalising this Profile, which will then be released through NEHTA's website. All comments should be directed to either:

Email: securemessaging@nehta.gov.au

Post: Secure Messaging Initiative
NEHTA
Suite A, Level 17
300 Adelaide Street Brisbane Queensland 4000

An electronic feedback form is also located on NEHTA's website www.nehta.gov.au.

2 Service oriented architecture

2.1 Recommendation

NEHTA recommends the use of a Service Oriented Architecture (SOA) approach for the design of health applications.

Health applications should be designed using a service oriented architecture, where appropriate. The SOA approach is an architectural style where services are designed to perform self-contained units of work. In this context, the term “service” refers to the functionality provided by a piece of software. In SOA, services have well defined and documented interfaces. The services are invoked by sending messages to the service interfaces, and applications are created from a set of interacting services.

2.2 Benefits of SOA

Systems designed using SOA can be aligned to the organisation’s business processes. This should lead to systems which better support the needs of the organisation.

The use of an SOA approach leads to more reusable and adaptable systems. This is important for e-health applications, because reuse can lead to improved efficiencies and adaptability can lead to systems which are better suited to the needs of providing health care.

The SOA approach creates a system where there is loose coupling between the services. This allows the system to be flexible and extensible—capable of adapting to change. This is important because fixed systems cannot be easily modified to meet the emerging and changing needs of a dynamically changing health care environment. Traditional approaches, such as object oriented programming and building monolithic applications, are not flexible enough to meet the needs of the e-health environment.

3 Web services

3.1 Recommendation

NEHTA recommends the use of Web services technology to implement health applications.

NEHTA recommends that Web services should be used to conduct secure messaging between health applications.

The term “Web services” refers to a specific set of technologies for application-to-application communications. It refers to using: Extensible Markup Language (XML) for representing data, SOAP as the protocol, Web Services Description Language (WSDL) for describing services, and other related Web service standards which will be described later in this chapter.

Web services does not refer to a Web browser interacting with a series of Web pages that make up an application—which is sometimes incorrectly referred to as a “web service.”

3.2 Benefits of Web services

The e-health environment is distributed and heterogeneous. It is spread out in many different locations across the country, and is made up of many different types of providers. These providers use different types of software—from many different vendors and running on many different types of platforms.

The features of Web services are:

- It is independent of how the applications are implemented. This means different organisations can choose to develop their applications using different tools and languages, and interoperability is still possible between them.
- It is interoperable across different computing platforms. This means different organisations can choose to deploy their applications on different platforms, and interoperability is still possible between them.
- It is vendor neutral. This means different organisations can choose to source their applications from different vendors, and interoperability is still possible between them.
- There is considerable momentum behind Web services. This means that the e-health environment can benefit from Web services developments in other industries. For example, it can benefit from the increased range of products and tools that support Web services, as well as an increasing number of people with skills and expertise with Web services.
- It is suitable for implementing a Service Oriented Architecture. This means it complements the recommended design approach.

The use of Web services will help provide interoperability in the e-health environment.

Web services should be used for application-to-application secure messaging. Application-to-application messaging is defined as the delivering of machine interpretable information.

3.3 Overview of Web services

Web services comprises of a family of standards for sending and receiving machine interpretable information.

The Web services standards each define a specific function, and they can be composed together in different combinations. The specifications complement each other, to form what is known as the “Web services stack.” The Web services stack can be divided up into a number of layers¹. These layers are illustrated in Figure 1.

- **Transport**, which defines how the messages are moved between the sender and the receiver.
- **Messaging**, which defines the structure and protocol for the messages. The messaging layer uses the transport layer mechanisms to send and receive the messages.
- **Description**, which defines mechanisms for describing the services which send and receive messages. Descriptions can be published so that others can know how to interact with the service. The description will include information about the service, as well as the messaging and transport protocols it uses.
- **Quality of service**, which provides assurances about the delivery of the messages. For example, the reliability or security of the messages.

In these layers, there can be a number of different specifications. There are a number of options available to choose from. However, one of the features of Web services is that these different specifications can work together. For example, a security specification can be independent of what transport specification is being used.

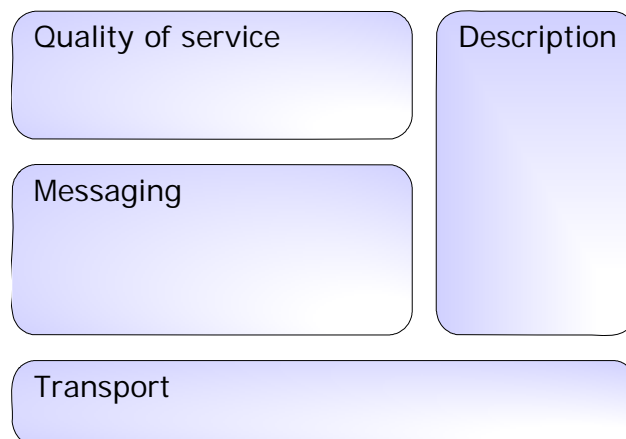


Figure 1 Web services stack

3.4 Standards

To ensure interoperability, a common set of standards must be used. NEHTA recommends a basic set of Web service standards which should be used for implementing e-health applications.

The recommended standards will be described in the rest of this section. They are illustrated in Figure 2, which shows how the standards fit into the Web services stack.

¹ There are more layers than those listed here. However, those layers contain advanced functionality that is not currently required in many e-health applications. Future recommendations from NEHTA may include additional specifications from those other layers.

These standards are recommended whenever messaging is conducted in the e-health environment—for communications between external organisations and providers. This does not prevent other standards from being used internally within organisations and in local applications. However, using the same standards internally is encouraged, because it would reduce the technical distinction between internal and external use. It may also simplify the development of application because a smaller set of specifications is required to support both types of communication.

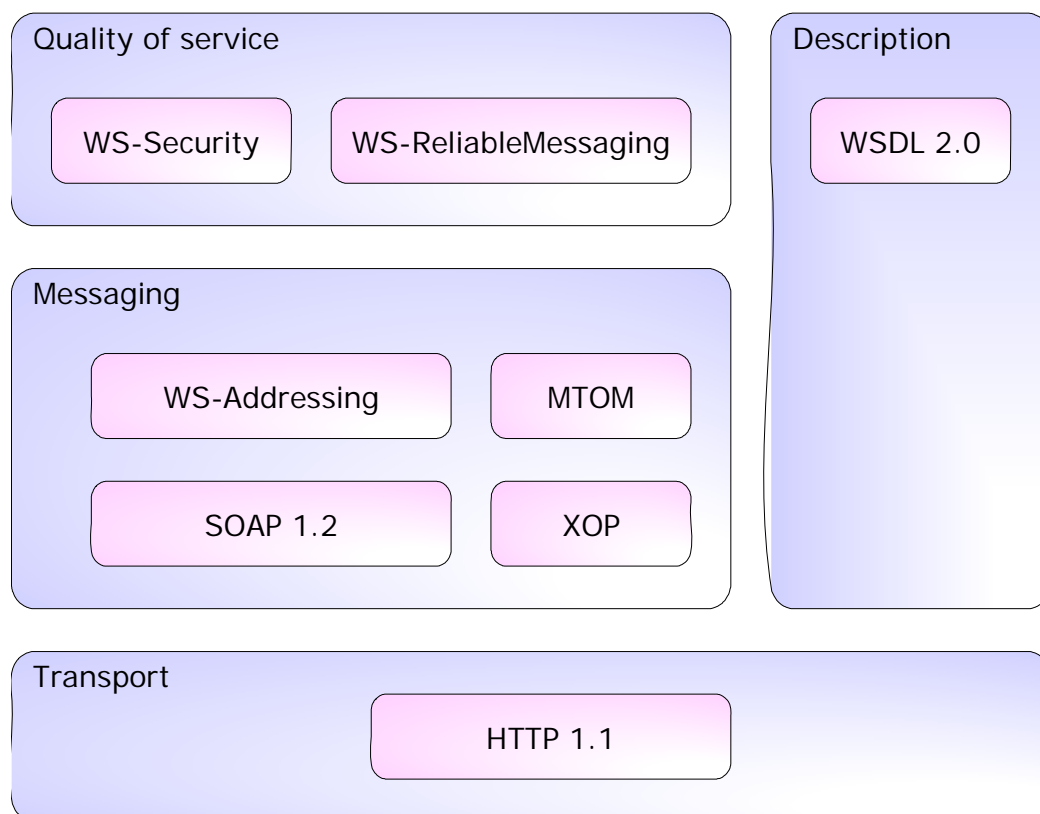


Figure 2 Recommended Web services standards

3.4.1 HTTP 1.1

NEHTA recommends the use of HyperText Transport Protocol version 1.1 as the transport mechanism for Web services [RFC2616].

HTTP 1.1 was originally developed for requesting and delivering Web content, but it is also the primary transport mechanism for Web services. It is a connection oriented, stateless mechanism. It supports content negotiation, and is extensible.

3.4.1.1 Motivation

HTTP 1.1 was chosen because it supports the development of Web services that uses an interactive paradigm. It is recognised that the majority of messaging currently conducted in health is non-interactive. However, it is expected that in the future more interactive applications will be developed. HTTP 1.1 is a single transport mechanism that is able to support both interactive and non-interactive style Web services.

In today's marketplace, HTTP 1.1 is the most widely supported transport mechanism for Web services. All of the major Web service vendors offer products that implement HTTP 1.1.

Other transport mechanisms are not recommended, because a proliferation of different mechanisms will make interoperability harder to achieve. A Web

service using one transport mechanism cannot directly exchange messages with a Web service that uses a different transport mechanism.

Connectionless transport mechanisms, such as Simple Mail Transport Protocol (SMTP) is not recommended. This is because it is not practical to build interactive Web services using them. They will not be useful to future e-health applications which require interactive services.

Secure HTTP (known as HTTPS or HTTP over SSL) is not recommended because it does not offer end-to-end security. Other security standards (namely WS-Security, which is described below) should be used to provide security, so the security provided by HTTPS is unnecessary.

3.4.2 SOAP 1.2

NEHTA recommends the use of SOAP 1.2 as the messaging protocol [SOAP2003a] [SOAP2003b].

SOAP defines a protocol for exchanging messages in a decentralised and distributed environment. It defines how messages are represented using the XML Infoset [INFO2004], which defines a set of XML data concepts. The Infoset is usually serialized using the XML 1.0 syntax [XML2004]. It also defines a framework for how the message are processed. SOAP is an extensible protocol which can support different types of payloads, behaviours, and interaction patterns.

3.4.2.1 Motivation

SOAP is the standard protocol for Web services. There are no other alternatives.

SOAP 1.2 is a W3C Recommendation. Although SOAP 1.1 is currently a part of the WS-I Basic Profile 1.1, it is not a W3C Recommendation and is not the focus for interoperable implementations of Web services. It is expected that the future of Web services will be based on SOAP 1.2 rather than SOAP 1.1.

3.4.3 MTOM and XOP

NEHTA recommends the use of the SOAP Message Transmission Optimization Mechanism (MTOM) and XML-binary Optimized Packaging (XOP) [MTOM2005] [XOP2005].

These two specifications may be used to reduce the size of messages containing binary data. Normally, SOAP messages are serialised using the XML syntax: a text based syntax in which binary data needs to be encoded using hexadecimal or base64 encoding (which increases the data size by a factor of 2 or 1.3, respectively). These specifications allow selected parts of the message to be transmitted as binary data, so there is very little size overheads.

3.4.3.1 Motivation

Transmitting binary data is sometimes necessary, to support messages containing legacy non-XML data formats or special data formats which must be in binary (e.g. image, audio, and video files). Sending these as binary data (instead of encoded text) will result in smaller message sizes which results in reduced transmission times.

The MTOM and XOP are W3C Recommendations, and is the standard mechanism for dealing with binary data.

Other approaches for dealing with binary data (such as: SOAP with Attachments, DIME and WS-Attachments) should not be used. Those approaches were earlier proposals, from which MTOM and XOP were developed. Although there are implementations of those earlier proposals, it is

expected that vendors will migrate (if they haven't done so already) to MTOM and XOP as the preferred approach.

3.4.4 WS-Addressing

NEHTA recommends the use of WS-Addressing [WSA2005].

WS-Addressing defines a mechanism for identifying messages and Web services endpoints. This information may be used in the processing of the messages.

3.4.4.1 Motivation

There may be situations where identifying an endpoint is important. For example, a receiver could apply different access control rules to a message, depending on where it originated from.

There may be situations where identifying a message is important. For example, to indicate in a result message which request message it corresponds to.

It is not mandatory that all messages use WS-Addressing. However, when addressing functionality is required, then WS-Addressing is the recommended mechanism for providing it.

WS-Addressing is a W3C Recommendation, and is the standard mechanism for addressing in Web services. There are no alternative standards for Web services addressing.

3.4.5 WSDL 1.1

NEHTA recommends the use of WSDL 1.1 for describing Web services [WSDL2001].

WSDL is a machine readable mechanism for describing what a service does, where it resides, and how to invoke it. Using this information, a program can be developed to invoke the Web service. The description can be used as formal documentation of the service, or it can be used as input for development tools and programs.

3.4.5.1 Motivation

The ability to describe a service is a fundamental part of the Web services stack.

WSDL 1.1 is not an official standard, but it is widely used and supported by current tools and products.

WSDL 2.0 is currently under development, and is at Last Call Working Draft stage (as of November 2005). It is expected that, once WSDL 2.0 becomes a W3C Recommendation and is supported by vendors, the NEHTA recommendation will be for WSDL 2.0 [WSDL2005a] [WSDL2005b] [WSDL2005c].

It is recommended that WSDL 1.1 should be used, but with a strategy to migrate to WSDL 2.0 when it becomes available.

3.4.6 WS-Security

NEHTA recommends the use of WS-Security to provide security for Web service messages [WSS2004].

WS-Security defines mechanisms for preserving the confidentiality and integrity of messages through the use of encryption and digital signatures. It provides a standard way of using XML Encryption to encrypt selected parts of the message [XENC2002], and it provides a standard way of using XML Digital

Signatures to associate a digital signature with parts of the message [DSIG2002].

It also provides a mechanism for sending security tokens as a part of the message. However, it does not dictate what types of tokens must be used. Examples of possible security tokens include: simple passwords, X.509 certificates, SAML Assertions, and Kerberos tickets. The types of security token that should be used will be determined by NEHTA's User Authentication initiative.

3.4.6.1 Motivation

WS-Security provides end-to-end security, which is important when messages are sent across multiple hops. The security must be maintained between the sender and the ultimate receiver. The intermediaries may or may not be trusted. WS-Security associates the security with the message, rather than the transport mechanism, so it is effective even if the intermediaries are not trusted.

Transport layer security mechanisms, such as SSL, TLS and IPsec, are not recommended because they only provide point-to-point security (across the single hop between one node and the next).

3.4.7 WS-ReliableMessaging

It is expected that NEHTA will recommend WS-ReliableMessaging for ensuring reliable delivery of messages.

WS-ReliableMessaging provides mechanisms to ensure that a message is delivered, even if messages are lost, duplicated, or reordered.

3.4.7.1 Motivation

Currently there are two competing standards for providing reliable delivery of messages: WS-Reliability and WS-ReliableMessaging.

- WS-ReliableMessaging is currently under development within the OASIS Web Services Reliable Exchange (WS-RX) technical committee.
- WS-Reliability is an OASIS Recommendation from the OASIS Web Services Reliable Messaging (WSRM) technical committee.

It is expected that WS-ReliableMessaging will be more widely adopted by industry, since it has the backing of more vendors. If that happens, then NEHTA will be recommending WS-ReliableMessaging as the standard approach to ensure the reliable delivery of messages.

4 References

- [DSIG2002] XML-Signature Syntax and Processing, W3C Recommendation, 12 February 2002, <<http://www.w3.org/TR/xmlsig-core/>>.
- [INFO2004] XML Information Set (Second Edition), W3C Recommendation, 4 February 2004, <<http://www.w3.org/TR/xml-infoset/>>.
- [MTOM2005] SOAP Message Transmission Optimization Mechanism, W3C Recommendation, 25 January 2005, <<http://www.w3.org/TR/soap12-mtom/>>.
- [RFC2616] IETF RFC 2616: Hypertext Transfer Protocol — HTTP/1.1, R. Fielding, J. Gettys, J. C. Mogul, H. Frystyk, T. Berners-Lee, January 1997, <<http://www.ietf.org/rfc/rfc2616.txt>>.
- [SOAP2003a] SOAP Version 1.2 Part 1: Messaging Framework, W3C Recommendation, 24 June 2003, <<http://www.w3.org/TR/soap12-part1/>>.
- [SOAP2003b] SOAP Version 1.2 Part 2: Adjuncts, W3C Recommendation, 24 June 2003, <<http://www.w3.org/TR/soap12-part2/>>.
- [WSA2005] Web Services Addressing 1.0 – Core, W3C Candidate Recommendation, 17 August , <<http://www.w3.org/TR/ws-addr-core>>.
- [WSDL2001] Web Services Description Language (WSDL) 1.1, W3C Note, 15 March 2001, <<http://www.w3.org/TR/wsdl>>.
- [WSDL2005a] Web Services Description Language (WSDL) Version 2.0 Part 0: Primer Language, W3C Working Draft, 3 August 2005, <<http://www.w3.org/TR/wsdl20-primer>>.
- [WSDL2005b] Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language, W3C Working Draft, 3 August 2005, <<http://www.w3.org/TR/wsdl20>>.
- [WSDL2005c] Web Services Description Language (WSDL) Version 2.0 Part 2: Adjuncts, W3C Working Draft, 3 August 2005, <<http://www.w3.org/TR/wsdl20-adjuncts>>.
- [WSI2005] Web Services Interoperability Organization, <<http://www.ws-i.org/>>.
- [WSS2004] Web Services Security 2004, Web Services Security: SOAP Message Security V1.0, OASIS Standard 200401, March 2004, <<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>>
- [WSSX2004] Web Services Security X.509 Certificate Token Profile, OASIS Standard 200401, March 2004, <<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0>>.
- [XENC2002] XML Encryption Syntax and Processing, W3C Recommendation, 10 December 2002, <<http://www.w3.org/TR/xmlenc-core/>>.
- [XML2004] Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation, 4 February 2004, <<http://www.w3.org/TR/REC-xml>>.
- [XOP2005] XML-binary Optimized Packaging, W3C Recommendation, 25 January 2005, <<http://www.w3.org/TR/xop10/>>.