UBL Digital Signature Profiles 1.0

Committee Draft 11

24 April 2011

Specification URIs:

This Version:

http://docs.oasis-open.org/ubl/cd11-UBL-DSig-1.0/UBL-DSig-1.0.html
http://docs.oasis-open.org/ubl/cd11-UBL-DSig-1.0/UBL-DSig-1.0.pdf
http://docs.oasis-open.org/ubl/cd11-UBL-DSig-1.0/UBL-DSig-1.0.xml (Authoritative)

Previous Version:

N/A

Latest Version:

http://docs.oasis-open.org/ubl/UBL-DSig-1.0/UBL-DSig-1.0.html
http://docs.oasis-open.org/ubl/UBL-DSig-1.0/UBL-DSig-1.0.pdf
http://docs.oasis-open.org/ubl/UBL-DSig-1.0/UBL-DSig-1.0.xml (Authoritative)

Technical Committee:

OASIS Universal Business Language Security SC

Chairs:

Andrea Caccia <andrea.caccia@studiocaccia.com>
Julián Inza <julian.inza@albalia.com>

Editors:

Oriol Bausà Peris <oriol@invinet.org>
Andrea Caccia <andrea.caccia@studiocaccia.com>
Roberto Cisternino <roberto@javest.com>
G. Ken Holman <gkholman@CraneSoftwrights.com>
Julián Inza <julian.inza@albalia.com>

Related Work:

This specification relates to all versions of OASIS Universal Business Language (UBL) 2.x.

Declared XML Namespaces:

urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2
urn:oasis:names:specification:ubl:schema:xsd:SignatureAggregateComponents-2
urn:oasis:names:specification:ubl:schema:xsd:SignatureBasicComponents-2

Abstract:

This specification defines two profiles for digitally signing Universal Business Language (UBL) 2.x documents and a standard digital signature extension for use with the enveloped profile.

The profiles are based on IETF/W3C XML Digital Signatures, with specific provisions to use XAdES extensions when the electronic signing of UBL documents addresses special advanced legal and technical requirements.

Status:

This document was last revised or approved by the UBL TC on the above date. The level of approval is also listed above. Check the current location noted above for possible later revisions of this document. This document is updated periodically on no particular schedule.

Technical Committee members should send comments on this specification to the Subcommittee's email list. Others should send comments to the Subcommittee by using the "Send A Comment" button on the Subcommittee's web page at http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=ubl-security.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page at http://www.oasis-open.org/committees/ubl/ipr.php.

Notices

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Committee Specification or OASIS Standard, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification.

OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this specification by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this specification. OASIS may include such claims on its website, but disclaims any obligation to do so.

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Committee Specification or OASIS Standard, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.

The name "OASIS" is a trademark of OASIS, the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see http://www.oasis-open.org/who/trademark.php for above guidance.

Table of Contents

Appendix

1. Introduction (Non-Normative)

There are certain circumstances in which it becomes necessary to electronically sign UBL documents. This can be the case when creating orders or invoices. In some countries, digitally signing electronic invoices is required by law.

UBL has an ABIE to define signatures and a number of ASBIEs to use such signatures in a document. (See current UBL documentation for more regarding these terms.) There are a number of standard initiatives in the electronic signature area that are being adopted or recommended by different organizations or bodies. This specification standardizes the use of the XML Signature Specification [XMLDSig] in and for UBL documents and recommends their association with the UBL BIEs.

Note

The implementation of the extension used in the "enveloped" profile described below also serves as a model of a typical UBL extension. Those wishing to create their own UBL extension can mimic the schema and namespace structures used here.

[XMLDSig] is a general framework for digitally signing XML documents. ETSI TS 101 903, also known as [XAdES], is an XML electronic signature standard that can be used to create different XML Advanced Electronic Signatures. XAdES extends XMLDSig for use with advanced and qualified electronic signatures as specified in European Union Directive [1999/93/EC]. Use of XAdES and the concept of Advanced Electronic Signature is not limited to Europe, as it is being adopted by many countries outside the EU, and, at the time of publication of this specification, it is undergoing standardization in ISO TC 154 as ISO/CD 14533-2.

One important benefit of XAdES is that it allows the addition of information and timestamps that extend the validity of a signature beyond the expiration or revocation of the electronic certificates involved in signature verification or the obsolescence of the underlying cryptographic keys and algorithms. By extending XMLDSig with additional embedded syntax and processing, XAdES satisfies the European Commission Directive on a Community Framework for Electronic Signatures as well as other use cases requiring long-term preservation of signed documents. XAdES contains several modules that permit various levels of security, such as non-repudiation with timestamps and long-term signature verification.

The work of standardizing electronic signatures was supported by the European Commission and assigned to the Information and Communication Technologies Standards Board (ICTSB), a round table of most European IT standards bodies and some international standards bodies such as the IETF and W3C.

This UBL Digital Signature Profiles specification defines two profiles that represent two approaches to signing UBL documents: enveloped and detached. Each of these approaches uses XMLDSig in a way that may or may not include XAdES features. In other words, the mechanisms implemented here can be used not only to implement XAdES in these two ways but also to implement other signature technologies based on XMLDSig as well.

Using UBL Digital Signature Profiles one can conform to, for example, the UN/CEFACT Signed Digital Evidence Interoperability Recommendation [UN/CEFACT Rec. 37]. *[To date, this recommendation has not been published by UN/CEFACT.]*

2. Terminology

2.1. Terms and Definitions

ASiC-S
Associated Signature Container (simple form). A standard container that associates a single data object with one or more detached signature(s) that apply to it. See [ASiC].

Digital Signature
A value generated from the application of a private key to a message via a cryptographic algorithm such that it has the properties of integrity and message authentication and/or signer authentication. A signature may be (non-exclusively) described as detached, enveloping, or enveloped ([XMLDSig], with modifications).

Transform
The processing of data from its source to its derived form. Typical transforms include XML Canonicalization [XML C14N] and XSLT [XSLT 2.0].

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY and OPTIONAL, when they appear in this document, are to be interpreted as described in [RFC2119].

2.2. Symbols and Abbreviations

ABIE
Aggregate Business Information Entity

AdES
Advanced Electronic Signature

ASBIE
Association Business Information Entity

BBIE
Basic Business Information Entity

BIE
Business Information Entity

C14N
Canonicalization

DSig
Digital Signature

QC
Qualified Certificate

QS
Qualified Signature

XAdES
XML Advanced Electronic Signatures [XAdES]

XML
Extensible Markup Language

XMLDSig
XML Digital Signature [XMLDSig]

XPath
XML Path Language (an XML data model and addressing language) [XPath 2.0]

XSLT
Extensible Stylesheet Language Transformations (a transformation language) [XSLT 2.0]

3. Normative References

[RFC2119] *Key words for use in RFCs to Indicate Requirement Levels, March 1997 [http://www.faqs.org/rfcs/rfc2119.html].*

[XAdES] *XML Advanced Electronic Signatures. ETSI TS 101 903 V1.4.1, June 2009 [http://uri.etsi.org/01903/v1.4.1/ts_101903v010401p.pdf].*

[XMLDSig] *XML-Signature Syntax and Processing. W3C Recommendation, 12 February 2002 [http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/].*

4. Non-Normative References

[1999/93/EC] *Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures [http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31999L0093:EN:NOT].*

[ASiC] *Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASIC). ETSI TS 102 918 V1.1.1, April 2011 [http://webapp.etsi.org/WorkProgram/Report_WorkItem.asp?WKI_ID=31946].*

[CWA15579] *CEN Workshop Agreement: E-invoices and digital signatures (CWA 15579), July 2006 [ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/eInvoicing/CWA15579-00-2006-Jul.pdf].*

[CWA15580] *CEN Workshop Agreement: Storage of Electronic Documents (CWA 15580), July 2006 [ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/eInvoicing/CWA15580-00-2006-jul.pdf].*

[ODFP] *OASIS Standard, Open Document Format for Office Applications (OpenDocument) Version 1.2 - Part 3 Packages, December 2006 [http://docs.oasis-open.org/office/v1.2/csprd03/OpenDocument-v1.2-csprd03-part3.pdf].*

[RFC3161] *Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP), August 2001 [http://www.faqs.org/rfcs/rfc3161.html].*

[XML C14N] John Boyer, *Canonical XML Version 1.0, 15 March 2001 [http://www.w3.org/TR/2001/REC-xml-c14n-20010315].*

[UN/CEFACT Rec. 37] *Signed Digital Evidence Interoperability Recommendation, 27 September 2010 [http://www.unece.org/cefact/cf_plenary/plenary10/ECE_TRADE_C_CEFACT_2010_14E.pdf].*

[XPath 2.0] Anders Berglund, et al., *XML Path Language (XPath) Version 2.0, 23 January 2007 [http://www.w3.org/TR/2007/REC-xpath20-20070123/].*

[XPointer] Steven DeRose, et al., *XML Pointer Language (XPointer) Version 1.0 Working Draft, 16 August 2002 [http://www.w3.org/TR/xptr/].*

[XSLT 2.0] Michael Kay, *XSL Transformations (XSLT) Version 2.0, 2007-01-23 [http://www.w3.org/TR/xslt20/].*

5. Referenced Namespaces

The table below lists the namespaces referenced in this specification. The prefixes on the left are only documentary conventions; their choice is not constrained by XML.

Table 1. Referenced Namespaces

Prefix	Namespace	Reference
ds	http://www.w3.org/2000/09/xmldsig#	[XMLDSig]
xades	http://uri.etsi.org/01903/v1.3.2#	[XAdES]
ext	urn:oasis:names:specification:ubl:schema:xsd:CommonExtensionComponents-2	UBL extension namespace
sig	urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2	UBL signature extension apex namespace
sac	urn:oasis:names:specification:ubl:schema:xsd:SignatureAggregateComponents-2	UBL signature extension aggregate namespace
sbc	urn:oasis:names:specification:ubl:schema:xsd:SignatureBasicComponents-2	UBL signature extension basic namespace

6. XML Digital Signatures

6.1. Overview (Non-Normative)

Digital signatures, when appropriate rules and functions are used, can support the following properties for a document:

- Integrity: the document has not been modified since it was signed.
- Authenticity: the identity of the party creating the signature that applies to the document is certified.
- Non-repudiation (or content commitment): the document signer cannot deny its involvement in creating and/or approving the document (depending on the context and signer role).
- Anteriority: associating a time-stamp to the signature, a proof that the signature (and therefore the signed document) existed before a certain point in time.

[XMLDSig] defines XML Signature processing rules and syntax to provide integrity and message authentication and/or signer authentication services for data of any type, whether located within the XML that includes the signature or elsewhere. [RFC3161] specifies a standard format for time-stamping that can be used with XMLDSig and XAdES.

The [1999/93/EC] directive defines the following technology-neutral requirements that an electronic signature must meet to be considered an Advanced Electronic Signature (AdES) and have legal validity:

- it is uniquely linked to the signatory;
- it is capable of identifying the signatory;
- it is created using means that the signatory can maintain under his sole control; and
- it is linked to the data to which it relates in such a manner that any subsequent change of the data is detectable.

The Qualified Signature (QS) is also defined as an AdES based on Qualified Certificates (QC) and Secure Signature Creation Devices for signing operations. In Europe, QS is equivalent to handwritten signature provided it is based on a QC issued by an accredited Certificate Service Provider. These references are provided only for informational use and refer to the framework defined in [1999/93/EC].

XAdES extends XMLDSig to support AdES, but its adoption is not limited to an EU context, as similar requirements are in place in other countries. The introduction to [XAdES] reads, in part,

> The XML advanced electronic signatures defined in the present document will be built by incorporating to the XML signatures as defined in XMLDSIG one new `ds:Object` XML element containing the additional qualifying information.

That XAdES is completely embedded in XMLDSig ensures that the UBL profiles for XMLDSig are sufficient to support XAdES. These profiles also support other existing or future extensions of XMLDSig that are completely embedded in XMLDSig syntax. These other possible UBL digital signature profiles may or may not use the XAdES extensions to XMLDSig.

It is important to note that XAdES and XMLDSig define digital signature processing rules and syntax but do not cover the implementation of security measures required for an AdES, which are out of scope for this document.

Implementation may depend on local regulations in place and specific provisions set by the authority issuing the certificates supporting the signature. The implementer has to determine the set of requirements that apply to the specific context of use and determine accordingly the suitability of the standards and

the specific profiles to be used. XAdES can help in fulfilling legal requirements, but this is not just a matter of correctly applying a technical standard. Users are advised to examine the regulations applicable to their specific context of use.

6.2. XML Signature Types (Non-Normative)

An XML signature may be (non-exclusively) described (per XMLDSig and XAdES) as detached, enveloping, or enveloped.

- **Detached.** The signature applies to content that is external to the `<ds:Signature>` element and can be identified via a URI or transform. Consequently, the signature is "detached" from the content it signs. This definition typically applies to separate data objects, but it also includes the case where the `<ds:Signature>` and signed data object are sibling elements residing within the same XML document.
- **Enveloping.** The signature applies to content found within a `<ds:Object>` element of the signature itself. The `<ds:Object>` (or its content) is identified via a `<ds:Reference>` (using a URI fragment identifier or transform).
- **Enveloped.** The signature applies to the XML content that contains `<ds:Signature>` as an element. Implementations of enveloped signature(s) must take care not to include the signature in the calculation of the signature value.

This specification defines two profiles for signing a UBL document: enveloped and detached.

6.3. XAdES (Non-Normative)

A compliant implementation of XAdES guarantees wide acceptance in implementing legal regulations, such as EC Directive [1999/93/EC], and supports best practices in eInvoicing, eProcurement, and eBusiness in general as set forth by relevant standard bodies such as CEN [CWA15580] and [CWA15579].

The UBL implementation of XAdES provides the following additional properties:

- A signed UBL document will be processed correctly by any compliant UBL software (including UBL software that is not XMLDSig/XAdES aware) and by any compliant XMLDSig/XAdES verification software (including software that is not UBL aware)
- No change is required for currently defined UBL or XMLDSig/XAdES syntaxes
- The extension mechanism specified here supports any XMLDSig/XAdES form, leaving to the implementer the choice of the most appropriate one according to the specific legal framework or application context.

XAdES defines a set of forms that extends XMLDSig and allows adding to the signature some validation data.

The two basic forms are:

- **XAdES-BES**, which satisfies the minimum requirements for AdES; and
- **XAdES-EPES**, which builds on XAdES-BES to include a security policy identifier that specifies the rules followed to validate the signature.

A conformant XAdES signature generation and verification implementation supports at least XAdES-BES or XAdES-EPES.

The other forms can be built by the signature generator or the signature verifier by extending one of the two basic forms. They are:

- **XAdES-T**, where a timestamp is added to enforce non-repudiation and as a proof of anteriority. This envelope allows ascertaining the validity of a signature in case the signer certificate is later revoked;
- **XAdES-C**, which adds to the signed document a complete reference to verification data (certificates and revocation lists) to support long-term signature verification;
- **XAdES-X**, which adds timestamps to XAdES-C references to protect against future compromise of certificates;
- **XAdES-X-L**, which is similar to XADES-X but adds real certificates and revocation lists instead of just references; and
- **XAdES-A**, which adds timestamps (periodically, as required) to extend the validity period for long-term storage, taking into account a possible weakening of the algorithms used to sign the document and related certificates during the storage period.

This specification does not recommend any specific XAdES form for a UBL document, as this choice depends on the specific context of use, agreements between the parties, and local regulations.

6.4. UBL Signature Profiles

This document specifies two profiles for use in digitally signing UBL documents:

- **Enveloped Signature Profile:** One or more signatures are added to the UBL document inside a single identifiable and dedicated UBL Extension. Other UBL extensions MAY be present provided they have different identifiers so that they can be distinguished from the one that contains the document signature(s). This profile is defined such that UBL content processing can be separated from electronic signature processing, both on the issuing side and on the receiving side, and specialized applications can be devoted to each function. The UBL application doesn't need to be electronic signature aware, and the electronic signature application does not need to be involved in the management of the UBL syntax. A signature business object in the UBL document may reference a particular electronic signature in the extension.
- **Detached Signature Profile:** The signature is outside the UBL document content in another information resource. Some mechanism has to be defined by the implementer to send or make available the signature to the recipient. This method of signing may be identified in the UBL document. This approach can be useful to avoid or minimize any kind of modification to the UBL document and is compatible with other signature methods not explicitly referenced by this profile.

6.5. Requirements for Digital Signatures in UBL

The main requirements to be addressed when choosing a specific signature profile can be divided into the following categories:

- **Legal requirements.** In some countries a digital signature is required on electronic invoices. It can also be compulsory in electronic procurement, especially in a cross border context, to have digital signature on the key document exchanged, e.g., on orders. Another important legal requirement is long-term document preservation, for a storage period that in general is specific in each country and can span many years. The requirement to guarantee the integrity and authenticity of all fiscally relevant archived documents, as specified, for example, by [CWA15580] for electronic invoices, can be met with digital signatures when proper XAdES forms are used.
- **Business requirements.** A digital signature can reduce the risks associated with a business transaction (e.g., non-repudiation of a commercial order, proof-of-origin and integrity of an invoice) and its use can be provided for in the interchange agreement between parties. The choice of the signature format and its application is a key element for interoperability.

- **Process requirements.**The presence of the digital signature should not add any specific constraints on UBL document content processing. If the signed document remains a valid UBL document, the signature can be verified at any stage of the process: it should be possible to validate a signed document at any time "as is" by UBL and XAdES verifiers.

7. Profiles for UBL Digital Signatures

The two profiles for adding one or more digital signatures to a UBL document are based on [XMLDSig]. These profiles and their associated methods decouple the UBL document to be signed from any specificity in the digital signature standard adopted within XMLDSig. The XAdES standard is an example of a standard use of XMLDSig. UBL users may use any standard built on XMLDSig or simply use XMLDSig as it stands without any extensions.

Managing XML signatures inside of a UBL document is described in Section 7.1, "Enveloped XML Signatures in UBL Documents". Managing XML signatures outside of a UBL document is described in Section 7.2, "Detached XML Signatures for UBL Documents".

Both profiles support co-signatures, i.e., a UBL document can be independently cosigned by multiple signers in any order and time. Both profiles support countersignatures, i.e., a UBL document can have its signatures signed by another signature. The enveloped signature profile supports a final signature, i.e., a UBL document once signed with a final signature cannot have any other signature added without invalidating the final signature.

The choice of the most suitable profile should take into account mainly the specific document processing and delivery infrastructure.

The main advantage of the enveloped profile is that the signature(s) are embedded in the UBL document (which syntactically remains a valid UBL document). This means that the transport of the signatures is guaranteed by the UBL document delivery infrastructure.

The detached signature profile has a simpler preparation phase and signature procedure, but specific means to send or make available the signature(s) to the recipient have to be implemented. A standard container like [ODFP] can be used to associate the UBL document with detached advanced electronic signature(s) that apply to it. The simple [ASiC] container (ASiC-S) can be created at a later time than signature generation so that it contains a UBL document and one or more detached signatures that apply to it.

Archiving of UBL documents also can be an important issue to consider, as document preservation has specific requirements.

7.1. Enveloped XML Signatures in UBL Documents

This profile supports one or more signatures to be applied to a UBL document and embedded in the UBL document itself inside a dedicated extension. This profile can be used with all UBL documents under their respective `<ext:UBLExtensions>` extension point.

Note

The `xml/UBL-Invoice-2.0-Signed.xml` sample document in the UBL 2.1 distribution illustrates the embedding of three extensions in a single document, one of which is the signature extension.

The user MAY choose to indicate in a `<cac:Signature>` element that the signature details are found in the signature extension. The URI `urn:oasis:names:specification:ubl:dsig:enveloped` is reserved as a value for `<cbc:SignatureMethod>` to signal this. The URI `urn:oasis:names:specification:ubl:dsig:enveloped:xades` MAY be used as a value for `<cbc:SignatureMethod>` to signal when XAdES is in use. Additionally, the user MAY include a `<cbc:ID>` child of `<cac:Signature>` for referencing purposes from the enveloped signature. The identifier used can be any value, but for convenience the URI of a URN beginning with `urn:oasis:names:specification:ubl:signature:` and ending with the local name of the parent of the signature business object and optionally followed with a colon and number, as in the `urn:oasis:names:specification:ubl:signature:Is-`

suerEndorsement example, is reserved for this purpose for UBL users. As with all identifier references, the referenced identifier SHOULD exist and be unique across all such identifier values. An example is as follows:

```
 <cac:Signature>
   <cbc:ID>urn:oasis:names:specification:ubl:signature:Invoice</cbc:ID>
   <cbc:SignatureMethod
>urn:oasis:names:specification:ubl:dsig:enveloped</cbc:SignatureMethod>
   <cac:SignatoryParty>
     <cac:PartyIdentification>
       <cbc:ID>MyParty</cbc:ID>
     </cac:PartyIdentification>
   </cac:SignatoryParty>
 </cac:Signature>
```

7.1.1. Enveloped Signature Syntax

There are two distinctive levels of syntax present: UBL-specified scaffolding under the extension point used to contain the signature information and IETF/W3C-specified information for each digital signature.

One or more signature extensions in a given document MAY each contain one or more sets of signature information. The standard UBL-specified scaffolding for a given signature extension begins with the <ext:UBLExtension> element. The extension's role as a UBL signature extension is indicated with a child <ext:ExtensionURI> element with the urn:oasis:names:specification:ubl:dsig:enveloped value. The urn:oasis:names:specification:ubl:dsig:enveloped:xades value MAY be used to indicate the use of XAdES in the extension. Other extension metadata elements defined in UBL are allowed to be included for the convenience of users without changing the meaning or use of the extension.

All uses of the optional <cbc:ID> metadata SHOULD be unique so that each extension can be uniquely identified. For the convenience of users, a URI with the URN beginning with urn:oasis:names:specification:ubl:signature: and ending with a number value is reserved for this purpose for UBL users, and MAY be used. The value urn:oasis:names:specification:ubl:signature:3 is a suitable example.

The mandatory <ext:ExtensionContent> element is the extension scaffolding that contains the UBL signature scaffolding. The apex element of the UBL signature information is <sig:UBLDocumentSignatures>. Note that three namespaces are used for signature information, in parallel with the UBL design of having a document namespace, aggregate namespace and basic namespace. The apex element is in the urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2 namespace, a parallel to a UBL document namespace. Signature-related aggregate entities are in the urn:oasis:names:specification:ubl:schema:xsd:SignatureAggregateComponents-2 namespace. Signature-related basic entities are in the urn:oasis:names:specification:ubl:schema:xsd:SignatureBasicComponents-2 namespace. Accordingly, there are three W3C Schema fragments in the distribution accommodating these three namespaces.

Note

Creators of other UBL extensions using this one as a model should review the UBL specification documentation for guidelines regarding this schema design pattern.

In each extension with signature information, the <sig:UBLDocumentSignatures> apex element contains one or more individual <sac:SignatureInformation> aggregates. One aggregate is used to contain the information related to a single IETF/W3C digital signature.

An aggregate MAY be identified for referencing purposes using the common library <cbc:ID> element. Such an identifier MAY be used in scenarios where a particular signature needs to be identified external to the document, e.g. in workflow applications. The identifier used can be any value, but for convenience

a URI consisting of a URN beginning with `urn:oasis:names:specification:ubl:signature:` and ending with a number value is reserved for this purpose for UBL users. An example is `urn:oasis:names:specification:ubl:signature:3`. As with all identifiers, each SHOULD be unique across all identifier values.

An aggregate MAY make reference to an existing `<cac:Signature>` business object in the same UBL document. When needed, the `<sbc:ReferencedSignatureID>` basic element is used to point to the `<cbc:ID>` identifier value of the referenced `<cac:Signature>`. The identifier used can be any value, but for convenience a URI consisting of a URN beginning with `urn:oasis:names:specification:ubl:signature:` and ending with the local name of the parent of the signature business object and optionally followed with a colon and number, as in the `urn:oasis:names:specification:ubl:signature:IssuerEndorsement` example, is reserved for this purpose for UBL users. As with all identifier references, the referenced identifier SHOULD exist and be unique across all such identifier values.

A single `<ds:Signature>` element is a child of the aggregate. It MAY be absent from the document, thus supporting workflow scenarios where the element is added by a subsequent process after the UBL scaffolding is added by an earlier process. However, the signature information is semantically incomplete without the IETF/W3C-defined element. To support countersignatures countersigning this signature, either this element or its `<ds:SignatureValue>` child element MUST use the `Id=` attribute with a value unique from other attributes of schema type `ID` in the instance.

A skeleton example of a single signature corresponding to the example `<cac:Signature>` above is as follows:

```
<ext:ExtensionContent>
  <sig:UBLDocumentSignatures xmlns:sig=
    "urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2"
    xmlns:sac=
 "urn:oasis:names:specification:ubl:schema:xsd:SignatureAggregateComponents-2"
    xmlns:sbc=
     "urn:oasis:names:specification:ubl:schema:xsd:SignatureBasicComponents-2">
    <sac:SignatureInformation>
      <cbc:ID>urn:oasis:names:specification:ubl:signature:1</cbc:ID>
      <sbc:ReferencedSignatureID
>urn:oasis:names:specification:ubl:signature:Invoice</sbc:ReferencedSignatureID>
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id=...>
        <ds:SignedInfo>
          ...
          <ds:Reference URI=...>
            ...
            <ds:Transform>
              ...
            </ds:Transform>
            ...
          </ds:Reference>
        </ds:SignedInfo>
        <ds:SignatureValue Id=...>
        ...
        </ds:SignatureValue>
        <ds:KeyInfo>
        ...
        </ds:KeyInfo>
        <ds:Object>
        ...
        </ds:Object>
       </ds:Signature>
    </sac:SignatureInformation>
  </sig:UBLDocumentSignatures>
</ext:ExtensionContent>
```

Note

The XAdES specification contains all qualifying XAdES information in a single <ds:Object> element located as shown above.

Note

A document with multiple <sac:SignatureInformation> elements is simply a document that is co-signed. By the appropriate use of the <ds:Reference> element, all such signatures are signing the content of the document but not each other. A *countersigning* document signature, on the other hand, signs signatures already present in the document at the time it is counter-signed. A digital countersignature <ds:Signature> includes additional <ds:Reference> elements, each pointing to either the <ds:Signature> element being signed or its respective <ds:SignatureValue> element.

Note

The XAdES specification supports an alternative countersignature approach where a <ds:Signature> element pointing to the countersigned signature's <ds:SignatureValue> is embedded in the <ds:Object> of the countersigning signature. The inclusion of an alternative method in this specification does not prohibit this approach.

7.1.2. Digital Signature Transformation (Enveloped Signatures)

The content to be signed is indicated in the URI= attribute of <ds:Reference>. Using the empty string indicates that the entire document (i.e., the enveloping UBL instance) is what is being signed:

```
<ds:Reference URI="">
```

A requirement when using digital signatures is to express in XPath that address that qualifies all nodes in the referenced content to be included in the calculation of the digital signature hash. For a signature added to a document to remain valid, none of the information can change, nor can any information be added or removed from that portion of the document included in the hash calculation.

One of two such transformation expressions SHOULD be used in the UBL signature extension; choose the appropriate one to meet the objectives of the signature being added to the document. Adding non-signature information to the UBL document will invalidate all signatures already in the extension in either case; when adding more signatures, the behaviour depends on the transformation expression used.

The following transformation element in a digital signature flexibly prevents the signature being invalidated by the subsequent addition of other signatures within the extension:

```
   <Transform
     Algorithm="http://www.w3.org/TR/1999/REC-xpath-19991116">
    <XPath xmlns:sig=
"urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2">
      count(ancestor-or-self::sig:UBLDocumentSignatures |
             here()/ancestor::sig:UBLDocumentSignatures[1]) >
      count(ancestor-or-self::sig:UBLDocumentSignatures)
    </XPath>
   </Transform>
```

The following transformation element in a digital signature is inflexible and thus would be considered a "final" signature to be added to the document. Such a signature will be invalidated by the subsequent addition of other signatures to the document:

```
   <Transform
     Algorithm="http://www.w3.org/TR/1999/REC-xpath-19991116">
```

```
    <XPath xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
      count(ancestor-or-self::ds:Signature |
            here()/ancestor::ds:Signature[1]) >
      count(ancestor-or-self::ds:Signature)
    </XPath>
   </Transform>
```

Multiple separate items of extra-document content (e.g., attachments) or embedded IETF/W3C signature content MAY be included in the signature by adding sibling `<ds:Reference>` elements with other `URI=` attribute values. For example, to countersign another signature in the same UBL document, make a local reference to that signature's unique identifier as in:

```
<ds:Reference URI="#{Id attribute of ds:Signature}">
```

To sign only a portion of a UBL document, an appropriate [XPointer] address for `URI=` SHOULD be used because UBL business object elements do not have attributes of type ID. This requires XPointer awareness on the part of the digital signature tools being used.

7.2. Detached XML Signatures for UBL Documents

This profile supports the application to a UBL document of one or more signatures located outside of the document itself in some other resource.

It is important to note that externally signing a UBL document with a detached signature imposes no requirements on the UBL document itself. Such a signature, in any kind of signature container, can digitally sign the content of a UBL document regardless of whether this is reflected in the document.

If a user knows the document will have a detached conformant IETF/W3C XML digital signature, the user MAY choose to signal in their UBL document that it is so signed. The URI value `urn:oasis:names:specification:ubl:dsig:detached` is reserved to indicate that the detached signature is an IETF/W3C XML digital signature. The URI `urn:oasis:names:specification:ubl:dsig:detached:xades` MAY be used as a value to signal when XAdES is in use. The value is used in the `<cbc:SignatureMethod>` child of `<cac:Signature>`.

If the location of the digital signature is known, the user MAY choose to indicate the location in a `<cbc:URI>` child element of a `<cac:ExternalReference>` child element of a `<cac:DigitalSignatureAttachment>` element.

A complete example of a `<cac:Signature>` business object in the UBL instance is:

```
 <cac:Signature>
   <cbc:ID>urn:oasis:names:specification:ubl:signature:Invoice</cbc:ID>
   <cbc:SignatureMethod
>urn:oasis:names:specification:ubl:dsig:detached</cbc:SignatureMethod>
   <cac:SignatoryParty>
     <cac:PartyIdentification>
       <cbc:ID>MyParty</cbc:ID>
     </cac:PartyIdentification>
   </cac:SignatoryParty>
   <cac:DigitalSignatureAttachment>
     <cac:ExternalReference>
       <cbc:URI>sigFile.xml</cbc:URI>
     </cac:ExternalReference>
   </cac:DigitalSignatureAttachment>
 </cac:Signature>
```

Note

A document with multiple detached signatures is simply a document that is co-signed. By the appropriate use of the <ds:Reference> element pointing to the UBL document from a detached signature file, all such signatures are signing the content of the document but not each other. A *countersigning* document signature, on the other hand, signs signatures already created for and external to or present in the document at the time it is countersigned. A digital countersignature <ds:Signature>, which may be located internal to the UBL document or in an external file, includes additional <ds:Reference> elements, each pointing either to the <ds:Signature> element or <ds:SignatureValue> element child of the signature being signed. In the first case, where the signature is detached, the <ds:Reference> element points to the external file for that signature; in the second case, where the signature is enveloped, the <ds:Reference> element points to the Id= value of either the <ds:Signature> or <ds:SignatureValue> element for that signature.

Note

The XAdES specification supports an alternative countersignature approach where a <ds:Signature> element pointing to the countersigned signature's <ds:SignatureValue> is embedded in the <ds:Object> of the countersigning signature. The inclusion of an alternative method in this specification does not prohibit this approach.

7.2.1. Digital Signature Transformation (Detached Signatures)

The content to be signed is addressed in the URI= attribute of <ds:Reference>:

```
<ds:Reference URI="myInvoice.xml">
```

An option when using detached digital signatures is to express in XPath that address that qualifies all nodes in the referenced content to be included in the calculation of the digital signature hash. For a signature calculated for a document to remain valid, none of the signed information can change, nor can any information be added or removed from that portion of the document included in the hash calculation.

Consider the need to create a detached signature for a UBL file in which there already exists an enveloped signature. The following transformation element in a digital signature flexibly prevents the signature being invalidated by the subsequent addition of any signatures using the enveloped profile within the extension of the document being signed:

```
   <Transform
     Algorithm="http://www.w3.org/TR/1999/REC-xpath-19991116">
    <XPath xmlns:sig=
"urn:oasis:names:specification:ubl:schema:xsd:CommonSignatureComponents-2">
      count(ancestor-or-self::sig:UBLDocumentSignatures)=0
    </XPath>
   </Transform>
```

A non-final transformation algorithm used in the detached signature signs all content outside of any enveloped signatures in the UBL document. When the UBL document does not already have an enveloped signature, one cannot be added without invalidating the detached signature. In effect, the entire document has been signed and cannot change, but the addition of the scaffolding for a signature constitutes a change. However, when the UBL document already has an enveloped signature, other signatures can be added without invalidating the detached signature, because the scaffolding doesn't change when other signatures are added within the existing scaffolding; the non-final transformation algorithm does not include the signatures found in the existing scaffolding. When there is no preexisting enveloped signature, the entire document must be signed in the detached signature.

To sign only a portion of a UBL document, an appropriate [XPointer] address SHOULD be used because UBL business object elements do not have attributes of type ID. This requires XPointer awareness on the part of the digital signature tools being used.

8. Conformance

Claiming syntax conformance to the enveloped profile of this specification requires:

- the schema-valid expression of a UBL extension when the UBL Signature apex element is the apex of the extension;
- the `<ext:Extension>` element is present in the UBL extension and has either `urn:oasis:names:specification:ubl:dsig:enveloped` or `urn:oasis:names:specification:ubl:dsig:enveloped:xades` as its value;
- the value in all uses of `<sbc:ReferencedSignatureID>`, when present, correlates to a corresponding `<cbc:ID>` element of a `<cac:Signature>` element in the same instance; and
- the `<cbc:SignatureMethod>` element, when present, of signature business objects whose signatures are in the UBL extension has either `urn:oasis:names:specification:ubl:dsig:enveloped` or `urn:oasis:names:specification:ubl:dsig:enveloped:xades` as its value.

Claiming processing conformance to the enveloped profile of this specification requires the conformant processing of all contained `<ds:Signature>` elements per [XMLDSig].

Claiming syntax conformance to the detached profile of this specification requires that the `<cbc:SignatureMethod>` element, when present, of signature business objects whose signatures are outside of the UBL document has either `urn:oasis:names:specification:ubl:dsig:detached` or `urn:oasis:names:specification:ubl:dsig:detached:xades` as its value.

8.1. XAdES Conformance

When conformance to XAdES in a UBL document is chosen, this specification requires the valid expression and processing of the XAdES syntax found in an XMLDSig per [XAdES].

Appendix A. Acknowledgments (Non-Normative)

The following OASIS members have participated in the creation of this specification and are gratefully acknowledged.

- Iñigo Barreira, iZenpe S.A., ETSI/ESI member
- Oriol Bausà Peris, Invinet Sistemes 2003, S.L.
- Andrea Caccia, Associazione Italiana Tesorieri d'Impresa, ETSI/ESI member
- Roberto Cisternino, JAVEST
- Juan Carlos Cruellas, Centre d'aplicacions avançades d'Internet (UPC), ETSI/ESI member
- G. Ken Holman, Crane Softwrights Ltd.
- Julián Inza, Albalia Interactiva