Creating PEPPOL Compliant Solutions

This document reviews the key elements of the PEPPOL e-signature specifications and then maps these to Ascertia's strategic technology, in particular its ADSS Server and Go>Sign Applet products. The document discusses how eID based digital signatures can be effectively created and used within a PEPPOL environment, how these can be verified by local products or by managed service providers and also reviews the way in which data can be archived and evidence for the long-term. Separate product datasheets describe ADSS Server and Go>Sign.

Contents

For further information on PEPPOL including the e-signature specifications visit www.peppol.com. For further technical information on Ascertia visit www.ascertia.com.

Background

PEPPOL (Pan-European Public Procurement On-Line) is a large-scale European Commission project. The PEPPOL vision is that any company, including SMEs in the EU can communicate electronically with any EU governmental institution for all procurement processes including both pre-award (e.g. tendering) and post-award (e.g. ordering and invoicing).

Although eProcurement processes may be implemented using manual or automated mechanisms, PEPPOL mainly addresses the automated approach. It is a system integration project focussing on how to automatically exchange structured information between the IT systems of the actors involved.

Ascertia is a global leader in delivering functionally rich yet easy to deploy security solutions. The company focuses on eID certificate validation as well as digital signature creation, verification, timestamping and secure archiving products. These deliver the essential trust services needed by governments and other organisations to conduct electronic business. Businesses need traceability, accountability and audit services plus clear originator authentication, signed approvals, assured data integrity and provenance to allow them to meet legislative, regulatory and internal controls requirements. Ascertia's products enable these security options within ERP, ECM and CRM deployments and within major managed service provider solutions.

What is PEPPOL trying to solve in the area of e-signatures?

PEPPOL Work Package 1 (WP1) addresses the important topic of "e-signatures". The results of this work package impact other packages since e-signatures must work end-to-end between the actors that engage in e-business.

The e-signature vision of PEPPOL is "*to have solutions that make it possible for economic operators in any European country to utilise the e-signatures of their own choice when submitting offers electronically to any European public sector awarding entity*." Economic operators are seen as product and service suppliers responding to public tenders. PEPPOL's ultimate interoperability aim for e-signatures can be expressed as:

- An eID holder shall be able to use the eID to sign a document towards any counterparty, even internationally. The eID holder independently selects the eID to use
- The receiver (relying party, RP) of a signed document shall be able to accept signatures from all counterparties, regardless of the eID used by the counterparty. In an open market, the RP has no influence on a counterparties' selection of eID
- A third party, receiving a document signed by other parties, shall be able to verify the signatures no matter which eIDs has been used by other parties. A signing party does not know at the time of signing who may need to verify their signature.

What is involved in accepting e-signatures?

The Relying Party (RP) role is clearly the one facing substantial complexity. The eID holder has one trusted party to rely on: the eID certificate issuer, or Certification Authority (CA). Given today's predominant trust models in the PKI area, the RP however must rely independently on each and every CA used by its counterparties.

PEPPOL therefore describes the interoperability challenges from the viewpoint of an RP as the receiver of a digitally signed document. How the eID holder digitally signs a document is largely considered to be outside the scope of PEPPOL.

In terms of verifying signatures PEPPOL recommends that the RP must check:

- The relevant signature formats (such as PKCS#7, CMS, XML DSIG etc.) including all necessary modes (enveloped, enveloping, and independent/detached) for multiple signatures.
- All necessary hash and crypto algorithms.
- The eIDs of all signers.

Processing of an eID consists of the following steps:

- Parsing and syntax checking of the eID certificate and its contents, including some semantic checking such as the use of certificate compared to allowed use (stated via key usage settings) and the presence of mandatory fields and critical extensions.
- Validation of the CA's signature on the eID certificate. This requires a trusted copy of the CA's own public key, either directly available, or obtained from further certificates in a certificate path.
- Checking that the eID is within its validity period, and that the eID is not revoked, i.e. declared invalid by the CA before the end of the certificate's validity period.
- Semantic processing of the eID content, extracting information that shall be used for presentation in a user interface or as parameters for further processing by applications. The name(s) in the eID and interpretation of naming attributes are particularly important.
- In the case of certificate chains, repeated processing for each certificate in the path.

Although the technical validation of signatures and eIDs has its challenges with respect to scaling, the real problem to the RP is the assessment of the risk implied by accepting the signature (or an eID used for some other purpose), determined by the legal status, the quality of the eID and the cryptography used, the liability position, and the trustworthiness of the CA.

At a high-level PEPPOL proposes to resolve the complexities of e-signature interoperability through the following:

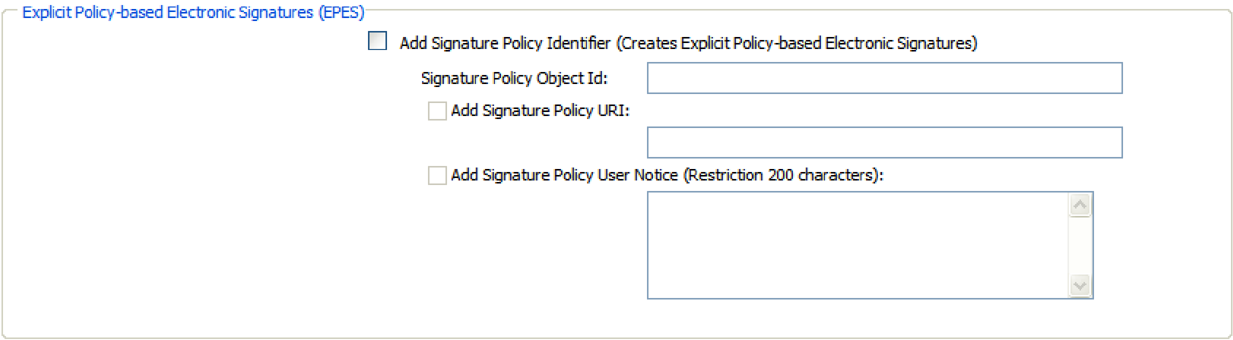
- Use of signature policies to define the acceptance criteria for e-signature.
- Provision of Validation Authority (VA) services based on OASIS DSS Verify protocol and W3C XKMS Validate protocol. Although it is recognised that RP's may perform the validation service locally by employing an appropriate software solution, the use of a validation authority service that also takes on the liability and risk associated with trusting e-signatures and eIDs makes more sense according to PEPPOL.

The following sections list specific PEPPOL requirements for signature creation and verification and how Ascertia addresses these within its ADSS Server and Go>Sign Applet products. Comments are also provided on the product roadmap for full compliance with PEPPOL.

Signature Policies

PEPPOL recognises that the use of Signature Policies as standardised by ETSI some time ago has not fully taken off but realises that these are exactly what is needed to clearly specify the rules of signature acceptance. Most of the signature policy rules defined by PEPPOL relate to the rules prescribed for the verifier / relying party.

Ascertia agrees with the use of signature policies and has supported this concept within its ADSS Server signing and verification services for some time. Although the signing environment is outside the scope of PEPPOL, it is worth mentioning that Ascertia's ADSS Signing Service fully supports the embedding of Signature Policy extension in XAdES and CAdES signatures to create Explicit Policy-based Electronic Signatures (EPES). Such EPES signatures have the signature policy OID, URI and user notice added to the digital signature:



Explicit Policy-based Electronic Signatures (EPES)

□ Add Signature Policy Identifier (Creates Explicit Policy-based Electronic Signatures)

Signature Policy Object Id:

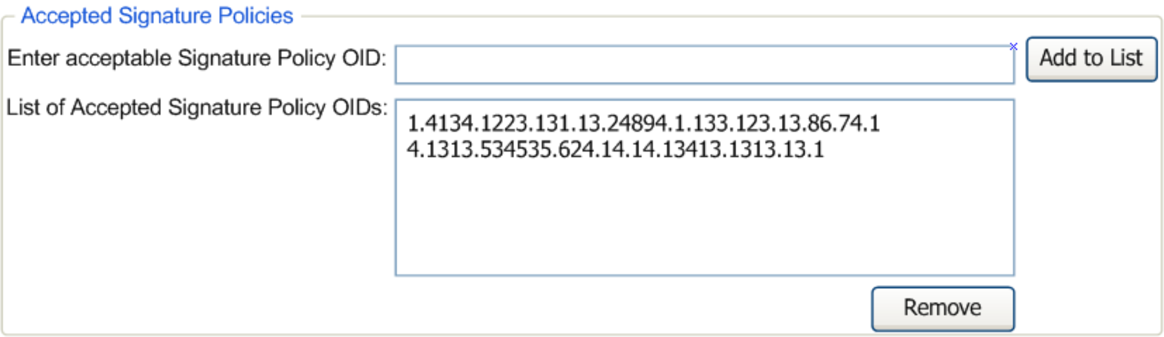
□ Add Signature Policy URI:

□ Add Signature Policy User Notice (Restriction 200 characters):

Although PEPPOL does not mention this, Ascertia recommends that signers should include the relevant signature policy identifier within the signatures they create, to acknowledge that the signer is aware of the rules defined within the signature policy and signing the document in accordance with these. The signature policy identifier embedded within the signature will also allow the RP to determine which signature policy (i.e. validation rules) to follow when verifying such signatures. The alternative is for the RP to follow only one set of validation rules, i.e. only one default configuration.

ADSS Server supports the configuration of multiple signature validation policy rules. From version 4.1 onwards, ADSS Verification Services optionally also support the verification of digital signatures against a list of acceptable signature policies configured on a per RP basis. A signature is then only trusted if it not only passes all the cryptographic and certificate path building checks but also if it contains an embedded signature policy identifier that is accepted by the RP application. This could even be considered as an initial check before processing the more advanced verification steps.

The following ADSS Server screenshot shows how an operator can define the accepted signature policy identifiers on a per client basis:



Enter acceptable Signature Policy OID:

List of Accepted Signature Policy OIDs:

1.4134.1223.131.13.24894.1.133.123.13.86.74.1
4.1313.534535.624.14.14.13413.1313.13.1

Commitment Rules – Names, Roles, Authorisations

With respect to commitment and authorisation, the usual requirement in EU Member States is that, when a signature is required, a personal signature from an authorised person is needed. A signature binds to the name in the eID, usually a person's name only. The Relying Party will then usually need additional assurance that this signature also represents the signer's organisation and that the person has the required role and authorisations.

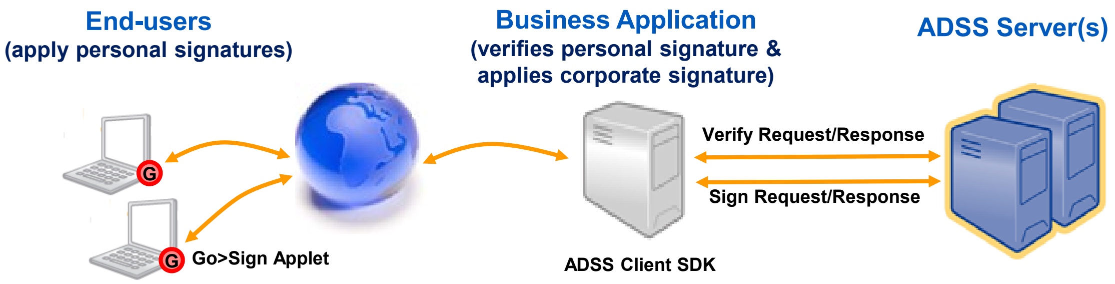
PEPPOL identifies various methods for dealing with this issue, i.e. verifying that the organisation has authorised the signer to sign on behalf of the organisation, these include:

- Just using a signature of sufficient quality - if something goes wrong then a strong proof exists through the signature
- A registration process which binds the eID to roles and authorisations within an organisation at the start of the tendering process
- Binding between names and roles/authorisations are "automatically" established by means of a VCD (Virtual Company Dossier, studied by PEPPOL WP2) or by use of business registers.
- Use of employee eIDs that also include the organisation name
- Use of corporate eIDs that only include the organisation name
- Combination of inner employee signature using personal eID and outer corporate signature using corporate eID

PEPPOL then goes on to make the conclusion that the first option above is the most pragmatic choice for the PEPPOL pilots. However it also states that corporate signatures and in particular combination of personal and corporate signatures may be studied at a later stage as these approaches hold promise for the future.

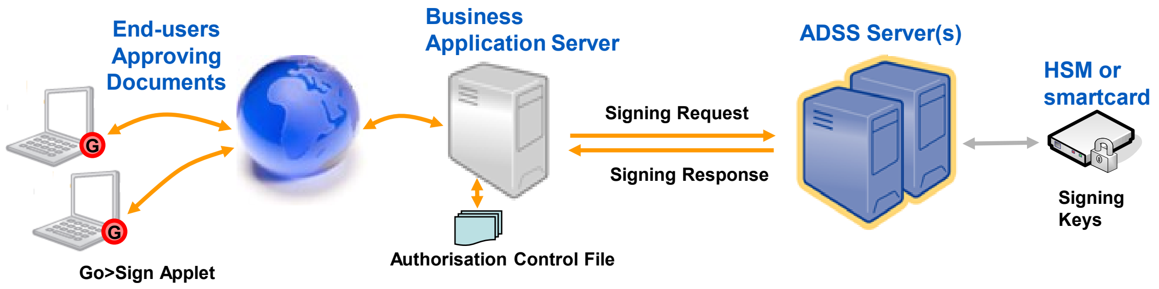
Ascertia ADSS Server together with Go>Sign Applet can generally support person signatures using locally held signing keys, corporate signatures using secure server-side signing keys, or the combination approach mentioned above. In particular the combination approach can be met in two different ways:

- In the first case the document is signed by the signer using Go>Sign Applet and a locally held Secure Signature Creation Device (SSCD) e.g. a secure smartcard or USB token. This personal signature can then be sent to ADSS Server for verification before applying a corporate signature using a corporate eID held within a Hardware Security Module (HSM) connected to the ADSS Server. The second signature could wrap the original document and the personal signature.



- In a second more advanced approach Ascertia has implemented the concept of authorising corporate signatures using an M of N approach. Under this scheme the server-side signing profile is assigned an authorisation policy that defines the employee(s) who can authorise the use of a corporate signature and the minimum number of these employees needed for the authorisation process to complete (i.e. M employees out of a set of N employees). Now each employee that is authorised can sign the document using a locally held SSCD and Go>Sign Applet, the business application can aggregate all the authorisations from multiple employees and then send the full set to ADSS Server, which verifies that the M of N rules are met before applying a corporate signature to the document or set of documents. This presents a very strong control and proof that a corporate signature was only applied after the correct number of employees had approved the document to be signed using the corporate eID. The personal authorisation signatures can be kept for later proof, or if required even supplied to the Relying Party.

The second approach is also very useful where a large number of documents, for example ten or more need to be signed. This can often be the case for large complex tender submissions. Asking a business user to sign these individually can become tedious with most signing software, especially as the PIN/password needs to be entered for each document signature.



Using the authorised signing scheme described above becomes attractive. Instead of the employee signing each document individually, they sign an authorisation control file which contains the hash values of all the documents they approve. This request is then sent to ADSS Server for processing, which verifies the approval of each document by comparing hashes and then automatically applies a corporate signature to each document if the M of N approval requirements are met. If the documents are changed after the employee has approved them then the corporate signing step will fail.

Supported Signature Formats

PEPPOL describes the following approaches of how the signature can be combined with the document it is signing:

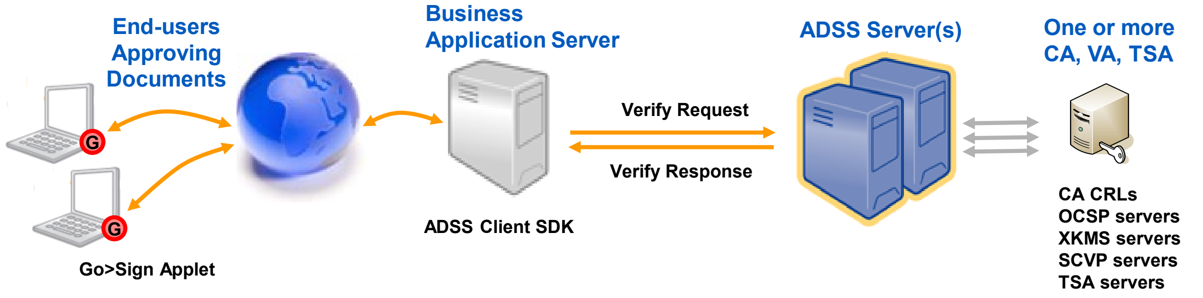
- Attached signatures: this is where the signature includes the original document content also. Ascertia ADSS Server and Go>Sign Applet support attached signatures.
- Detached signatures: this is where the signature is managed as a separate object to the document. Ascertia ADSS Server and Go>Sign Applet support detached signatures.

PEPPOL describes the following approaches for applying multiple signatures to the same document(s):

- Sequential signatures: The new signature is created over a data set made by data and previous signature(s). ADSS Server and Go>Sign Applet support sequential signatures.
- Parallel signatures: The new signature covers the data set only, meaning signatures are at the same level. Currently Ascertia does not support parallel signatures within its ADSS Signing Service, however parallel signatures can be verified within the ADSS Verification Service. The ability to create parallel signatures requires a minor update to ADSS Signing Service and can be implemented upon request.
- Countersignatures: The new signature covers old signature(s) only, the latter signature attesting to the first signature only and not to the content of the document. Ascertia does not currently support this within its ADSS Signing or Verification Service due to lack of market demand for such signatures. PEPPOL also recognises that such signatures are not widely used and are therefore discouraged as they may lead to interoperability problems.

In terms of signature formats PEPPOL recommends the use XML DSig for post-award documentation because these documents tend to based on structured XML. In terms of advanced ETSI XAdES signatures (e.g. XAdES-T or XAdES-X-L) it leaves it to the verifier to convert a basic XML DSig to one of the XAdES signature formats and does not recommend that the signer should do this. For documents exchanged during the tendering process, PEPPOL recognises the need for PDF and also PKCS#7/CMS signatures.

Ascertia's products exceed the PEPPOL requirements and they are unique in supporting all signature formats for both signature creation (within the ADSS Server for corporate signatures and within the Go>Sign Applet for personal signatures) and signature verification (within the ADSS Server).



The following table illustrates the Ascertia coverage against all the popular and advanced signature formats:

Signature Format	Ascertia Product Compliance for Signature Creation	Ascertia Product Compliance for Signature Verification
XML Signatures		
XML DSig	✓	✓
XAdES-BES	✓	✓

Signature Format	Ascertia Product Compliance for Signature Creation	Ascertia Product Compliance for Signature Verification
XAdES-EPES	✓	✓
XAdES-T	✓	✓
XAdES-X (Type 2)	✓	✓
XAdES-X-L	✓	✓
XAdES-A	✓	✓
CMS/PKCS#7 and S/MIME Signatures		
CMS/PKCS#7/ and S/MIME	✓	✓
CAdES-BES	✓	✓
CAdES-EPES	✓	✓
CAdES-T	✓	✓
CAdES-X (Type 2)	✓	✓
CAdES-X-L	✓	✓
CAdES-A	✓	✓
PDF Signatures		
Visible Signatures	✓	✓
Invisible Signatures	✓	✓
Certify (Author) Signatures	✓	✓
Adobe® CDS Signatures	✓	✓
PAdES (part 2) Long Term Signatures	✓	✓

In the PDF world it is important to support Adobe CDS signatures because these are signed using eIDs that chain to the Adobe Root CA which is pre-embedded in Adobe Reader and hence these signatures give a green tick if the signature is verified correctly without having to manually import a Root CA (a task that many end-users find difficult). The PAdES signature format is an alignment of the PDF Signatures with ETSI CAdES and XAdES formats – therefore this is likely to be an important future requirement. Ascertia currently supports PAdES part 2, but will support the other parts once they reach a formal standard status and are implemented in the widely available Adobe Reader. An increasing number of CAs are also joining the Adobe Approved Trust List program that enables other Root CAs to be trusted by Adobe Reader 9.

PKI Interfaces

PEPPOL's approach imposes few strict requirements on the signer / sender. This raises higher requirements for flexibility on the verification / Relying Party side. It is here that complexity is found. A Relying Party (RP) may handle all verification on its own or it may rely on trusted, external validation services (technical services or a full validation authority service that may also provide liability cover). PEPPOL identifies XKMS v2 certificate validation interfaces and OASIS DSS signature verification services as possible options.

Although PEPPOL specifies both XKMS and OASIS DSS, its pilot is based solely on XKMS. In Ascertia's opinion this is a significant limitation since with XKMS the RP application can only delegate certificate validation to the trusted external authority leaving it to handle all the signature complexity itself. Bearing in mind multiple different signature formats that a signer may use, verification of all of these becomes very complex unless an advanced product such as ADSS Server is used by the RP either locally or via a service. There is also a need for the RP to convert basic signatures to timestamped and archive format signatures (XAdES) although such tasks are best left to a managed service provider.

As the standards in this area have matured, Ascertia ADSS Server now supports both XKMS certificate validation services and also OASIS DSS signature verification services. Furthermore ADSS Server v4.1 also supports the latest DSS-X Signature Verification Report specifications. These provide a very detailed set of validation results for each signature on a document as requested by the RP. The DSS-X specifications are not yet fully stabilised however Ascertia continues to show its commitment and leadership in this area by delivering these capabilities before the market demand has really started. DSS also allows historic verification checks to be requested.

Ascertia is a pioneer in the Validation Authority arena. Several years ago an extension to the OASIS DSS protocol was developed for a Global Validation Service project with DNV to handle advanced digital signature verification services and an associated signature and certificate quality rating mechanism. This ADSS protocol has been effectively superseded by OASIS DSS v1.0. Today BBS AS now operates this GVS service and uses the full power of ADSS Server to offer comprehensive signature verification and certificate validation services. BBS offers effective liability and service availability and today supports around 45 CAs - see www.bbs-nordic.com/en/ for further details.

PEPPOL D1.1 part 4 recognises that sending the entire content of a signed document to a validation service may reveal confidential information to the validation service and since documents may be large, response time may be slow due to the time needed to transmit the request. Ascertia agrees with this approach and again has for some time provided an ADSS Server gateway product as part of the DNV / BBS Global Verification Service. The purpose of this product is to strip signatures from documents and only send the signature objects for verification.

ADSS Server v4.1 also offers an SCVP (Server-side Certificate Validation Protocol) service which, although not recommended by PEPPOL because of its ASN.1 encoding rather than an XML/SOAP web interface, still has certain advantages including the ability to request historic certificate validation (which is not possible with the XKMS standard).

Standard Signature Verification & Certificate Validation Protocols/Methods	PEPPOL Requirement	ADSS Server Compliance
CRL	✓	✓
OCSP	✓	✓
XKMS v2	✓	✓
OASIS DSS Verify Protocol	✓	✓
OASS DSS-X Verification Reports		✓
SCVP		✓
Gateway interface for confidentiality (hash verification)	✓	✓

Timestamping and Archiving

PEPPOL does not require the signer to timestamp the signatures, however it does require the Relying Party (RP) to time stamp all events and all validation processes.

Specifically PEPPOL D1.1 part 3 states that although logging may be sufficient to trace events during the business process execution and shortly afterwards, however trying to solve retention requirements such as those imposed by the public procurement Directives (typically 10 years) by retaining logs is seen as problematic. At some point the (original) documents must be preserved as archival records with the necessary time information and validation information as metadata.

Within archive records, time stamps are associated with documents as metadata. Records may be used in the execution of a business process, or be created at a later stage based on logs and other information collected during the process. An example of a record structure is a signed data object such as XAdES-A [ETSI-101-903] or CAdES-A [ETSI-101-733] archive formats.

Long term archival as such, and specifically the use of "advanced" archival formats of XAdES-A and CAdES-A, are not addressed by PEPPOL, and left as a local matter to the receiving e-procurement system.

PEPPOL also states that a validation service may support "historical verification and validation", i.e. verification of a signed document or validation of an eID relative to either a time indicated in the request or to time stamps in the signed data object submitted in the request. In order to achieve this, the validation service must either rely on revocation information (OCSP response or CRL) provided within the signed data object, or it must have access to old CRLs (a CRL archive) for the CAs in question.

The Ascertia ADSS Server supports all of the above time-stamping, archiving and historic verification requirements and goes much further:

Time Stamping, Archiving & Historic Verification Protocols/Methods	PEPPOL Recommendations	ADSS Server Compliance
TSP (RFC3161)	✓	✓
Archiving using XAdES-A and CAdES-A	✓	✓
Archiving using IETF LTANS specifications		✓
Historic verification using embedded CRLs/OCSP	✓	✓
Historic verification using archived CRLs		✓

Ascertia ADSS Server supports the creation of XAdES-A and CAdES-A archived format signatures at the RP side, by timestamping the basic signatures sent by the eID signers (economic operators) and also embedding the full certificate status information as meta-data.

Ascertia offers an IETF compliant Long-Term Archive & Notary Service (LTANS) service module within ADSS Server. The advantage of this service module is that it can be used to securely archive any type of data and not just signed documents as in the case of XAdES-A and CAdES-A archive formats. Therefore LTANS could be used by an e-tendering system to archive all data objects associated with the tender rather than just signed responses from the "economic operators".

ADSS Server also supports the historic verification of signed archive objects which complies with the XAdES-A and CAdES-A formats, by using the embedded timestamps and certificate status information, as well as the verification of basic signatures (XML DSig or PKCS#7/CMS or PDF signatures) using an archive of old CRLs maintained by ADSS Server.

Signature & Certificate Quality Requirements

PEPPOL notes that the quality and approval requirements vary significantly across EU member states for e-procurement. Specifically it notes that out of 15 countries with e-procurement services for tendering in operation, 6 require qualified signatures, 7 require advanced signatures (sometimes with the additional requirement of a qualified eID), while two countries require only authentication. The services furthermore either list one or a few eID issuers or are able to accept all domestic issuers and perhaps a few foreign issuers.

In PEPPOL's view, differences in national legislation as well as different requirements for different e-procurement processes necessitate development of a framework to enable specification of the crucial elements of signature policies. The specification must provide nondiscriminatory rules for acceptance of eIDs to replace present policies for national solutions, which refer to domestic issuers or national accreditation schemes.

As a part of the quality requirements, PEPPOL D1.1 Part 7 defines a framework for assessing the quality of e-signatures. This framework extends the original framework developed by DNV in conjunction with Ascertia as part of the DNV Global Validation Service.

The PEPPOL quality rating framework is based on the following aspects:

- eID quality: consisting of a certificate quality parameter ranging from 0 to 6 and an independent assurance parameter ranging from 0 to 7
- Hash quality: ranging from 0 to 5
- Public key quality: ranging from 0 to 5

Each quality aspect is briefly summarised below:

Certificate Quality level

Quality Level	Definition	Explanation
0	Very low or non determined level	Very low confidence or assessment not possible, usually because a certificate policy does not exist.
1	Low level	Low confidence in certificate but certificate policy exists or quality assessment is possible by other means.
2	Medium level	Certificates governed by a Certificate Policy in compliance with the ETSI TS 102 042 standard for LCP or a similar standard.
3	High level	Certificates governed by a Certificate Policy in compliance with the ETSI TS 102 042 standard for NCP or a similar standard.
4	High level +	Certificates governed by a Certificate Policy in compliance with the ETSI TS 102 042 standard for NCP+ or a similar standard. (Use of a SSCD is mandated in the CP.)
5	Very high level	Certificates governed by a Certificate Policy in compliance with the ETSI TS 101 456 standard for QCP or a similar standard
6	Very high level +	Certificates governed by a Certificate Policy in compliance with the ETSI TS 101 456 standard for QCP+ or a similar standard. (Use of a SSCD is mandated in the CP. Thus, this level supports qualified signatures according to the EU Directive on electronic signatures.)

Note:
LCP = Lightweight Certificate Policy
NCP = Normalized Certificate Policy
QCP = Qualified Certificate Policy
SSCD = Secure Signature Creation Device

Independent Assurance Level

Assurance Level	Definition	Explanation
0	No independent assurance	Self assessment only.
1	Independent document review	Statement of compliance issued by an independent, external unit based on document review only.
2	Internal compliance audit	Internal audit carried out periodically concludes compliance to applicable requirements.
3	Supervision without compliance audit	CA is supervised by a public, national or international authority according to applicable law to the CA.
4	External compliance audit	Audit carried out periodically by external, independent auditor concludes compliance to applicable requirements.
5	External compliance audit and certification	Audit carried out periodically by external, independent auditor concludes compliance to applicable requirements. CA operations are certified in accordance with a relevant standard; OR cross certification with a relevant bridge CA has been made; OR the CA has obtained membership in a PKI

Assurance Level	Definition	Explanation
		hierarchy as a result of appropriate assessment. Note: Relevant standards include ETSI TS 101 456, ETSI TS 102 042, WebTrust Program for CAs, tScheme Approval Profile for CAs, ISO9001, ISO27001.
6	Supervision with external compliance audit	Audit carried out periodically by external, independent auditor concludes compliance to applicable requirements. CA is supervised by a public, national or international authority according to applicable law to the CA
7	Accreditation with external compliance audit	Audit carried out periodically by external, independent auditor concludes compliance to applicable requirements. CA is accredited by a public, national or international authority according to applicable law to the CA.

Cryptographic Quality Level

The parameters of concern here are hash algorithm quality for the signed document and quality of the combination public key algorithm and key length. The hash algorithm for the eID certificate is considered part of eID quality.

Adapted from US recommendations [NIST01] that seem to be agreed to by most European countries as well, a starting point for quality classification can be as follows:

Quality Level	Explanation
0	Should not be trusted
1	Reasonably secure for 3 years
2	Regarded as trustworthy for 5-10 years.
3 - 5	Increasing levels of security.

There seem to be agreed judgements about which algorithms should go in which classes. This assumes no inherent (undetected) weakness in the algorithms and no implementation flaws.

As examples of hash algorithms: MD5 = 0, SHA-1 = 1, SHA-224/256/384/512 = 2/3/4/5,
and public key algorithms with key lengths: RSA-1024 = 1; RSA-2048 = 2; RSA-4096 = 4.

Example 1: Qualified Certificate and SSCD, Accredited CA

A qualified electronic signature created with an SSCD and a qualified certificate issued by an accredited CA and using the SHA-224 hash algorithm and a cryptographic key length of 2048, would have signature quality parameters as follows:

- eID quality: (6,7) – meaning certificate quality level 6 & independent assurance level 7
- Hash quality: 2 – regarded as trustworthy for 5-10 years
- Public key quality: 2 – regarded as trustworthy for 5-10 years

With the notation suggested above, this signature example would have a signature quality: signature quality = {(6,7),2,2}

XKMS and OASIS DSS Protocol Enhancements by PEPPOL

PEPPOL has extended the XKMS and OASIS DSS specifications to allow a Relying Party to identify the signature and certificate quality levels that are acceptable. The Validation Authority service can then respond to whether the signature (or certificate) meets the required quality level. A signature can be deemed to be of insufficient quality if it fails to meet the quality level but passes normal cryptographic checking, certificate path building and certificate validation checking.

ADSS Server v4.0 and earlier supported an older DNV quality rating framework, from ADSS Server v4.1 Ascertia has enhanced this to use the PEPPOL trust rating framework within the

ADSS Server Verification service using the OASIS DSS protocol. Ascertia plans to update its ADSS Server XKMS service to support the certificate quality framework in mid 2010.

Trust Models & Trust Service Status Lists

PEPPOL recognises that a Validation Authority (VA) Service may not have registered all eID issuers in a cross-border context, as there could be hundreds of such issuers (qualified CAs and non-qualified CAs). Hence it sees the need for a VA to identify a peer VA to which it can forward certificate validation requests (i.e. XKMS requests). So PEPPOL prefers the local VA to verify the signature locally but if the eID issuer is not registered then it should identify a peer VA that is responsible for the eID issuer and forward the certificate to that peer VA for validation using the XKMS protocol.

Identifying suitable peer VAs is left to Trust-service Status List (TSL) issuers. This is a signed data structure standardised by ETSI and is used to identify Trust Service Providers (TSPs). So as long as the VA trusts the local TSL issuer it can download an up-to-date TSL and use this to identify a peer VA to which it can send an XKMS request.

However PEPPOL recognises that TSLs are currently in experimental stage and no public implementations exists, so it recommends building a manually configured routing mechanism within its VAs.

The Ascertia ADSS Server does not currently support automated TSL processing but this is identified as a mid-term roadmap item. The use of a manual TSL configuration file within the VA is a relatively simple task and can be implemented fairly quickly. Ascertia is currently closely monitoring market demand for trust anchor management and can deliver this quickly to meet project requirements. Note the IETF is working on a similar trust anchor protocol called Trust Anchor Management Protocol (TAMP). Ascertia continues to track this also.

Encryption of Tender Documents

For end-to-end confidentiality, business documents should be encrypted. To encrypt, the sender (signer) needs a trusted eID certificate for the receiver, where the certificate (key usage settings) allows encryption.

PEPPOL considers encryption out of its scope however because it feels that most eID cards do not carry encryption certificates. However in Ascertia's opinion this misses the point somewhat as encryption is not required person to person, but rather person to system (e.g. tendering application).

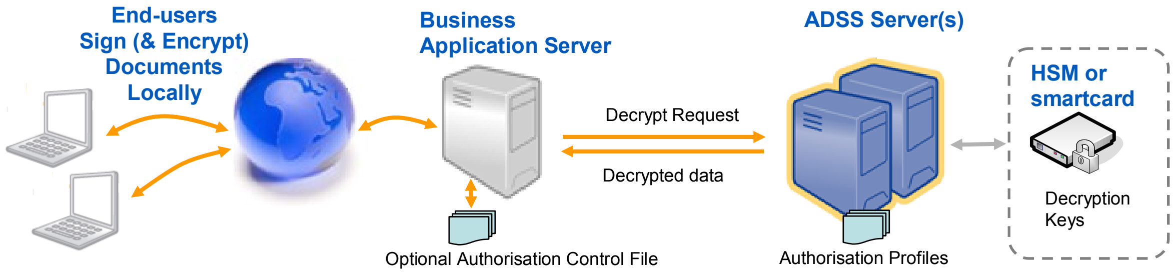
PEPPOL does recognise that the solution is to use corporate certificates for the receiver, but states that it may be too long-term for the PEPPOL pilots to support this. It further states that such a solution, and its inherent trust issues such as being able to obtain and trust the encryption certificate of the receiver, are possibly for further study by PEPPOL WP1.

It also states that there are requirements (e.g. in France) for encryption of tendering documents until time of opening of the bids. In such cases, PEPPOL WP1 recommends tendering platforms provide an "upload and encrypt" function to this effect. On upload over a protected channel, the receiving system will immediately encrypt all documents using a certificate and public key whose corresponding private key will only be made available to the receiver after a certain time. However such a solution is considered to be out of scope for PEPPOL.

Interestingly Ascertia has recently participated within a European Healthcare e-procurement project that had exactly these requirements for confidentiality. As a result of this work ADSS Server now offers this feature in the following ways:

- Encrypt and upload: Ascertia Go>Sign Applet was enhanced to not only create XAdES-X signatures over the tender submission documents but to also encrypt (using XML Encryption) this payload using a certificate provided by the Awarding Entity.

- Decryption Service: ADSS Server was enhanced to offer a server-side OASIS DSS-X based decryption service. In this scenario ADSS Server is the secure custodian of the decryption key and will only allow decryption based on an authorised client application making a request. If required M of N end-user authorisers may approve the decryption request by signing the decryption request message. As explained above for corporate signatures, ADSS Server can verify that the required number of authorisers have approved the decryption request before performing the decryption and returning the cleartext document(s) to the client application. Embargo dates can also be supported upon request where each encrypted object contains a date before which the decryption cannot occur; ADSS Server verifies that this date is passed before allowing decryption operation to proceed.



Summary

As can be seen from the above Ascertia today meets and often exceeds the PEPPOL requirements for e-signatures. OASIS DSS is seen as a very comprehensive standard for digital signature creation and verification and together with Trust Anchor selection and historic validation is therefore recommended over XKMS.

Ascertia's product strategy is to track, support and exceed PEPPOL requirements. For further information on any of the above concepts or on how to use Ascertia technology to deliver trust within your business documents and workflow processes contact info@ascertia.com