Setting Up an AS4 System

Version 2 – 2016-11-15

Table of contents

1 Introduction

This document is aimed at users that need to set up the AS4 protocol in their organisations and need a basic understanding of how B2B communication using AS4 fits in IT environments. It explains, at a high level, the concepts of communication using the AS4 protocol [AS4], describes the communication layer in an AS4 data exchange and explains the concept of a B2B Gateway. Some general requirements on B2B gateways are presented and the benefits of using a B2B gateway are explained. Finally, a sample deployment scenario is presented.

The purpose of this document is to provide general high-level information on B2B document exchange and its position in the enterprise IT landscape, and some AS4-specific information. Furthermore, it describes key steps that organisations need to take to implement AS4 in their organisation.

For AS4, the concept of Processing Modes is introduced and the various parameters that need to be configured to use AS4. For partner communication, three cases will be described: initial configuration of an AS4 gateway for communication with a partner; configuring a specific service for use with a partner; and updating existing partner configurations.

This document is informative only. It may be used as a guideline or good practice and provides some example setups, but does not mandate a particular way of implementing AS4. Parts of this document cover generic B2B communication topics that are not tied to any distinguishing feature of the AS4 protocol.

The audience for this document are IT managers, B2B integration project teams and IT infrastructure managements that are starting to implement AS4 in their organisations, with a focus on Transmission System Operators for gas that will implement the ENTSOG AS4 Usage Profile for TSO [AS4TSO]. *It does not cover the AS4 standard or the ENTSOG usage profile in any detail.*

2 AS4 Communication Concept

2.1 Data Exchange Concepts

The AS4 protocol supports the concept of *document-based* data exchange. This is a model where enterprises in a market collaborate and synchronise their business processes at specific agreed process steps. The synchronisation involves the exchange of information between enterprises as *business documents*. Documents are encoded in a structured format that is standardised in the sector (like EASEE-gas EDIG@S-XML) or otherwise agreed. Business documents are exchanged using B2B communication protocols (like AS4) using agreed implementation guidelines. The ENTSOG AS4 Usage Profile is an example of such an implementation guideline for AS4. Because of the requirements in the business processes it is needed to assure the integrity and identify the sender of the document, therefore security measures have to be taken and implemented.

In document-based data exchange, the exchanged information is produced and consumed by business applications. This is a key difference with paper-based communication, electronic mail or using Web portals, all of which require human intervention.

2.2 Data Exchange Layers

In data exchange, a distinction can be made between the business operational view (the *what*) and the IT functional service view (the *how*). Market rules and regulations determine the business process and activities, from which in turn the structure and content of the information to be exchanged follows. The Information Technology view is concerned with the exchange of information across a public or (virtual) private computer network using message exchange protocols. These layers can be visualised as in Figure 1.

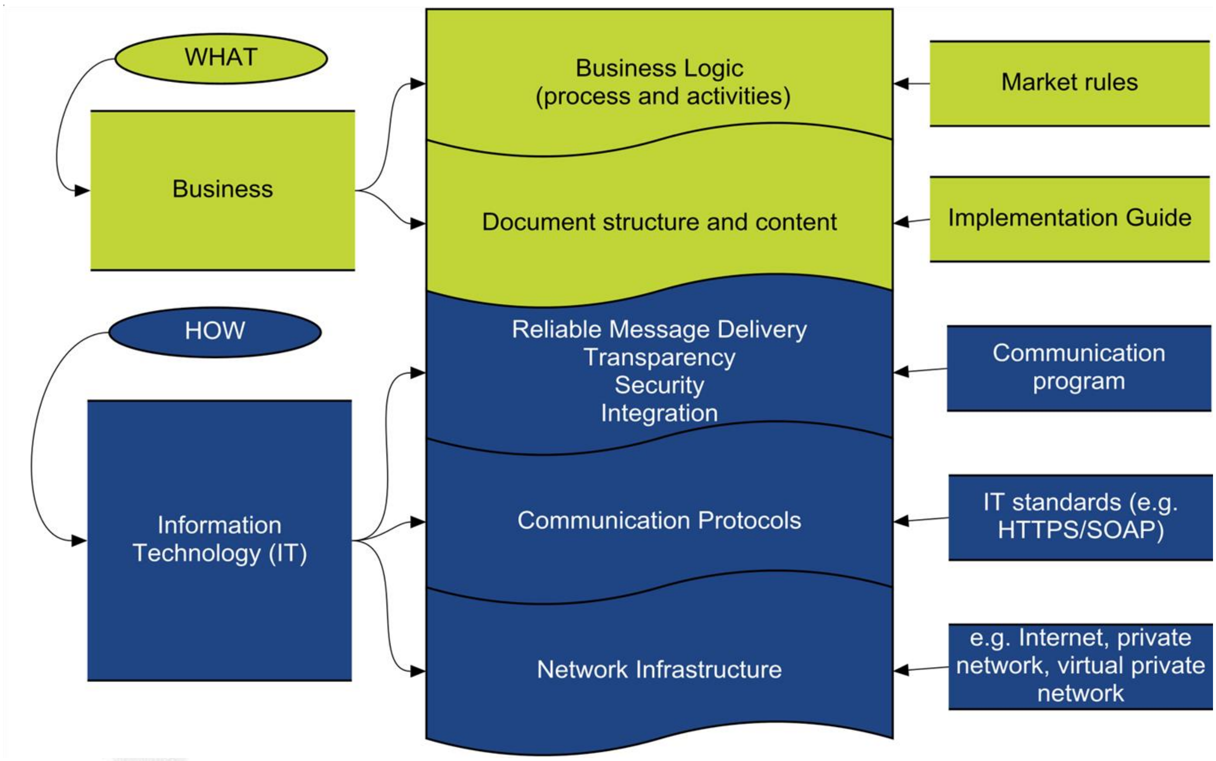

WHAT
Business
Business Logic (process and activities)
Document structure and content
Market rules
Implementation Guide
HOW
Information Technology (IT)
Reliable Message Delivery
Transparency
Security
Integration
Communication Protocols
Network Infrastructure
Communication program
IT standards (e.g. HTTPS/SOAP)
e.g. Internet, private network, virtual private network

Figure 1 Data Exchange

2.3 B2B Gateway Concept

It is a common practice in data exchange to not directly connect one’s business applications to business applications of one’s counterparties, but to use architectural components called *B2B Gateways*, which are responsible for document-based B2B data exchange. A B2B gateway serves as an intermediary between an enterprise and its communication partners. The concept of a B2B Gateway is sufficiently common that a class of commercial off-the-shelf software products and related services exists that can be used to implement such a gateway in general and communication protocols like AS4.

A B2B Gateway has an enterprise interface and a trading partner interface and supports bidirectional communication. On the enterprise side, the gateway behaves as an application in the enterprise IT landscape and should adhere to corporate standards and support to the enterprise’s *private* processes. On the partner side, it functions as the partner interface and should conform to the partner community standards and its *public* processes. Whereas the enterprise side is under the control of the organisation and closed to (possibly malicious) third parties, the partner side is not. It involves the use of third party infrastructure and public networks and therefore security and reliability require special attention.

The processing of documents by B2B gateways and Enterprise Service Bus (ESB) or other middleware (if used) is typically not immediately visible to the end-user. The end-user may therefore still have the impression that communication is directly between applications. This is visualised in Figure 2.

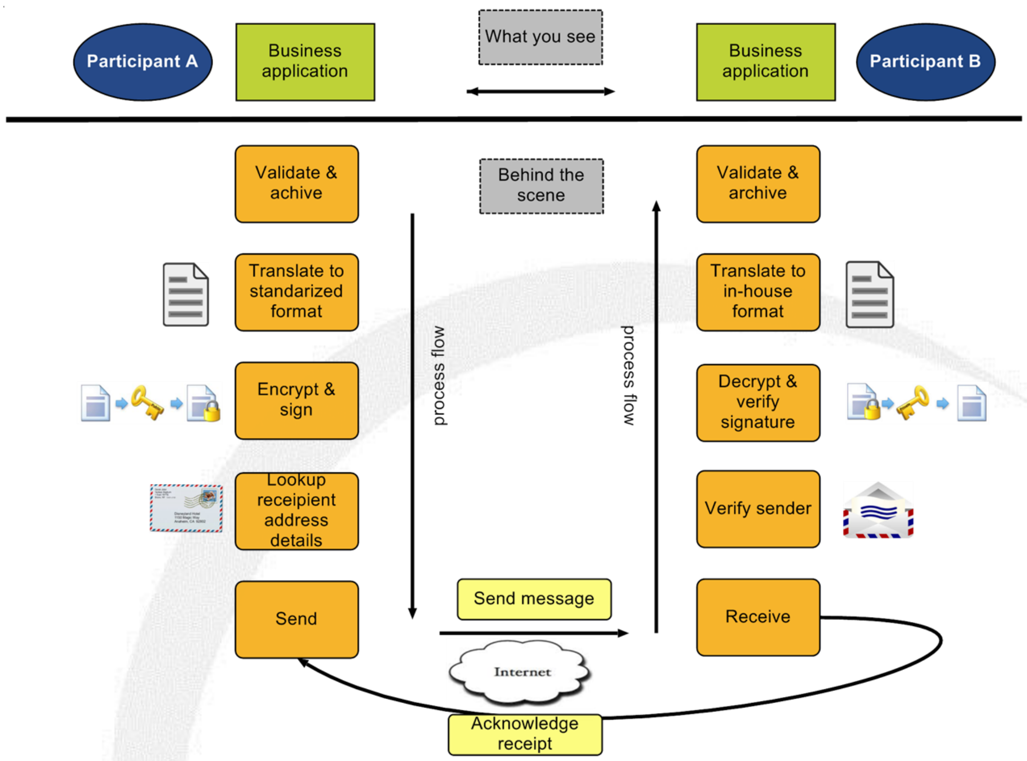


Figure 2 What the User Sees

On the enterprise side The B2B Gateway can be connected directly to business applications using a variety of mechanisms including enterprise communication protocols like FTP(File Transfer Protocol) , messaging APIs like JMS, shared file systems or databases. However, enterprises are increasingly adopting service-oriented concepts and integrating business applications using an *Enterprise Service Bus* (ESB). In such a model, B2B communication is exposed by the B2B gateway to the ESB, just like business applications expose business services, and the gateway and applications are not directly connected.

2.4 B2B Gateway Requirements

A B2B gateway must support fully automatic processing. This means it must support the exchange of structured business content as well as metadata to express the purpose and requested processing.

A B2B gateway must also support secure and reliable communication, by protecting the integrity and confidentiality of content, and to authenticate the identity of sender and a receiver and to support non-repudiation of origin and receipt.

B2B Communication should be based on open standards, and independent of specific vendor products. Transmission System Operators should be able to procure solutions in a competitive environment. AS4 is such an open standard and is implemented by a variety of solutions. The ENTSOG AS4 Usage Profile provides additional detailed guidance and interoperability; it limits the configuration options and usage to a defined set.

2.5 *Benefits of a B2B Gateway*

A B2B gateway decouples the IT systems of a party and its counterparty and therefore supports interoperability at the business process layer amongst organisations that use IT systems that may be very different. The decoupling covers a range of aspects:

- At the *network (security) layer*, the gateway is connected externally (to partner gateways) and internally (to enterprise IT), obviating the need for direct network connectivity between enterprise systems and partner systems. This simplifies configuration and management of partner connectivity. Only the gateway needs to know about IP addresses, ports and transport layer security configuration for specific partners.
- At the *application layer*, the gateway intermediates between internal systems and trading partners. Trading partner do not need to know which business application is responsible for handling specific messages, as the gateway (or ESB) is responsible for routing messages appropriately. AS4 support such routing by providing rich metadata.
- At the *communication protocol* layer, the gateway is responsible for selecting the communication protocol to use for a partner and message type. Communication may switch from older protocols to newer (e.g. from AS2 to AS4) without any the need for reconfiguring business applications. Similarly, an enterprise can drastically change its internal integration (e.g. introducing an ESB or switching from one type of middleware to another) without impacting its trading partner.
- At the *business content* layer, some B2B gateway products support the mapping of document formats, or version of formats. For example, they may transform XML to in-house formats or transform one type of XML to another. (In some ESBs, this transformation may itself be a service that is invoked from the ESB rather than the gateway).

2.6 *Sample AS4 Gateway System Perspective*

A sample deployment scenario for a B2B Gateway is displayed in Figure 3 Sample System Perspective. This diagram illustrates how an AS4 gateway may be implemented and may fit in an enterprise IT landscape, not precluding other possible alternative architectural options.

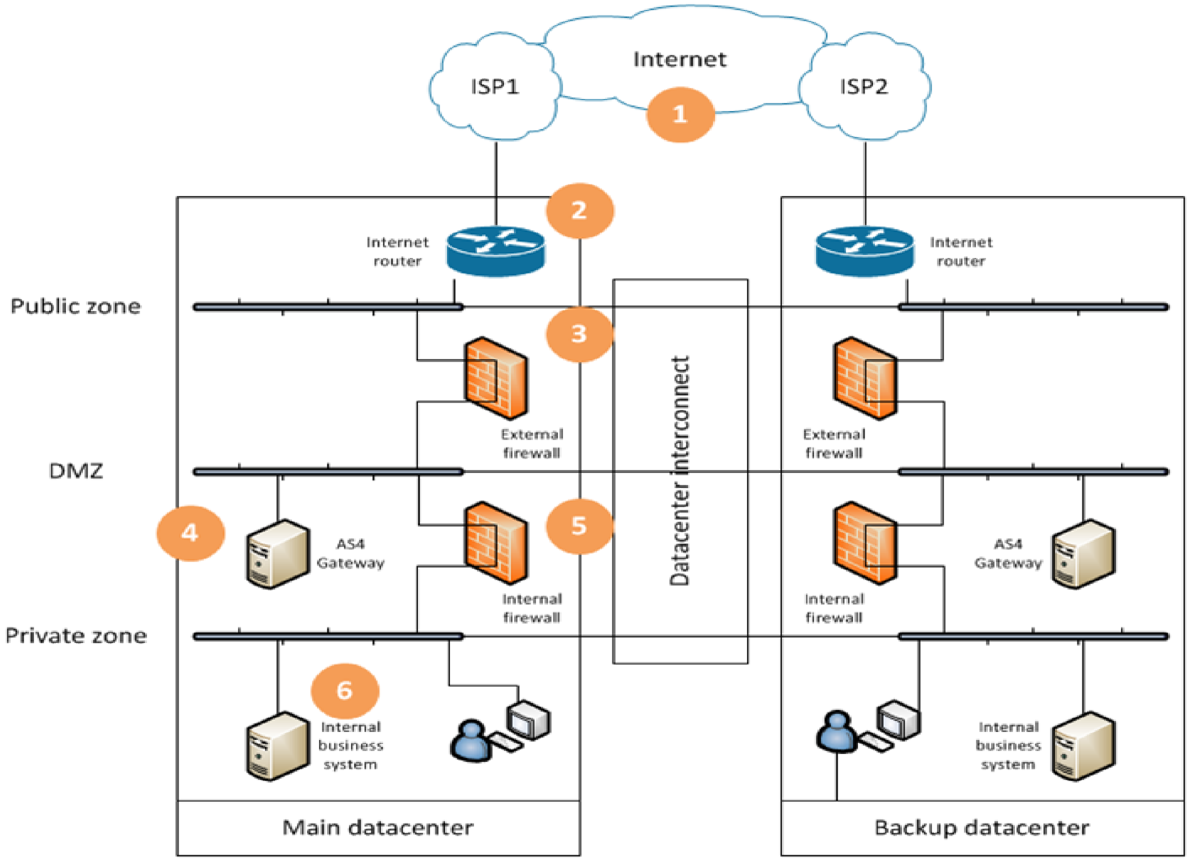


Figure 3 Sample System Perspective

The AS4 gateway, in this sample scenario, is separated from the Internet by an External Firewall, which is configured to allow communication with communication partners, for which the IP addresses are known. A separate firewall separates the AS4 Gateway from the organisation's internal business systems (possible connected using an ESB or other middleware) and end user computers. No direct communication is possible from external systems to the internal systems.

The diagram also shows the use of a backup data centre, which mirrors the main datacentre. It has a separate Internet connection and an AS4 gateway that can take over from the main gateway for failover. Of course measures should be taken towards the internal business systems to synchronise between main and backup datacentres in order to guarantee business continuity and no loss of data. In case of a switchover, the partners should not need to change anything in their systems. Established mechanisms exist to handle such events. They are not dependent on AS4 or B2B messaging in general, and will therefore not be elaborated on in this document. The approach illustrated in this diagram is a good practice of a so-called active-passive cluster configuration.

Another option is to deploy multiple gateway server instances in parallel in a so-called active-active cluster configuration. The server address communicated to communication partners is the address of a load balancer that forwards incoming messages to the various server nodes. Outgoing messages will still be sent directly from the cluster nodes to communication partners. In addition to providing continuity in case of failure of some cluster node (as in the active-passive model), this allows the cluster to scale out to process message volumes that are larger than a single AS4 gateway instance could process.

When deploying a gateway product in a cluster, similar consideration apply to supporting infrastructure such as databases and file systems used by the gateway.

3 Deploying AS4

When implementing AS4, a number of steps need to be taken; some in sequence (due to dependencies) but some of these steps can take place in parallel. Some are to be taken once, and some need to be revisited if certain events or changes occur.

3.1 Selecting an AS4 Gateway

To implement AS4, an organisation needs to deploy an AS4 gateway product. As AS4 is an open standard [AS4], organisations are in principle free to choose any conformant product that is interoperable with other available AS4 products used in the community and that otherwise meets the business or technical requirements of the organisation. Reasons for preferring one product over the other may include compatibility with other IT applications or frameworks, established vendor relations or commercial considerations and will lead to different choice in different organisations.

To support the practical implementation in the gas community, ENTSOG publishes a Usage Profile of AS4 on its public Internet site [AS4TSO] that reduces the feature set to be implemented by the AS4 product and provides interoperability guidelines. When contacting potential suppliers of AS4 solutions, implementers are strongly recommended to ask the vendor to provide a formal assurance that their solution fully and correctly implements this profile and can demonstrate experience in using its product interoperably with other vendor products. Some vendors participated in the ENTSOG interoperability proof-of-concept in 2014 and successfully demonstrated interoperability [AS4POC], and since then other vendors have implemented the profile as well. Some AS4 products have been successfully deployed by TSOs and are used in production.

It should be noted that some AS4 products (including products marketed to companies in the gas market in Europe) do not (yet) support all the features mandated in the Usage Profile or do not support them interoperably. Users are recommended to obtain information from their (prospective) suppliers regarding (non)compliance and/or (lack of) interoperability with other AS4 solutions for the ENTSOG AS4 Usage Profile.

Many organisations already deploy a B2B gateway for AS2 or other protocols. As many B2B gateway vendors support multiple B2B protocols in a single gateway product, in some cases an upgrade to a more recent version of the product, or deploying some optional module, may be all it takes to be enable an AS4 feature.

3.2 Initial Deployment

The initial deployment of an AS4 gateway consists of the installation of the AS4 gateway software, internal integration (within the enterprise) and preparations for external integration (to the communication partners). Installation of an AS4 gateway is done in a particular environment (single server or cluster) and involves some initial software configuration. For example, the gateway may require a database for which the connection properties need to be set.

The result of the initial deployment is an AS4 gateway to which message payload and metadata can be submitted, which can deliver received payloads and metadata, and which has a basic configuration (known server URL, IP address, certificates) to enable communication with partners.

Note that this initial installation and configuration step typically needs to be repeated for each environment the software is deployed in (e.g. test, pre-production, production).

3.2.1 Internal Integration

On the *internal integration* side (integration with business applications and/or middleware within the enterprise), any AS4 product offers interfaces to *submit* messages produced by enterprise applications to be sent to B2B partners and to *deliver* messages received from B2B partners to an internal consumer. The AS4 standard defines abstract operations for submitting and delivery, but the actual implementation is product-dependent.

B2B products often offer multiple interfaces, such as shared folders, APIs for certain programming languages, JMS or other enterprise messaging systems, FTP or other transport protocols, SOAP etc. Which of these an organisation should use typically depends on the approach to enterprise integration in an organisation. Many organisations adopt Enterprise Service Bus (ESB) technology to connect their business applications. In these organisations, the AS4 gateway should be connected to the ESB and use ESB services, rather than be connected to business applications directly, though the latter is an option.

When submitting payloads to be sent, a B2B gateway typically needs some metadata to know how to process the data, in particular minimally the intended recipient. Using the party identifier of the recipient, the endpoint of the recipient and other relevant parameters are retrieved from configuration so the message can be sent. Compared with other protocols like AS2, more metadata may be required for AS4 beyond the recipient party identifier, such as the *Service* to be addressed. The Usage Profile describes this and specifies how this metadata can be extracted (or inferred, using lookup tables) from EDIG@S content. To reuse unmodified enterprise software applications, this metadata handling should be done in an ESB or other middleware service. This metadata allows the AS4 gateway to determine the processing mode to apply to the message. For more information on the concept of "processing modes" in AS4, see section 3.3.

3.2.2 External Integration

On the *external integration* side (integration with partners), AS4 gateway products may terminate AS4 communication from the public zone directly (as in Figure 3), or use a separate Web Server or other networking software or hardware (such as an XML Appliance). To be accessible, the AS4 gateway must be resolvable via the Internet Domain Name Service (DNS) using a static IP address. While DNS configuration changes are simple changes, they should be addressed early in the project as in large organisations they may involve different departments and change processes can take time.

Like other B2B protocols, AS4 and the ENTSOG Usage Profile rely on X.509 Digital Certificates for message-layer sender and receiver authentication, non-repudiation and confidentiality and for server (and optionally client) authentication at transport layer. The Usage Profile defines requirements on certificates to be used but does not currently mandate a specific Certificate Authority. Many TSOs and partners use certificates issued by EASEE-gas for use with AS2. In principle, these certificates can also be used with AS4 and will be readily accepted as many organisations are used to working with EASEE-gas certificates. Organisations that want to deploy certificates from other Certificate Authorities should be aware that their counterparties may ask them to provide evidence that these authorities are trustworthy and comply with the requirements defined in the Usage Profile section 2.3.4.5. Their counterparties may find it difficult to accept certificates from authorities in case no such evidence is provided or in case any evidence provided is difficult to verify. The latter is the case if the CA is a local certificate authority from a member state that is unknown outside the country and only publishes its certificate policy and other documentation in a local language. Organisations should also be aware that certificates issued by other Certificate Authorities may have various interoperability issues.

3.3 Processing Modes

In AS4, a “Processing Mode (or P-Mode) is a collection of parameters that determine how messages are exchanged between a pair of MSHs with respect to quality of service, transmission mode, and error handling.

A P-Mode may be viewed and used in two ways:

- It is an agreement between two parties as to how messages must be processed, on both the sending and receiving sides. Both MSHs must be able to associate the same P-Mode with a message, as this is necessary for consistent processing (of security, reliability, message exchange pattern, etc.) end-to-end.
- It is configuration data for a Sending MSH, as well as for a Receiving MSH.

Several P-Mode instances may be used to govern the processing of different messages between two MSHs. A P-Mode is usually associated with a class of messages that is identified by some common header values – e.g. the class of messages sharing same values for *eb:Service*, *eb:Action*, and *eb:AgreementRef*.

More abstractly, a P-Mode is said to be *deployed* on an MSH when it is governing the processing of an associated class of messages on the MSH.” [EBMS3].

The process of configuring an AS4 gateway for communication between parties therefore involves the configuration of P-Modes for those parties. This sub-section explains the AS4 concept of P-Modes and the various parameters. The next sub-section will explain how this fits into implementing ENTSOG AS4, and in which situations this configuration needs to be reviewed and possibly updated.

Processing Mode parameters can be assigned to one of the following groups:

- The Sender of the message.
- The Receiver of the message.

- The Business Process.
- The Sender-Receiver pair.
- Use of specific AS4 features, and constraints on the use of those features.

The ENTSOG Usage Profile provides detailed additional guidance on how parameters for the various P-Modes are to be set. The following sub-sections describes these parameters and their values in more detail.

Note that products have their own interfaces and data formats for storing these parameters. The information in this section therefore must be mapped to the (product-specific) configuration mechanisms.

3.3.1 Party-related Parameters

AS4 encodes the sender and receiver party and party type identifiers in the message and has P-Mode parameters to specify these values. The ENTSOG AS4 profile requires the party ID values to be EIC codes and defines a fixed format for the Party type attribute.

Example of the use of these values in an AS4 header:

```
<eb3:PartyId type="http://www.entsoe.eu/eic-codes/eic-party-codes-x">21X-EU-A-X0A0Y-Z</eb3:PartyId>
```

For every party, the signing certificate must be recorded and shared. This certificate is used to sign AS4 messages for the party as a sender, and to sign receipts for the party as a receiver.

For every party that receives a user message (i.e. a message carrying some EDIG@S XML or other payload), in addition to a signing certificate an encryption certificate is needed. The sender uses this certificate to encrypt the message such that only the receiver is able to decrypt the message.

For each party that receives a message, the URL of the AS4 gateway must be specified and shared.[1] This URL starts with the https:// prefix, because AS4 uses HTTP and ENTSOG AS4 requires TLS.

Within a community of companies exchanging gas business messages, parties act in particular roles. These roles constrain the types of documents that can be exchanged between parties. See below, section 3.3.2, for more related information.

Over time, a party uses one set of certificates during one time period and another in another time period. Therefore each party is associated, not with a signing certificate/encryption

[1] Note that, in theory, ebMS3 and AS4 allow party identifiers, certificates and URLs to be specified per P-Mode. For example, a party might use one certificate when sending one type of message to one party and another certificate to send one to another party. Or a receiver might receive AS4 messages of a particular type and/or from a particular sender on one URL and other messages on another URL. The Usage Profile currently does not preclude such more complex configurations and for simplicity these parameter values should be fixed for all P-Modes that use them.

certificate pair, but with possibly multiple such pairs, each of which has an associated validity period. See below, section 3.3.3, for discussion.

3.3.2 Business Process-related Parameters

In AS4, the message reflects the business process, or service, that it relates to, by including *Service* and *Action* headers in the message. For ENTSOG AS4, the following cases can be distinguished:

- The Test service defined in section 2.3.6 of the ENTSOG profile. This should be the first service to configure when implementing ENTSOG AS4. This service uses a fixed combination of *Service* and *Action* values defined in the ebMS3 standard. More information on configuring the P-Mode for the test service is given below, in section 3.4.1. The test service uses the ebMS3 default roles.
- Gas business services as defined in EDIGAS. The ENTSOG AS4 profile describes the values to use for *Service* (in section 2.3.1.2.1), *Action* (in section 2.3.1.2.2) and initiator and responder *Role* (in section 2.3.1.2.3). Specifically, it states that the values are to be taken from the ENTSOG AS4 Mapping Table [AS4MT]. The values of this table that are relevant to a party are those in which the sender or receiver *Role* is (one of) the role(s) of the company. More information on configuring these services is given below, in section 3.4.2. In the table, the *Action* is constrained to be the AS4 default action. The *Service* reflects the process area. The *Role* reflects the roles of the parties in the process.
- A future version of the ENTSOG profile will support ebCore Agreement Update [AU]. That protocol defines its own *Service* and *Action* values. This allows these update messages to be routed to the service responsible for managing the configuration of the AS4 service. More information on configuring these services is given below, in section 3.4.3.

The various combinations of *Service*, *Action* and *Role*, and directionality of these messages, require separate P-Modes to be configured.

3.3.3 Sender, Receiver and Agreement

Some P-Mode parameters relate to both the Sender and the Receiver. The only such parameter used in the ENTSOG profile is *AgreementRef*, which identifies a particular agreement between those parties. In the ENTSOG profile, this agreement is just an identifier of a particular set of P-Modes, valid in a particular validity period. It is configured in the PMode.Agreement parameter. The ENTSOG AS4 Usage Profile defines a string format convention that combines the party identifiers and a version number. Example of the use of this value in an AS4 header:

```
<eb3:AgreementRef>http://entsog.eu/communication/agreements/21X-EU-A-X0A0Y-Z/21X-EU-B-P0Q0R-S/3
</eb3:AgreementRef>
```

Section 3.3.1 mentioned that parties are identified and associated with particular sets of certificates. The ENTSOG AS4 Usage Profile requires there to be a functional relation between agreements and a pairs of Sender/Receiver certificate sets. That is, each agreement is linked to a specific fixed specific pair of signing/encryption certificates for the sender and a fixed specific pair of signing/encryption certificates for the receiver. Note that in an agreement, a party may be a sender for one type of message and a receiver for another. The agreement identifier indicates which set of values applies to message. The validity period of an agreement is constrained by the validity period of the certificates associated with it.

For example, agreement version 1 between P1 and P2 could valid from 1st of June 2016 to 1st of June 2019 and version 2 could be valid from 1st of May 2019 to 1st of May 2022. Agreements 1 and 2 could then be exactly the same in all parameter values except for the certificates used. In May 2019 both the version 1 and version 2 agreement are valid. As the agreement identifier is a header element, each message unambiguously indicates which certificates it is expected to use.[2] Agreement 1 P-Modes uses one set of certificates, which must be valid in the validity period of the agreement, whereas Agreement 2 use another set of certificates that are valid in the validity period for Agreement 2.

3.3.4 Use of ebMS3/AS4 Features

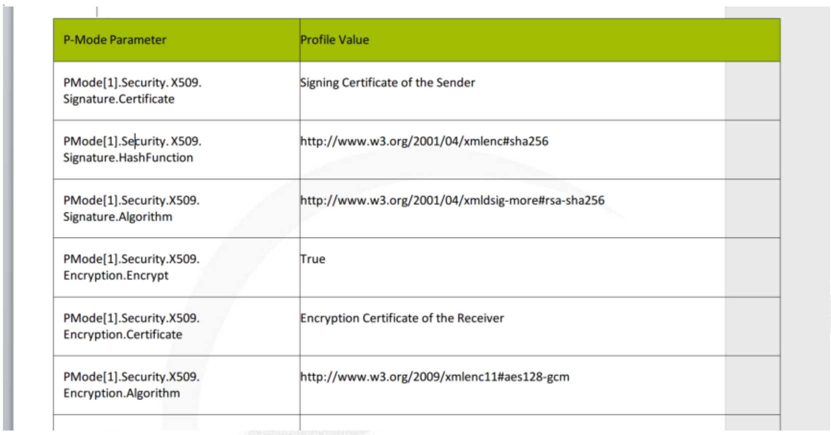
The ebMS3 standard on which AS4 is based is a highly configurable protocol with many technical features and options. Some solutions aim to implement a large subset of ebMS3 and therefore allow the user to select from a broad range of options, rather than constraining their product to a specific profile. In practice, most of these options are not used, because:

1. AS4 already profile the use of ebMS3. For example, the version of SOAP to be used is always SOAP 1.2, even though ebMS3 allows a choice of SOAP 1.1 or 1.2. Most ebMS3 products are focused on AS4 rather than on ebMS3 in general.

2. The ENTSOG profile further narrows down the choices of AS4. For example, it specifies that all messages are secured using WS-Security (in ebMS3, this is optional); and moreover, that all messages are encrypted using XML Encryption; and moreover, that the AES-128-GCM algorithm is used for that encryption.

A succinct overview of AS4 P-Mode parameters for the ENTSOG AS4 is provided in chapter 4 of the ENTSOG AS4 Usage Profile. This table is a summary of the Usage Profile. An excerpt of the table is in Figure 4.

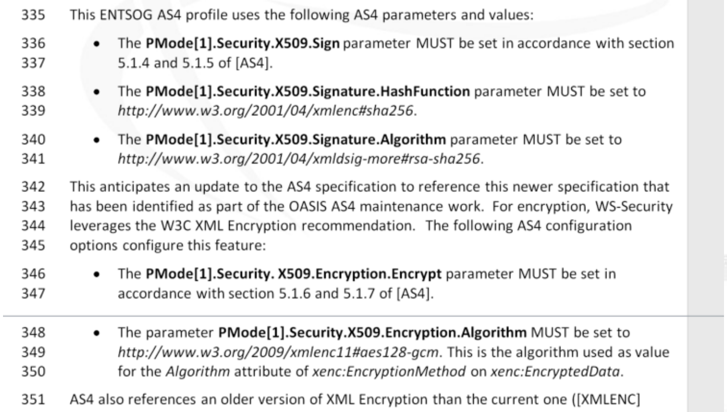
[2] Note that this requires ENTSOG AS4 compliant solutions to use this header in the selection of the P-Mode to use when sending or receiving a message. Implementers are encouraged to check with any (prospective) supplier of their AS4 solution that they meet this requirement.



P-Mode Parameter	Profile Value
PMode[1].Security. X509. Signature.Certificate	Signing Certificate of the Sender
PMode[1].Security. X509. Signature.HashFunction	http://www.w3.org/2001/04/xmlenc#sha256
PMode[1].Security.X509. Signature.Algorithm	http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
PMode[1].Security.X509. Encryption.Encrypt	True
PMode[1].Security.X509. Encryption.Certificate	Encryption Certificate of the Receiver
PMode[1].Security.X509. Encryption.Algorithm	http://www.w3.org/2009/xmlenc11#aes128-gcm

Figure 4 Part of the P-Mode Parameter Table in ENTSOG Usage Profile

Other sections of the profile provide additional explanation for these parameters. For examples, the following excerpt of the security section describes this in textual form, which the P-Mode table summarizes.



This ENTSOG AS4 profile uses the following AS4 parameters and values:

- The **PMode[1].Security.X509.Sign** parameter MUST be set in accordance with section 5.1.4 and 5.1.5 of [AS4].
- The **PMode[1].Security.X509.Signature.HashFunction** parameter MUST be set to *http://www.w3.org/2001/04/xmlenc#sha256.*
- The **PMode[1].Security.X509.Signature.Algorithm** parameter MUST be set to *http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.*

This anticipates an update to the AS4 specification to reference this newer specification that has been identified as part of the OASIS AS4 maintenance work. For encryption, WS-Security leverages the W3C XML Encryption recommendation. The following AS4 configuration options configure this feature:

- The **PMode[1].Security. X509.Encryption.Encrypt** parameter MUST be set in accordance with section 5.1.6 and 5.1.7 of [AS4].
- The parameter **PMode[1].Security.X509.Encryption.Algorithm** MUST be set to *http://www.w3.org/2009/xmlenc11#aes128-gcm*. This is the algorithm used as value for the *Algorithm* attribute of *xenc:EncryptionMethod* on *xenc:EncryptedData*.

AS4 also references an older version of XML Encryption than the current one ([XMLENC]

Figure 5 Excerpt of the Usage Profiling

3.4 How to set up a Connection

3.4.1 Initial Configuration of a Communication Partner

After the initial deployment of the AS4 system at a company, the next step is to connect the AS4 gateway to the company’s communication partners. This involves exchanging essential configuration parameter sets with the partners, such as: the organisation’s party identifier, certificates, endpoint URL, and inbound and outbound IP addresses (or address ranges), and the same parameter set for the counterparty.

Firewalls must be configured to allow incoming connections from communication partners. In some organisations, outgoing connections (from all AS4 cluster nodes) must also be explicitly allowed. While, like DNS changes, firewall configuration changes are simple changes, they should be addressed early in the project as in large organisations they often involve different departments and service management change processes can be time-consuming.

When using AS4, communication with a partner is configured using P-Modes. As first mentioned above, under 3.3.3, several P-Modes can be grouped under an “agreement”. Section 2.3.2 of the ENTSOG A4 profile defines a convention for agreement identifiers that includes the party identifiers, sorted alphabetically, and a version number. By convention, the version number of the first agreement with a partner is “1”.

Before using the established configuration for any “real” (gas business) service, it is important to test it is configured properly. Taking advantage of its richer metadata (*Service* and *Action* headers), AS4 has a useful mechanism that allows partners to determine if their AS4 gateways can successfully exchange messages: the *test* service. This service is defined in section 5.2.2 of [AS4] and further described in section 2.3.6 of the ENTSOG Usage Profile for TSOs [AS4TSO]. The first P-Modes to configure for a new partner therefore relate to the use of this service.

A (hypothetical) P-Mode for a test message from the first party in the ENTSOG EIC code table (as published in September 2016 at http://www.entsog.eu/eic-codes/eic-party-codes-x), which has identifier 21X-AT-A-A0A0A-T to the second, which has EIC value 21X-AT-B-A0A0A-K, is provided in Table 1 below.

Parameter	**Value**
PMode.Agreement	http://entsog.eu/communication/agreements/21X-AT-A-A0A0A-T /21X-AT-B-A0A0A-K /1
PMode.Initiator.Party	21X-AT-A-A0A0A-T
PMode.Initiator.Party Type	http://www.entsoe.eu/eic-codes/eic-party-codes-x
PMode.Initiator.Role	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/initiator
PMode.Responder.Party	21X-AT-B-A0A0A-K
PMode.Responder.Party Type	http://www.entsoe.eu/eic-codes/eic-party-codes-x
PMode.Responder.Role	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/responder
PMode[1].Protocol.Address	https://hypothetical.url.at.example.com/as4
PMode[1].BusinessInfo.Service (No Service Type for this Service)	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service
PMode[1].BusinessInfo.Action	http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test
PMode[1].Security. X509.	..

Signature.Certificate	
PMode[1].Security.X509.Encryption.Certificate	..

Table 1 P-Mode for Test Service from 21X-AT-A-A0A0A-T to 21X-AT-B-A0A0A-K

Note that this P-Mode only configures test messages from 21X-AT-A-A0A0A-T to 21X-AT-B-A0A0A-K. A separate P-Mode is needed to configure test messages in the reverse direction.

Support of the test feature is mandated in section 2.3.6 of the ENTSOG Usage Profile for TSOs [AS4TSO]. If a party is able to successfully send an AS4 *test* message to a counterparty and receive a corresponding AS4 receipt, and if the counterparty is similarly able to access the *test* service of the party, both party and counterparty know their AS4 configuration (party identifiers, endpoints, certificates) and network configurations (firewalls) are consistent and fully functional. In AS4, the *test* service is a service like any service except that AS4 *test* messages are never delivered to any business service but are consumed internally in the AS4 gateway. Therefore implementers can assume that no data is ever accidentally delivered to any business application in any environment.

Note that if an organisation deploys multiple AS4 Gateways for different services behind an XML routing appliance (or similar component), using the *test* service only tests connectivity to the gateway that handles the test service. This may be acceptable if all gateways are synchronised to use the same certificate set.

If it is necessary to test connectivity to all such gateways, another header field could be configured for routing at the appliance (such as *AgreementRef*) to route to a specific gateway, as there is only one test service. Alternatively, it may be possible to configure the appliance to load-balance *test* service messages over all AS4 Gateways. The sender can then send a batch of messages to the *test* services to test that all gateways are functioning correctly, assuming eventually all gateways receive and reply to at least one test message.

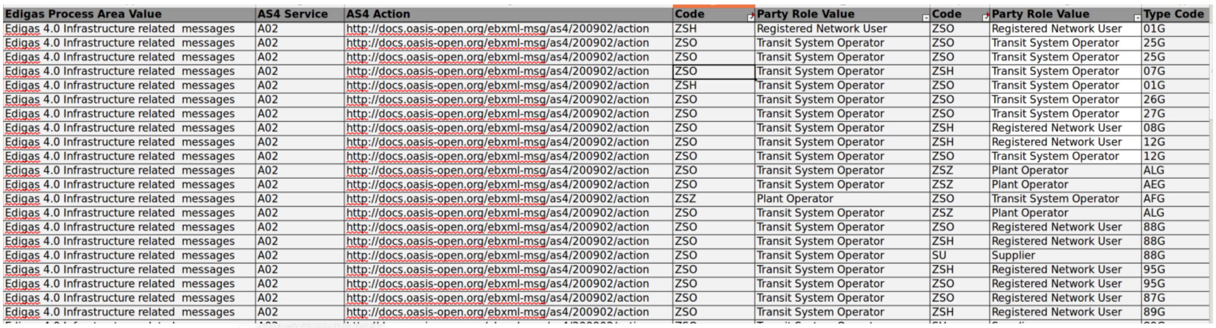
It should be noted that if the Communication Partner has different AS4 Gateways for different environments (e.g. test, pre-production, production) this step needs to be done for each environment that needs to be connected with.

If more than one agreement is in place between two parties (as discussed in section 3.4.3, below), test service P-Modes are needed for each agreement.

3.4.2 Configuring a Partner for a Business Service

Once AS4 communication is successfully established with the corresponding environment of the counterparty using the *test* service, the AS4 gateway configuration can be extended to support additional services beyond the *test* service. The configuration for other services will be the same as the *test* configuration except for *Service, Action* and *Role* values. Unlike the *test* service, payload data will be delivered to enterprise service consumers of the counterparty rather than being consumed within counterparty's AS4 gateway.

As described in the ENTSOG Usage Profile [AS4TSO], information on the actual values to be used for services supporting specific business processes is provided by ENTSOG for the business processes for which it provides Business Requirements Specifications (BRSs) in the AS4 mapping table [AS4MT]. Table 2 shows a subset of the content in this table.



Edigas Process Area Value	AS4 Service	AS4 Action	Code	Party Role Value	Code	Party Role Value	Type Code
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSH	Registered Network User	ZSO	Registered Network User	01G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Transit System Operator	25G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Transit System Operator	25G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Transit System Operator	07G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSH	Transit System Operator	ZSO	Transit System Operator	01G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Transit System Operator	26G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Transit System Operator	27G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Registered Network User	08G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Registered Network User	12G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Transit System Operator	12G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSZ	Plant Operator	ALG
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSZ	Plant Operator	AEG
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSZ	Plant Operator	ZSO	Transit System Operator	AFG
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSZ	Plant Operator	ALG
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Registered Network User	88G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Registered Network User	88G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	SU	Supplier	88G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Registered Network User	95G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Registered Network User	95G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSO	Registered Network User	87G
Edigas 4.0 Infrastructure related messages	A02	http://docs.oasis-open.org/ebxml-msg/as4/200902/action	ZSO	Transit System Operator	ZSH	Registered Network User	89G

Table 2 Subset of ENTSOG AS4 Mapping Table

In the shown part of the table, a party that is a Transit System Operator (ZSO) may send messages in the A02 service to, and receive A02 messages from, other ZSO parties, as well as to/from SU, ZSH and ZSZ parties. For any particular ZSO, such as 21X-AT-A-A0A0A-T and 21X-AT-B-A0A0A-K, all or a subset of these values apply. Each of these rows relates to an AS4 P-Mode, if the exchange is a document-based exchange. Most of the parameter and values in these P-Modes would be identical to the settings for the test service shown in Table 1. For the exchange from ZSO to ZSO, Table 3 shows the five parameters that have different values from the ones in Table 1 are displayed.

Parameter	Value
PMode.Initiator.Role	ZSO
PMode.Responder.Role	ZSO
PMode[1].BusinessInfo.Service	A02
PMode[1].BusinessInfo.Service Type	http://edigas.org/service
PMode[1].BusinessInfo.Action	http://docs.oasis-open.org/ebxml-msg/as4/200902/action

Table 3 P-Mode for an EDIG@S Business Message (only differences with Table 2 shown)

As noted before, if the Communication Partner has different AS4 Gateways for different environments (e.g. test, pre-production, production) in which the Service is implemented, there are likely to be different configuration for the various environments, in particular the endpoint address.

3.4.3 Updating Configurations and Certificates

The lifecycle of all service configurations needs to be managed, i.e. new services will be provided for existing counterparties, and new counterparties may emerge. Furthermore, it may be that an organisation changes one of its technical configuration parameters for AS4, such as a server URL, a reliable messaging parameter, or an IP address. These changes need to be bilaterally agreed and coordinated.

A specific case is the update of X.509 certificates, because they have a fixed lifetime and need to be replaced once they expire. The EASEE-gas community has developed an approach to certificate replacement that assumes a coordinated change of all certificates in the community. In the future version, the ENTSOG AS4 profile will help improve on this and allow updates to take place at any time, support a phased transition, use AS4 to exchange the update information and allow the update to be automated (or as automated or manual as parties want it to be).

This improved certificate exchange process is based on ebCore Agreement Update [AU], a recently developed protocol to exchange update information using AS4 messages. When used with AS4, this protocol allows the creation of a new set of P-Modes, for a new agreement, that relate one-to-one to P-Modes in the previous agreement, and are identical except for the updates applied.

The Agreement Update protocol consists of three pre-defined messages to transmit, accept or reject updates. To allow an agreement to update itself, it must support these three messages. Since both parties in an agreement may initiate an update request, this means six P-Modes must be specified for every agreement.

1. Update request from P1 to P2
2. Update acceptance by P2
3. Update rejection by P2
4. Update request from P2 to P1
5. Update acceptance by P1
6. Update rejection by P1

Note that the P-Modes configuring these exchanges are themselves updated using the update protocol. This means the updated agreement can use the mechanism to update itself again, and so on. A future version of the Usage Profile will provide all details for configuring these P-Modes.

3.5 Using a Service Provider

Some organisations do not operate a B2B Gateway themselves, but use communication services provided by a third party. For example, a service provider may provide a Protocol Bridge service to allow their customers to use other messaging protocols to communicate with them, and AS4 with their counterparties. If a service provider sends and receives AS4 messages on behalf of an organisation, it is the service provider that is responsible for selecting and deploying the AS4 Gateway, external integration, partner configuration and maintenance of such configurations.

As the **Party** identifiers of an AS4 message relate to the communication partner, their values will identify the service provider and will be different from the issuer and recipient parties identified in the EDIG@S XML document, which will identify the business partner. This difference must be communicated to and agreed with the partners. To support this, organisations will need to implement a lookup mechanism to map business partner identifiers to communication partner identifiers. This is explained in the section “Party Identification” in the ENTSOG Usage Profile. This table also needs to be managed, because organisations may switch from one service provider to another, or may decide to in-source or out-source AS4 connectivity after the initial connections with partners are established.

530 ## 4 Revision History

Revision	Date	Editor	Changes Made
Rev_0		PvdE	First Draft for discussion
Rev_1	17 Jul 2015	PvdE	• Published
Rev_1.1	14 Sep 2016	PvdE	• Document Reviewed for updates • Processing Modes details added • Addition of details of Agreement Update Specification • Reviewed at ITC KG 20 Sep 2016
Rev_1.2	5 Oct 2016	PvdE	• Feedback incorporated from ITC KG 20 Sep 2016
Rev_2	15 Nov 2016	JM	• Creation of Revision 2 for approval at ITC KG and INT WG, then publication • All tracked changes accepted

531

5 References

[AS4] AS4 Profile of ebMS 3.0 Version 1.0. OASIS Standard, 23 January 2013. http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/

[AU] OASIS ebCore Agreement Update Specification Version 1.0. OASIS Committee Specification. http://docs.oasis-open.org/ebcore/ebcore-au/v1.0/

[AS4MT] ENTSOG AS4 Mapping Table. http://www.entsog.eu/publications/as4#ENTSOG-AS4-MAPPING-TABLE

[AS4POC] ENTSOG AS4 Proof of Concept Final Report. ENTSOG . 2014-08-01. http://www.entsog.eu/public/uploads/files/publications/Events/2014/ENTSOG%20AS4%20PoC%20Final%20Report%20final.pdf

[AS4TSO] ENTSOG AS4 Usage Profile for TSOs. V3 R0 2016-11-15. ENTSOG INT 0488. http://www.entsog.eu/public/uploads/files/publications/INT%20Network%20Code/2014/int0488%20131206%20as4%20usage%20profile%20v1r0.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/os/