AS4: Web Services for B2B

GS1 eTG White Paper

Issue 1, Approved, July 2011

Document Summary

Document Item	Current Value
Document Title	AS4: Web Services for B2B GS1 eTG White Paper
Date Last Modified	July 2011
Document Issue	Issue 1
Document Status	Approved
Document Description	This white paper explains the rationale for the AS4 messaging protocol, both from a business and an information technology point of view, and its relevance to the GS1 community. It explains how AS4 relates to previous work on e-business protocols in ebXML and within OASIS. The paper also explains AS4 in some details and shows the similarities, differences and added value of AS4 compared to both AS2 and to other Web Services profiles.

Contributors

Name	Organization
David Darnell	Systrends
John Duker	P&G
Dale Moberg	Axway Inc.
Pim van der Eijk	Sonnenglanz
Shan Welch	GS1 UK

Log of Changes in Issue 1.0

Issue No.	Date of Change	Changed By	Summary of Change
1 Draft 002	2011-02-08	Pim van der Eijk	First Draft for Review
1 Draft 003	2011-03-04	Pim van der Eijk	Added missing content in sections
1 Draft 004	2011-04-12	Pim van der Eijk	Clarifications and enhancements based on March conference call and review comments. Interim version for review by ebMS TC members.
1 Draft 005	2011-04-15	Pim van der Eijk	Draft for review by eTG members and discussion at April 21st call.
1 Draft 006	2011-05-11	Pim van der Eijk	Additional review comments, confirmed contributors added
1.0	2011-07-21	Pim van der Eijk	Motion to publish white paper approved after GS1 Community Review

Disclaimer

Whilst every effort has been made to ensure that the guidelines to use the GS1 standards contained in the document are correct, GS1 and any other party involved in the creation of the document HEREBY STATE that the document is provided without warranty, either expressed or implied, of accuracy or fitness for purpose, AND HEREBY DISCLAIM any liability, direct or indirect, for damages or loss relating to the use of the document. The document may be modified, subject to developments in technology, changes to the standards, or new legal requirements. Several products and company names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Table of Contents

1. Introduction

Companies in the GS1 community use B2B data exchange technology extensively. The most widely deployed B2B exchange protocol in the community is the EDIINT AS2 protocol. Companies have made significant investments in setting up AS2 connectivity to trading partners, and many exchanges are operational and function to their satisfaction. However, connectivity to Small and Medium-Size companies that cannot afford, or lack the skills, to operate an AS2 messaging gateway server successfully, remains a challenge for many companies. Recently, the new AS4 B2B protocol has been developed. It has all the functionality that AS2 offers today, but also provides better support for SMEs and is based on more modern Web Services technology. The GS1 eCom Technology Group has provided input to the development of the AS4 specification.

The goal of this white paper is to explain the rationale for AS4, both from a business and an information technology point of view, and its relevance to the GS1 community. It explains how AS4 relates to previous work on e-business protocols in ebXML and within OASIS. The paper also explains AS4 in some details and shows the similarities, differences and added value of AS4 compared to both AS2 and to other Web Services profiles. The paper concludes with a discussion on current status, positioning and deployment options for AS4.

2. Trends in B2B Integration

2.1. Industry Trends

The past decade has seen a massive increase in the use of B2B document exchange. Companies that already exchanged documents electronically have increased volumes of exchanged messages, the number of partners they exchange documents with and the number and diversity of business processes that are supported by B2B document exchange. Electronic document exchange has enabled the creation of new applications that did not exist in the paper-based world.

The increase in B2B exchanges has been supported by the rapid emergence of the Internet and the availability of standard exchange protocols. The Internet allows businesses to exchange data at far lower cost than using value-added network service providers. Some large retailers and companies in the consumer goods industry adopted the EDIINT AS2 protocol as a standard in 2002 and started requiring use of it by their suppliers. This created a network effect that resulted in strong adoption of AS2 across industries. AS2 also offers important business functionality such as authentication and non-repudiation of origin (based on message signatures), confidentiality (based on message encryption), and non-repudiation of receipt (based on exchange of signed message disposition notifications).

Historically, the adoption of electronic data interchange has been stronger among larger companies than among small and medium-size companies. AS2 has allowed many companies to use the public Internet instead of value-added networks and has therefore made EDI more affordable to more companies. However, AS2 requires companies to operate a messaging server that needs to be accessible from the Internet to receive messages from trading partners. The business case to operate such a server is weaker for a small or medium-size company that exchanges relatively few documents, and with a smaller number of partners, than a larger company. SMEs also often have a skills issue, and cannot guarantee service availability to the same extent as larger companies.

A common analogy is email: small companies can both send and receive email using only email clients using an infrastructure of SMTP relays offered by Internet Service Providers. These are analogous to AS2 gateways in B2B exchanges. But there is no AS2 equivalent to the POP3 and IMAP

functionality that allows businesses to use email clients to receive messages without running an email server themselves.

2.2. Technology Trends

An important and unmistakable trend in Information Technology in the past decade has been the emergence of Service Oriented Architecture (SOA) and the increasing use of Web Services to interconnect applications and services, especially within enterprises. Web Services are horizontal standards that are intended to be used for a very broad range of applications, which differ greatly in requirements. B2B document exchange is just one of these applications. To address this diversity of requirements, Web Services protocols have been developed (in organizations like W3C and OASIS) modularly, as separate specifications focussed on specific aspects (like packaging, security, reliable messaging). These modules are intended to be used in bundles to address specific use cases. While this modularity offers great flexibility and customizability, this choice has come at the expense of issues in interoperability, complexity and slow development and deployment of specifications.

One application area of Web Services specifications is B2B integration. The ebXML initiative, a joint project of UN/CEFACT and OASIS, was one of the first initiatives to seize on this opportunity. It pioneered the use of Web Services specifications like SOAP, SOAP with attachments, and XML security in the version 1.0 and 2.0 releases of the ebXML Messaging Services (ebMS) Specification. The ebMS 2.0 OASIS Standard, published in 2002, also innovated in areas of key importance for B2B like message security and reliable messaging, and inspired more specialized subsequent Web Services standards work.

The ebMS 3.0, Core Specification, standard, published in 2007, represents the next evolutionary step in Web Services for B2B. As the AS4 introduction puts it, ebMS 3.0 "*provides as single comprehensive specification for defining the secure and reliable exchange of documents using Web Services. ebMS 3.0 composes the fundamental Web Services standards like SOAP 1.1/1.2, SOAP with Attachments and MTOM, WS-Security 1.0/1.1, and WS-Reliability 1.1/WS-ReliableMessaging 1.1 together with guidance for the packaging of messages and receipts along with definitions of messaging choreographies for orchestrating document exchanges.*" The new standard recognized and took advantage of the emergence and availability of newer Web Services standards but also continued to innovate in areas of relevance to B2B integration. A key example of this is the "pull" mode feature which AS4 adopts and which we will discuss in section 4.3.

2.3. Requirements for a B2B Web Services Standard

Compared to AS2, which was designed a decade ago and is based on the Internet technology that was available at that time, a B2B protocol based on Web Services technology uses a more modern technical basis. However, this is hardly sufficient for such a protocol to be successful. Instead, a B2B Web Services protocol should:

- Provide all the functionality that established B2B protocols like AS2 provide today. This means support for AS2 features, like non-repudiation of receipt, that are not supported by Web Services specifications.
- Provide added value functionality that is not provided by AS2 or similar established protocols. A key feature is support for "client only" implementations that can be used by small and medium size businesses that cannot afford, or do not have the skills, to run a messaging gateway server.
- Provide flexibility in security: not all business processes require the use of electronic signatures. For instance, the European regulation on e-invoicing (part of the VAT Directive) specifies that electronic exchange of invoices does not require the use of electronic signatures. There are many situations in which message confidentiality (as provided by SSL/TLS transport-level security) is sufficient.

- Be simple, light-weight, easy to implement and lower-cost to set up and operate than traditional B2B protocols.

3. AS4: Web Services for B2B

In 2008, several software vendors and Drummond Group, the interoperability testing and certification group, came together to work on a Web Services interoperability profile for B2B. As the ebMS 3.0 Core Specification had just been published as an OASIS Standard and was designed for B2B integration, it was decided that this standard would be the base standard for this new B2B profile, which would be called Applicability Statement 4 (AS4), as a continuation of the AS1, AS2 and AS3 protocols for EDIINT developed in the Internet Engineering Task Force (IETF). To specify AS4, a Sub-Committee was set up within the OASIS ebXML Messaging Services TC in 2008, which delivered its specification as an OASIS Committee Specification in April 2010. The GS1 eCom Technology Group has provided input during the development of the AS4 specification. AS4 has delivered results on the following activities:

- Profile the ebMS 3.0 standard. While ebMS 3.0 constrains the use of the Web Services standards on top of which it is built, it inherits some of the interoperability issues in these standards. AS4 makes specific choices in areas where the ebMS 3.0 Core Specification allows multiple choices, to simplify and lower cost of implementation and to increase interoperability. For instance, ebMS 3.0 supports both SMTP and HTTP transport. The AS4 profile, like AS2, only supports HTTP.
- Add specific functionality to ebMS 3.0 to cover a complete feature set for EDIINT. This includes features like Receipts, “Reception Awareness” and payload compression.

The AS4 specification has been presented to the GS1 eCom Technology Groups on several occasions, and some key requirements for AS4 come from this community of AS2 users. AS4 provides two conformance profiles for ebMS 3.0.

- The **ebHandler** conformance profile supports both Sending and Receiving roles, and for each role both message pushing and message pulling.
- The **Light Client** conformance profile supports both Sending and Receiving roles, but only message pushing for Sending and message pulling for Receiving. In other words, it does not support incoming HTTP requests, and may have no IP address

As the AS4 specification states it: “*Using ebMS 3.0 as a base, a subset of functionality has been defined along with implementation guidelines adopted based on the “just-enough” design principles and AS2 functional requirements to trim down ebMS 3.0 into a more simplified and AS2-like specification for Web Services B2B messaging. The main benefits of AS4 compared to its previous version are compatibility with Web services standards, message pulling capability, and a built-in Receipt mechanism.*”

The AS4 profile has been developed in iterations, and an OASIS Committee Specification (CS) was approved in April 2010. Based on feedback from users, the OASIS ebXML Messaging Services TC has updated the specification and intends to put the AS4 profile specification out for approval as an OASIS Standard after the summer break in 2011. Several implementations of AS4 exist, including one open source implementation. The AS4 specification defines five conformance clauses, to which implementations can refer to explain their AS4 capabilities. The ebHandler and Light Client conformance clauses indicate full implementation of the ebHandler and Light Client conformance profiles, respectively. Two other defined conformance clauses are:

- The **Minimal Client** clause indicates an implementation supports the Light Client except for support for the Web Services Security X.509 Certificate Token profile. This AS4 profile does not support non-repudiation. It is useful as a low-end FTP replacement or for business processes that do not require signing and signed receipts.

- The **AS2/AS4 ebHandler** clause indicates an implementation supports both AS2 and AS4 and can be configured to use one or the other protocol per trading partner or exchange type. A product that conforms to this clause is of interest to companies that have an existing AS2 trading network they need to maintain, but want to use AS4 to connect to some of their current or new trading partners.

Finally, an optional capability is defined to allow AS4 messages to be exchanged via intermediary messaging nodes. Those nodes support store-and-forward capabilities and allow two client-only AS4 endpoints to exchange messages. This AS4 feature is a subset of functionality defined in Part 2, of ebMS version 3, Advanced Features.

4. AS4 from an AS2 Point of View

For users coming from an AS2 or other EDIINT background, AS4 provides a mix of familiar and new functionality. However, the wire format of AS4 messages and the exchange protocol are very different as the standards used to provide that functionality are based on Web Services. To see how AS4 relates to the familiar AS2 protocol, we will discuss:

- Functionality that is offered by both AS2 and AS4.
- AS4 business header functionality that has no counterpart in AS2, but is similar to the UN/CEFACT Standard Business Document Header (SBDH).
- AS4 enhancements over both AS2 and SBDH.

4.1. Functionality common to AS2 and AS4

AS4 is more than a profile of ebMS 3.0. In addition to the **ebHandler** and **Light Client** Conformance Profiles, it provides an AS4-specific Usage Profile that adds additional functionality. These additions make AS4 a functional superset of AS2, and ensure that AS4 offers the functionality that AS2 offers and that AS2 users have come to rely on in their day to day business. However, AS4 has a different technical basis and uses a different set of underlying standards than AS2. So in many cases a functional requirement is met using different technologies. The following table gives an overview of these common features and the base technologies used to provide them.

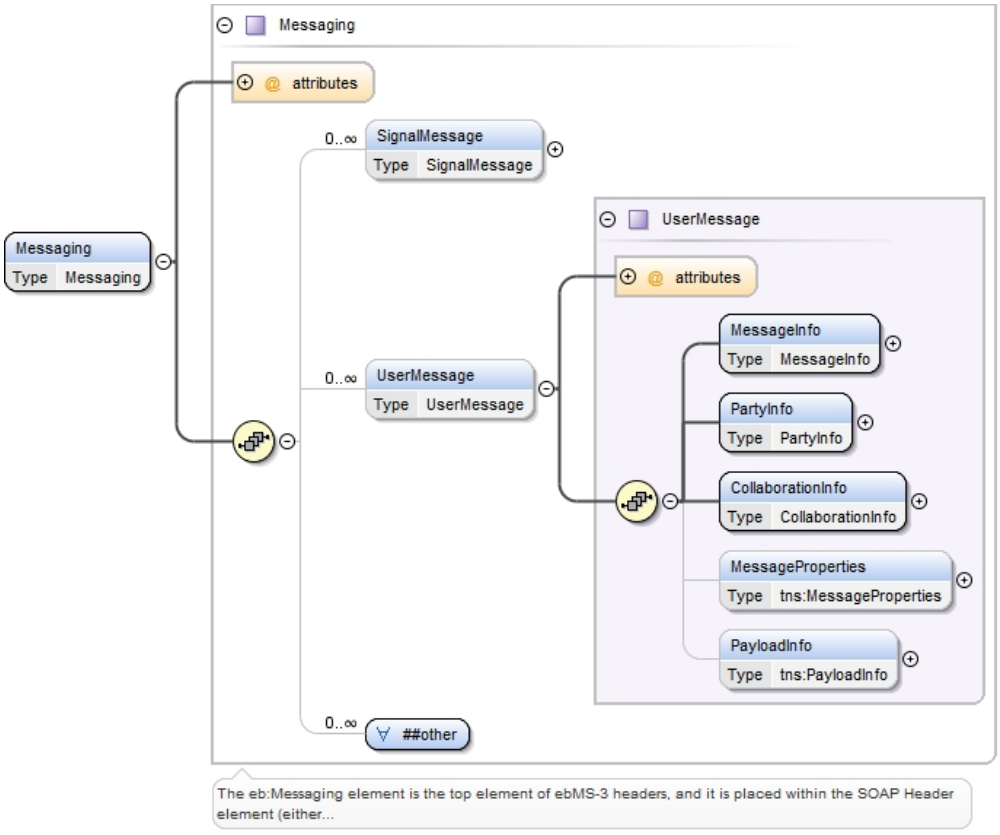
Functionality	AS2	AS4
Core Messaging	HTTP 1.1 and MIME	Web Services
Internet Transport	HTTP 1.1	HTTP 1.1
Transport Layer Integrity, Sender Authentication, Receiver Authentication and Message Confidentiality (Non-Persistent)	Transport Layer (SSL / TLS) Security (Optional)	Transport Layer (SSL / TLS) Security (Optional)
Message and Payload Packaging	MIME	SOAP 1.2 with attachments (MIME)
Message Identification	AS2 “Message-Id”	ebMS 3.0 “MessageId”
Message Timestamp	MIME “Date” header	ebMS 3.0 “Timestamp”
Party Identification	AS2 “From” and “To” system identifiers	ebMS 3.0 “From” and “To” party identifiers.

Functionality	AS2	AS4
Non-Repudiation of Origin	MIME Multipart/Signed (optional)	WS-Security 1.1 using XML Security (optional)
Message Confidentiality	MIME Multipart/Encrypted (optional)	WS-Security 1.1 using XML Encryption (optional)
Non-Repudiation of Receipt	Signed Message Disposition Notification	Signed Receipt Signal Message

As the table shows, AS4 is broadly comparable to AS2. Both offer a secure document exchange protocol for use over the Internet that leverages the MIME envelope structure to transport arbitrary payloads. The main difference is that an AS4 message contains a SOAP 1.2 envelope as the start MIME part. This SOAP envelope contains the ebMS 3.0-specific **Messaging** SOAP header and, optionally, SOAP headers for other purposes, such as WS-Security headers. The format and correct use of these other SOAP headers is defined in other Web Services standards, as specified in detail in ebMS 3.0.

4.2. AS4 User Message Header

One of the differences between ebMS 3.0 and both AS2 and Web Services is that ebMS 3.0 defines a standard SOAP header block to express message metadata. For user messages, the values of the elements in this header are an agreement between business partners, though some elements are optional and defaults are defined for some of the mandatory elements. The structure of the AS4 **Messaging** SOAP header is visualized in the following diagram.



From an AS4 perspective, this header is part of the message and is not a payload. It is processed by the AS4 Message Service Handler. The format and content of the AS4 user message header are similar to the header structure defined in the earlier ebMS 2.0 standard. The header allows the MSH to:

- route or deliver messaging to specific back-end applications using delivery criteria,
- monitor business activity with specific partners, services, or business process
- track messages based on AS4 headers only, and in a payload-agnostic fashion

The AS2 protocol defines a much more limited set of AS2 headers. In many situations, AS2 users need to convey additional metadata about the message or about a payload document in the message. The Standard Business Document Header (SBDH) is a standard XML notation to provide this information. It can be used as an XML header in business documents in AS2 messages. From an AS2 perspective, the SBDH is part of a payload, even though some AS2 products may offer generic support for processing documents with an SBDH header. The following table provides a comparison of the AS4 Messaging header and the SBDH.

AS4 Messaging	SBDH
The AS4 **PartyInfo** group contains information about the From and To parties.	The corresponding SBDH elements are the **Sender** and **Receiver** elements.
AS4 **From** and **To** elements have a **PartyId** element that can be **type**d to a particular type system, and a **Role** element.	The SBDH **Sender** and **Recipient** elements have an **Identifier** element. The value of the **Authority** attribute provides type information as in AS4. In addition to the Identifier element there is an optional **ContactInformation** group, which has no direct counterpart in AS4.
The AS4 **CollaborationInfo** group contains an optional **AgreementRef** and mandatory **Service**, **Action** and **ConversationId** elements.	The optional **BusinessScope** group in the SBDH and the related **BusinessScope** schema provide the elements **BusinessServiceName** and **ServiceTransaction** that have a similar purpose.
The optional **MessageProperties** group contains a series of arbitrary name/value properties.	SBDH has a similar extensibility mechanism based on XML schema type substitution.
The AS4 **PayloadInfo** group contains information about the business document, or business documents and any attachments to those documents. The payloads themselves are stored in separate MIME parts in the AS4 MIME message and referenced via the **href** attribute. For each part, if it is an XML part, there is an option to specify its XML schema. Properties are available for parts as they are for the AS4 message as a whole.	In SBDH, the **Manifest** group is used for (non-XML) attachments. The SBDH itself is part of a standard business document, i.e. an XML payload. Attachments can be in separate MIME parts as is the case in AS4. Since the mechanism is not intended for XML attachments, schema information is not expressed.

When using AS4 instead of AS2 to exchange documents, there are several options to handle situations where more metadata needs to be specified for the exchanged documents. The choice depends on the metadata required, the capabilities of the AS4 processor and its back-end interface, and the expectations of business applications:

- An AS4 user message can have multiple payloads, one of which can be an XML document with an SBDH structure. If the back-end application supplies the SBDH header in the payload

for an outbound message, a SBDH-aware AS4 processor could copy corresponding elements to the AS4 header. Similarly, it could match agreement between SBDH values and AS4 headers on inbound messages before passing on the document to the business application.

- If the metadata can be encoded using structures in the AS4 **UserMessage** SOAP header, and if there is no requirement to include an SBDH in the business document then there may be no need to use the SBDH. The metadata in the AS4 **UserMessage** can be exchanged with the back-end application in other ways.
- Additional information that is not encoded in the SBDH can be encoded using the **MessageProperties** element in the AS4 header.

4.3. AS4 Improvements over AS2

AS4 is not just a newer B2B protocol that offers AS2-like functionality. AS4 also offers functionality that is not currently available in AS2 and that increases the application domain of B2B messaging to a broader category of users and applications. The main functional enhancement offered by AS4 is its support for alternative message exchange styles. In the AS2 protocol, the sender of a message is connecting to the recipient by initiating, as an HTTP client, to the AS2 HTTP server of the recipient. In ebMS 3.0 and AS4 jargon, this is referred to as a "push" mode exchange, as the sender is the initiator of the transmission and determines when the transmission takes place.

In "push" exchanges:

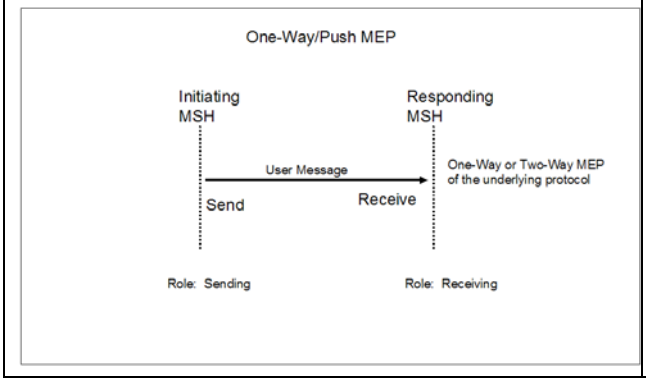
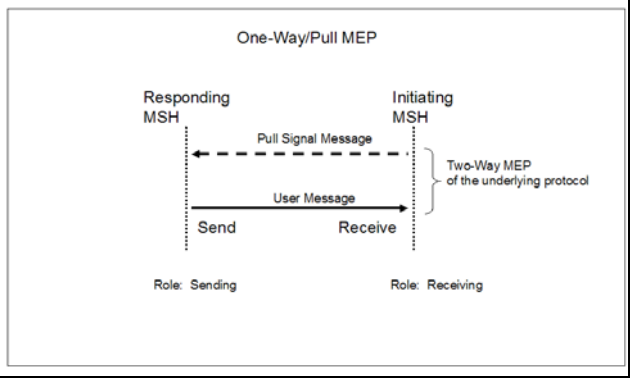
- The receiver is a responder and needs to have an MSH that is actively accepting incoming connections. This means that its AS2 server must be accessible on the network. It should have a resolvable URL or a known, fixed IP address. Firewalls and other networking equipment must be configured to allow the incoming HTTP requests to reach the AS2 server.
- The receiver must be able and ready to accept incoming connections when their business partners want to send them messages.

AS4 supports both AS2-style "push" exchanges but also offers a new message exchange feature called "pull" mode. In this mode:

- The receiver is the initiator. It receives the message from the sender in response (and as an HTTP response) to a "pull" signal message to the sender, which acts as the responder. The pull signal message is an outgoing HTTP message, which typically means no or significantly reduced configuration of firewalls, and no need to have a resolvable URL or fixed IP address.
- The receiver MSH also controls when the pull signal is sent. This means the business partner operating a pull-capable MSH can shut off its MSH outside office hours.

The following table summarizes and contrasts the "pull" and "push" mode for a "one way" message exchange.

One-Way/Push MEP	One-Way/Pull MEP
The **Sending** Message Service Handler (MSH) is the **Initiator** for the message exchange.	The **Sending** MSH is the **Responding** MSH for the message exchange
The **Receiving** MSH is the **Responding** MSH for the message exchange.	The **Receiving** MSH is the **Initiating** MSH for the message exchange.

One-Way/Push MEP	One-Way/Pull MEP
	

AS4 processors can use WS-Security Username/Password tokens to authorize pull signal messages to access specific messages.

Compared to AS2 "push" style exchanges, the newer "pull" exchange offered by AS4 has major benefits for Small and Medium Size Enterprises (SMEs) or even Micro-Enterprises. Rather than requiring an HTTP server, an AS4 MSH can be a configured to only need HTTP client capabilities, and could be run from any occasionally connected or even mobile device. AS4 is not restricted to "pull" mode though. Whether to use "push" or "pull" mode is a configuration that can be agreed with a trading partner for a particular type of exchange.

5. AS4 from a Web Services Point of View

Clearly, AS4 is not the only or the first Web Services profile. However, it is one of just a few standardized Web Services profiles designed specifically for the B2B application domain. As such, it goes, and needs to go, well beyond the various Web Services profiles developed in the Web Service Interoperability organization (WS-I, now part of OASIS). WS-I profiles are horizontal and provide a bare minimum level of interoperability. AS4 also builds on a decade of experience in large scale B2B deployments and uses of EDIINT and ebXML standards, and on work on interoperability of EDIINT and ebXML standards and product implementations. As a result, AS4 makes different choices than some other Web Services deployments in areas like synchronous replies, reliable messaging, payload compression and use of service description languages like WSDL.

Many deployments of Web Services are using synchronous SOAP request/response interactions to connect consumers of services to service providers. In a B2B context, this is not always a workable option. For example, the Cisco AS4 white paper describes a RosettaNet case study of a service where, sometimes, a synchronous response could not be produced in time and the request timed out. Their solution was to move to AS4 and use the AS4 pull mode to retrieve responses asynchronously.

The WS-I Reliable Secure Profile (RSP) defines an interoperability profile based on the WS-ReliableMessaging (WSRM) version 1.2 OASIS standard. This standard protocol is based on a concept of reliable message sequences. This is significantly more efficient for exchanging large amounts of messages reliably. However, it has few benefits for SMEs that, with many of their trading partners, only exchange messages sporadically. Instead of requiring the use of WSRM, AS4 uses **Receipt** signal messages from ebMS 3.0 as message acknowledgments. A Receipt is like an AS2 Message Disposition Notification (MDN) and, unlike the sequence acknowledgments of WSRM, supports non-repudiation of receipt. AS4 defines a "reception awareness" concept that is based on duplicate elimination and message retries based on ebMS 3.0 **MessageId** values.

AS4 also supports payload compression at the AS4 message level. The only functionality in Web Services that reduces message size is the SOAP Message Transfer Optimization Mechanism (MTOM), which obviates the need for base64 encoding binary data included in XML messages.

However, this binary data is not compressed and MTOM does not reduce the size of non-binary data (such as XML or EDI payloads, which typically can be compressed significantly). AS4 defines a simple payload compression mechanism based on GZIP which allows all exchanged data other than the AS4 SOAP envelope to be compressed.

AS4 is a payload-agnostic message exchange standard. All information needed for the MSH to process an AS4 message is encoded in, or can be inferred from, the AS4 header. AS4 does not prohibit the use of service description languages like WSDL, but in many B2B scenarios WSDL is of limited use:

- Like any B2B MSH, an AS4 processor is typically a gateway to enterprise applications. It passes on message payloads in the native (XML or EDI) formats. The AS4 processor does not include any business logic and does not process those documents. Therefore translation of payloads to objects for use in programming languages, which WSDL-based code generators are often used for, is of no use.
- In B2B, it is common to use industry-standard XML schemas that come with extensive libraries of common basic or aggregate components and provide many optional elements. While WSDL allows association of those schemas with a message, B2B communities typically define XML profiles for these B2B schemas. Like EDI message implementation guidelines, these profiles define usage constraints on these schemas that can often only be captured in non-XSD formats, such as Schematron, which WSDL does not support. Some of these constraints even differ according to trading partner.

6. Discussion

AS4 is a modern, Web Services-based B2B protocol. Like AS2, it offers secure and reliable message exchange. AS4 offers a richer message header that facilitates message routing, monitoring and tracking. In addition to AS2's "push" exchanges, AS4 also supports "pull" exchanges. Small and medium-size enterprise can use low-end, AS4-based client-only messaging endpoints to connect to their trading partners. Larger enterprises can send messages by storing them on their B2B messaging server, allowing their SME trading partners to securely pull them at a convenient time.

The GS1 eCom Technology Group (eTG) surveyed the interest among its members in new B2B technology. The survey included both questions related to AS2 enhancements and to AS4. Among the respondents:

- 81% voted in favour of having vendor products tested to be capable of interoperating using the AS4 profile.
- 84% voted in favour of GS1 recommending that AS4 products be tested for interoperability.

In discussions, eTG members indicated a need to make a business case for AS4 adoption in their industry and in their company. For reasons of investment protection, AS4 is seen mainly as a complement to AS2 for new trading partners and for SME partners, rather than as a replacement of existing AS2 partner connections.

End user companies generally use commercial off-the-shelf B2B gateway software products for AS2 connectivity. They are therefore looking to the providers of these products to add AS4 capability to their gateway products, and to enhance these products so that partner connections can be managed and monitored in a unified way, irrespective of the specific protocol (AS2 or AS4) used for a particular trading partner. Messaging products that conform to the AS2/AS4 ebHandler Conformance Clause support both AS2 and AS4. These products allow these users to leverage existing investments in software and (product-specific) skills, while adopting a new protocol.

7. References

Identifier	Description
AS2	MIME-Based Secure Peer-to-Peer Business Data Interchange Using HTTP, Applicability Statement 2 (AS2). IETF RFC, July 2005. http://www.ietf.org/rfc/rfc4130.txt
AS4	AS4 Profile of ebMS V3 Version 1.0. OASIS Committee Specification 01. April 2010. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/200707/AS4-profile.pdf
Cisco AS4	Web Services External. An AS4 Implementation at Cisco. Cisco White Paper. http://www.oagi.org/oagi/Website/Case_Studies/OAGIS_AS4Cisco-final-1.pdf
DGI AS4	AS4: Secure B2B Document Exchange Using Web Services. Drummond Group. http://www.drummondgroup.com/html-v2/as4.html
ebMS2	Message Service Specification. Version 2.0. OASIS Standard, April 2002. http://www.oasis-open.org/committees/ebxml-msg/documents/ebMS_v2_0.pdf
ebMS3	OASIS ebXML Messaging Services Version 3.0: Part 1, Core Features. OASIS Standard, 1 October 2007. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/ebms_core-3.0-spec.pdf
ebMS3P2	OASIS ebXML Messaging Services Version 3.0: Part 2, Advanced Features. OASIS Committee Specification Draft. February 2011 http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/part2/201004/csd03/ebms-v3.0-part2-csd03.odt
MTOM	SOAP Message Transmission Optimization Mechanism. W3C Recommendation. January 2005. http://www.w3.org/TR/soap12-mtom/
RSP	Reliable Secure Profile Version 1.0. WS-I Final Material. November 2010. http://www.ws-i.org/Profiles/ReliableSecureProfile-1.0-2010-11-09.html
SBDH	UN/CEFACT 2 STANDARD BUSINESS DOCUMENT HEADER. Version 1.3. June 2004. http://www.gs1.org/docs/gsmp/xml/sbdh/CEFACT_SBDH_TS_version1.3.pdf

8. Acknowledgments

The editor acknowledges the input and feedback from members of the GS1 eCOM Technology Group and from members of the OASIS ebXML Messaging Services Technical Committee.