PEPPOL
PAN-EUROPEAN PUBLIC PROCUREMENT ONLINE

cip competitiveness and innovation framework programme 2007-2013

DELIVERABLE

Project Acronym: PEPPOL

Grant Agreement number: 224974

Project Title: Pan-European Public Procurement Online

PEPPOL Deliverable D1.3 Demonstrator and functional Specifications for Cross-Border Use of eSignatures in Public Procurement

Part 5: XKMS v2 Interface Specification

Profiling and Extensions Specification

Revision: 2.2

Authors:

Germany: bremen online services

Norway: Difi

Italy: InfoCamere, InfoCert

France: ADETEF, DILA, Lex Persona, ANSSI, Esteral Consulting

Greece: University of Piraeus

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	X
C	Confidential, only for members of the consortium and the Commission Services	

Revision History

Revision	Date	Author	Organisation	Description
1.0	2009/02/11			Complete version of D1.1 for internal quality assurance.
1.1	2009/02/27			D1.1 submitted to PEPPOL project management, approved with comments at project management meeting 2009/03/27.
1.2	2009/04/30			D1.1 for publication, updated according to comments.
1.3	2009/11/06			Formal update of D1.1 after EC approval
1.8	2010/09/22			Complete D1.3 version edited from D1.1 part 1. For internal quality assurance.
1.9	2010/09/30			D1.3 submitted to PEPPOL project operating office (POO) for approval.
1.9.5	2010/11/05			D1.3 ready for publication, updated according to comments from POO. Uploaded for EC approval.
1.9.6	2011/01/19			Correction of description of ResponderDetailsType according schema; section 7 : actual schema included
1.9.7	2011/07/06			Section 3.1.4, prefix for ID values now "_"
2.0	2010/07/15			Formal update after EC approval.
2.1	2011/08/30			Implementation of EC recommendations.
2.2	2011/10/25			3.1.1. Detailed need of request signing 4.2, 5.1: Textual clarification for request, ChainingTo General: sharpening terms "TSL" and "TL" Finalise for hand over

Statement of originality

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Statement of copyright

Table of Contents

1 Summary and Structure of Document

1.1 Scope and Structure of Deliverable D1.3

This document is a part of the multi-part deliverable D1.3 "Functional Specifications for Cross-Border Use of eSignatures in Public Procurement" issued by the PEPPOL[1] (Pan-European Public Procurement On-Line) project. PEPPOL is a 4-year (May 2008 – end April 2012[2]) large scale pilot under the CIP (Competitiveness and Innovation Programme) initiative of the European Commission. D1.3 is an updated version of the deliverable D1.1 "Requirements for Use of Signatures in the Procurement Processes" [PEPPOL-D1.1].

D1.3 consists of the following documents:

Part 1: Background and Scope

(Part 2: Not included – was the D1.1 part on E-tendering Pilot Specifications)

Part 3: Signature Policies

Part 4: Architecture and Trust Models

Part 5: XKMS v2 Interface Specification

Part 6: OASIS DSS Interface Specification

Part 7: eID and eSignature Quality Classification

The D1.3 deliverable is the second version of **functional specifications** for cross-border interoperability of e-signatures in Europe. The specifications are specifically targeted at cross-border public procurement, the topic of PEPPOL. However, a successful solution should be applicable also to other application areas in need of e-signature interoperability.

Signature interoperability in PEPPOL focuses on verification of e-signatures and their associated eIDs. Interoperability of signing solutions is not handled as it is assumed that all actors are capable of signing documents within their corporate infrastructure.

The specifications in deliverable D1.1 has guided the implementation and testing of e-signature interoperability solutions in PEPPOL. In the course of this work, the specifications have by necessity evolved, leading to the revised version published in this deliverable D1.3. These are the specifications for the solutions used for the e-signature interoperability pilots in PEPPOL [PEPPOL-D1.2] in the period 1st November 2010 to 30th April 2012.

The specifications are publicly available and comments from any interested party are most welcome. Note that further evaluation of the specifications of D1.3 is expected as a result of further work in PEPPOL and any party using or referring to the specifications must ensure that the latest version is used; contact the PEPPOL project for information.

1.2 Demonstrator Software Components and Documentation

In addition to the specifications in this deliverable D1.3, PEPPOL WP1 provides software components for cross-border validation of e-signatures:

[1] http://www.peppol.eu

[2] Originally, PEPPOL was scheduled for 3 years. The project has been prolonged twice, both times by 6 months.

- PEPPOL XKMS responder component (server side component) according to the specifications of D1.3 part 5 is provided as open source. The software component, source code and documentation are available on Joinup[3],
- A free to use client side component for signature validation, available as a standalone version and a version for integration into other software applications.
- Open source software components for own development (XKMS requester, Report Agent, Verify Agent, Hashing API, System Configuration API)

The software components are used for PEPPOL's pilot demonstrators on e-signature interoperability as described in PEPPOL Deliverable D1.2 [PEPPOL-D1.2]. Attachments A and B to D1.2 provide documentation on respectively the XKMS responder and the validation client, other documentation is published along with the software.

1.3 Scope and Structure of this Document

Cross-border interoperability for verification of e-signatures requires more information than merely an assessment that the signature is valid. Signature validity is just one aspect of signature acceptance, which is governed by the signature policy in force (see D1.3 part 3).

PEPPOL specifies validation services and their interfaces. A validation service must be able to assess and return information related to signature policy adherence, which necessitates a richer interface than merely OCSP or CRL for revocation checking. Two interfaces are specified:

- XKMS v2 for eID certificate validation (this document);
- OASIS DSS for verification of entire, signed documents (part 6 of D1.3).

The W3C "XML Key Management Specification" [XKMS], part "Key Information Service Specification" (X-KISS) has been chosen as standard interface for the validation process of X509-Certificates used for digital signatures and other purposes in the context of PEPPOL.

XKMS defines a service named "XKMS-Responder", which in the case of X-KISS is able the check the validity of X509-Certificates with regard to a given time instant and appropriate operational model – in case of certificates issued by PKI at least according to relevant specifications as defined by the IETF PKIX Working Group[4]. For this scenario, a XKMS-Responder is in the role of kind of a relay:

- accepting certificate validation requests according to the XKMS protocol;
- in case of an unknown certificate issuer mediating request to other XKMS responder instances able to serve the request[5];
- checking certificates and certificate chains locally;
- connecting to issuer CAs using the respective served protocols (OCSP, CRL, LDAP…);
- if available at the responder instance, including assertions on certificate quality and CSP status either as locally configured in the responder instance (see D1.3 part 7) and/or as specified in the Trusted List (TL) entry covering the certificate issuer;
- building up and delivering the validation response with detailed information as defined by the XKMS protocol.

---

[3] Open source software, semantic assets and other interoperability solutions for public administrations, https://joinup.ec.europa.eu/.

[4] Public-Key Infrastructure X.509 Working Group (PKIX-WG) of the Internet Engineering Task Force

5 This feature is especially defined by PEPPOL with regard to be able to reach any known CA in the EU over the initially contacted XKMS-Responder instance, see D1.3 part 4 for an architectural description.

For the sake of interoperability, this document defines restrictions made by PEPPOL to the relevant parts of the XKMS specification in chapter [3].

In addition, the XKMS extension mechanism is used to define sets of optional attributes, which seem to be valuable for already existing implementations of XKMS responders/requestors. As these extensions are seen as Member State (MS) specific requirements, they should optionally be servable on a profile base. Chapter [4] outlines the extensions defined for PEPPOL. An MS may define its own extensions in co-ordination with the PEPPOL WP1 technical subgroup.

It is an assumption of PEPPOL that there will be several XKMS responder instances with different sets of CAs that can be connected directly – one imaginable XKMS Responder landscape could be a model where each member state (MS) operates an XKMS Responder instance covering connectivity to the CAs of this MS. In reality, there might be *n* specialised instances per MS or even instances covering connectivity to CAs located in different MS.

Another assumption is that a certificate validating client connects to one standard XKMS responder of his choice with trust established to this instance, which – in case the certificate issuer is unknown to this instance – contacts other instances on behalf of the client. This scenario leads to the requirement that XKMS responders must be able to mediate requests to other appropriate instances. In addition, trust relationships must be federated when mediating. Chapter [4] outlines in detail these additional requirements that are out of scope of the standard XKMS specification.

Chapter [2] describes conventions and XML namespaces used in this document.

Sufficient knowledge of XKMS and other referenced specifications is assumed for the addressed audience of this document.

1.4 Evolution of this Document and Changes from D1.1

Note: This document, like the other parts of D1.3, continues the version numbers deriving from D1.1.

Since the publishing of D1.1, the following changes have been made to the specification:

- Namespace changed;
- XKMS Extension Schema revised; in particular:
- XKMS Extension Schema, ResponderDetails completed by TL_Identifer and AlgPolicy_Identifier;
- XKMS Extension Schema, ValidationDetails completed by ValidationTimeQueried;
- "RespondWith" URN for OSCP defined in own namespace (Section 3.2).

The following evolution of this document may be envisaged in future versions:

- The specification should be promoted as a standard profile. PEPPOL will consider submission and follow up to ETSI, W3C or OASIS; this process will necessarily lead to changes in specifications.
- The specification could be further aligned with D1.3 part 6 (OASIS DSS); however the OASIS DSS interface is for the time being not being worked on in PEPPOL.
- Changes due to experience gained in PEPPOL and due to comments from external sources must be expected.

1.5 List of Contributors

The following organisations, in alphabetical order, have contributed to Deliverable D1.3:

- **ADETEF, France** http://www.adetef.fr

- **ANSSI, French Network and Information Security Agency, France** http://www.ssi.gouv.fr
- **bos, bremen online services, Germany,** http://www.bos-bremen.de
- **Difi, Agency for Public Management and eGovernment, Norway** http://www.difi.no
- **DILA, Direction de l'Administration Légale et Administrative Of French Prime Minister Office, France** http://www.dila.premier-ministre.gouv.fr
- **Esteral Consulting, France** http://www.esteralconsulting.com
- **InfoCamere, Italy** http://www.infocamere.it
- **InfoCert, Italy** http://www.infocert.it
- **Lex Persona, France** http://www.lex-persona.com
- **University of Piraeus, Greece** http://www.unipi.gr

The following persons (alphabetical ordering for each participating organisation) have contributed to the D1.3 work:

Jörg Apitzsch	bos	Piero Milani	InfoCamere	Alain Ducass	ADETEF
Nils Büngener	bos	Luca Boldrin	InfoCert	Ahmed Yacine	DILA
Mark Horstmann	bos	Daniele Mongiello	InfoCert	François Devoret	Lex Persona
Ralf Lindemann	bos	Lefteris Leontaridis	Univ. Piraeus	Julien Pasquier	Lex Persona
Dr Jan Pelz	bos	Dr Andriana Prentza	Univ. Piraeus	Sébastien Herniote	ANSSI
Lars Thölken	bos	Alain Esterle	Esteral Cons.	Jon Ølnes (editor)	Difi

D1.3 is a revised version of D1.1. The D1.3 team acknowledges the contributions of organisations and persons that helped producing D1.1 but are no longer active in PEPPOL's e-signature work. These are not listed above; please refer to D1.1 for the names.

2 Document Conventions

2.1 Notational Conventions

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

This specification uses the following syntax to define normative outlines for messages:

- The syntax appears as an XML instance, but values in *italics* indicate data types instead of values.
- Characters are appended to elements and attributes to indicate cardinality:
  - "?" (0 or 1)
  - "*" (0 or more)
  - "+" (1 or more)
- The character "|" is used to indicate a choice between alternatives.
- The characters "(" and ")" are used to indicate that contained items are to be treated as a group with respect to cardinality or choice.
- An ellipsis (i.e. "...“) indicates a point of extensibility that allows other child or attributes content specified in this document. Additional children elements and/or attributes MAY be added at the indicated extension points but they MUST NOT contradict the semantics of the parent and/or owner, respectively. If an extension is not recognised it SHOULD be ignored.
- XML namespace prefixes (see chapter 2.2) are used to indicate the namespace of the element being defined.

Elements and Attributes defined by this specification are referred to in the text of this document using [XPATH 1.0] expressions. Extensibility points are referred to using an extended version of this syntax:

- An element extensibility point is referred to using {any} in place of the element name. This indicates that any element name can be used, from any namespace other than the **`xkms:`** or **`xkmsEU:`** namespaces.
- An attribute extensibility point is referred to using @{any} in place of the attribute name. This indicates that any attribute name from any namespace can be used.

For those parts of this specification where referenced specifications are profiled, normative statements of requirements are presented in the following manner:

Rnnnn - *Statement text here*

Where "nnnn" is replaced by a number that is unique among the requirements in this document, thereby forming a unique requirement identifier.

If needed for clarification, indentation "(gen)" is used, when a software instance is required to support generation of a certain requirement or XML Infoset, indentation "(proc)" if processing is required; "(gen/proc)" if both.

2.2 XML Namespaces

Following XML namespaces are referenced:

Prefix	XML Namespace	Specification
ds	**http://www.w3.org/2000/09/xmldsig#**	[XMLDSIG]
isocc	**http://www.tm-xml.org/XMLSchema/common**	[ISOCC]
tsl	**http://uri.etsi.org/02231/v2#**	[ETSI102231]
xades	**http://uri.etsi.org/01903/v1.3.2#**	[XAdES]
xkms	**http://www.w3.org/2002/03/xkms#**	[XKMS]
xkmsEU	**http://uri.peppol.eu/xkmsExt/v2.3#**	This document
xs	**http://www.w3.org/2001/XMLSchema**	[XMLSchema]

Table 1: Referenced Namespaces

The namespace chosen for the XKMS extension outlined in this document is preliminary. It is intended to align details with other large scale pilot projects which may use outcomes of PEPPOL.

3 XKMS 2.0 Restrictions

For XKMS in general and X-KISS in detail, definitions of [XKMS] apply; only deviations from the standard are outlined here.

3.1 General

R0100 - For simplification of processing and implementation, conformant XKMS requestors (gen) and responders (proc) MUST use synchronous request/response processing as defined in ([XKMS], chapter 2.4.1). For the PEPPOL pilot, asynchronous processing MUST NOT be used.[6]

R0110 - For optimisation reasons, conformant XKMS requestors (gen/proc) and responders (gen/proc) MUST support compound request/responses as defined in ([XKMS], chapter 3.4).

R0110 applies in conjunction with

R0120 - Conformant XKMS implementations MUST support the validate service on base of the XML infosets **`xkms:ValidateRequest`** and **`xkms:ValidateResult`** ([XKMS], chapters 4.2 and 5.3).

These restrictions lead to the following schemas of XKMS request respective response which MUST be supported:

```
<!-- CompoundRequest -->
   <element name="CompoundRequest" type="xkms:CompoundRequestType"/>
   <complexType name="CompoundRequestType">
      <complexContent>
         <extension base="xkms:RequestAbstractType">
            <choice maxOccurs="unbounded">
               <element ref="xkms:ValidateRequest"/>
            </choice>
         </extension>
      </complexContent>
   </complexType>
<!-- /CompoundRequest -->
```

```
<!-- CompoundResult -->
   <element name="CompoundResult" type="xkms:CompoundResultType"/>
   <complexType name="CompoundResultType">
      <complexContent>
         <extension base="xkms:ResultType">
            <choice minOccurs="0" maxOccurs="unbounded">
               <element ref="xkms:ValidateResult"/>
            </choice>
         </extension>
      </complexContent>
   </complexType>
<!-- /CompoundResult -->
```

[6] Support of asynchronous processing is foreseen for a future version. For the pilot version, XKMS clients should be aware that XKMS responders used in the PEPPOL infrastructure are not obligated to support asynchronous requests.

3.1.1 Processing Requirements

R0130 - XKMS responders conformant to this profiling MUST try to obtain all missing data needed for the validation process from the underlying PKI service and hence MUST provide interfaces to underlying PKIs (both is marked optional in the XKMS specification). The *validation processing* MUST at least follow the PKIX-model as summarised in [COMMPKI], Part 5: Certificate Path Validation.

For *CA access*, XKMS responders MUST support the interfaces as summarised in [COMMPKI] , Part 4: Operational Protocols.

3.1.2 XKMS Message Transport

R0140 - XKMS MUST be bound to SOAP 1.2 over HTTPS as defined as one option in the XKMS bindings specification [XKMSBIND].

3.1.3 Message Signing Requirements and Processing Recommendations

Following message signing requirements are mandatory for XKMS responder instances of the PEPPOL infrastructure. The concrete precautions for authentication between a relying end entity and it's trusted XKMS responder may be established by other means out of scope of this specification. E.g., end entities may use a relay to set up the XKMS messages described here, thus acting as an XKMS requestor "on behalf". The first XKMS responder contacted by either the end entity itself or the relay used MUST authenticate the **`/xkms:CompoundRequest`** using the applied signature element and certificate; this may e.g. be done on base of certificate list outlining permissible requestors.

R0150 - For integrity protection and authentication reasons, XKMS messages MUST be signed by the respective producer. Implementations MUST ensure that all the bytes in the XKMS messages be included in hashing and in the resulting signature value of the message (see [XKMS], chapter 3.1.1); message consumers MUST validate the signatures. For compound requests and responses, the **`/xkms:CompoundRequest/ds:Signature`** respective **`/xkms:CompoundResponse/ds:Signature`** element MUST be generated, the inner …**`/ds:Signature`** elements of the contained …**`/xkms:ValidateRequest`** respective …**`/xkms:ValidateResult`** containers SHOULD NOT be generated in addition. The latter MUST be generated if simple requests/responses are used, which are not enveloped in a compound request respective response.

R0160 - XKMS signatures MUST be generated using X509 certificates, which MUST be embedded in the **`ds:Signature`** elements according to [XMLDSIG].

XKMS responders MAY decide service processing or denial on base of known the requestor certificates, which in addition may be taken for accounting issues. Responder instances MUST publish their policies concerning the regulations in effect for these issues.

For XKMS requestors, the signing certificate of the used responder is in the role of a trust anchor. Requestors MUST NOT consume response messages, for which untrusted or unknown certificates were used for message signing.

3.1.4 Id Attributes, Identifying Requests and Responses

R0170 - Following [XMLSchema], Id attributes used in a XML Infoset instance MUST have unique values. To fulfil this requirement, Id attribute values SHOULD be generated according to IETF RFC "A Universally Unique Identifier (UUID) URN Namespace" [RFC4122], whereby this value SHOULD be preceded by the underscore character "_"[7].

[7] Values generated following [RFC4122] may have leading characters which violate the production rules of the **`xs:ID`** type

R0180 - To enable requestor-side correlation of requests and responses, the values of the request `@Id` attributes of elements `/xkms:CompoundRequest` and `/xkms:ValidateRequest` MUST be copied to the corresponding `@RequestId` attributes of the `/xkms:CompoundResult` and `/xkms:ValidateResult`.[8]

3.2 ValidateRequest

R0200 - `xkms:ValidateRequest` is an extension of `xkms:RequestAbstractType`, which itself is an extension of `xkms:MessageAbstractType`. The extensions defined by `xkms:RequestAbstractType` are defined optional. Following elements and attributes of these extensions MUST NOT be used, as they are meaningful only in the context of asynchronous processing:

`@OriginalRequestId, @ResponseLimit, xkms:ResponseMechanism, xkms:PendingNotification`

R0210 - The `xkms:RespondWith` extension of `xkms:RequestAbstractType` SHOULD be used to indicate the base PKI validation data required in the response. `xkms:RespondWith` is based on the URI enumeration simple type `xkms:RespondWithEnum`. Following table outlines the meaningful choices in this context, which MUST be understood by conformant XKMS responders. Other values MAY be used[9], for which standard XKMS responders are not obliged to support them:

RespondWith URI	Meaning
`http://www.w3.org/2002/03/xkms#X509Cert`	Return certificate (default behaviour, if no element `xkms:RespondWith` present in the request)
`http://www.w3.org/2002/03/xkms#X509Chain`	Return certificate chain build by responder
`http://www.w3.org/2002/03/xkms#X509CRL`	Return CRL acquired by responder
`http://uri.peppol.eu/xkmsExt/v2#OCSP`[10]	Return acquired OCSP response for validated certificate (not multiple OCSPs of the whole chain!)

Table 2: RespondWith URIs of the XKMS standard set to be supported

[8] [XKMS] outlines the `@RequestId` as on optional attribute

[9] This is covered by the XKMS schema, as the underlying type is a `xs:union` of defined URI enumerations and `xs:anyURI`

[10] There is no enumeration defined for OCSP in [XKMS], thus we define an new one in the namespace xkmsEU:.

R0220 - Extended response information can be requested by following additional URIs; XKMS responders used in the PEPPOL context SHOULD support this functionality:

RespondWith URI	Meaning
`http://uri.peppol.eu/xkmsExt/v2`	Return extended response as defined in this document. Request for further child elements: see next table entries.
`http://uri.etsi.org/02231/v2#ServiceInformation`[11]	Return information as defined by ETSI TS 102 231 for the according `tsl:ServiceInformation` element (contains - besides other details - the quality of certificate and status of issuing CSP)
`http://uri.peppol.eu/xkmsExt/v2#eIDQuality`	Return quality of certificate and status of issuing CSP according to the rating defined in chapter [5.2] for eIDQuality (further detailed in PEPPOL D1.3 Part 7 "eID and eSignature Quality Classification"). This is the default behaviour, if no element `xkms:RespondWith` or the foregoing `xkms:RespondWith` URI pointing to the TL alternative present in the request)
`http://uri.peppol.eu/xkmsExt/v2#OCSPNoCache`	Attention: If not provided, XKMS responder MAY use cached OCSP response for validation[12]
`http://uri.peppol.eu/xkmsExt/v2#ValidationDetails`	Details on validation process to be delivered

Table 3: RespondWith URIs that SHOULD be supported for extended responses

If a XKMS responder instance does not understand one of these RespondWith URIs, processing MUST continue and an entry in of <xkmsEU:ErrorExtension> MUST be generated:

```
<xkmsEU:ErrorExtension>
  <xkmsEU:Reason>
     http://uri.peppol.eu/xkmsExt/v2#reasonNotUnderstood
  </xkmsEU:Reason>
  <xkmsEU:Detail>
     copy of RespondWith URI not understood to be placed here
  </xkmsEU:Detail>
</xkmsEU:ErrorExtension>
```

R0230 - `xkms:ValidateRequest` carries an element `xkms:QueryKeyBinding`. It's according type `xkms:QueryKeyBindingType` is an extension of

[11] Both `http://uri.etsi.org/02231/v2#ServiceInformation` and `http://uri.peppol.eu/xkmsExt/v2#eIDQuality` URI may be given for `xkms:RespondWith`

[12] OCSP cacheing may be an implementation feature to reduce network latencies

xkms:KeyBindingAbstractType, which in case of a **xkms:ValidateRequest** MUST contain at least the **ds:KeyInfo** element.

```
<!-- KeyBindingAbstractType-->
<complexType name="KeyBindingAbstractType" abstract="true">
  <sequence>
    <element ref="ds:KeyInfo" minOccurs="1"/>
    <element ref="xkms:KeyUsage" minOccurs="0" maxOccurs="3"/>
    <element ref="xkms:UseKeyWith" minOccurs="0"
                          maxOccurs="unbounded"/>
  </sequence>
  <attribute name="Id" type="ID" use="optional"/>
</complexType>
<!-- /KeyBindingAbstractType-->
```

R0240 - **ds:KeyInfo** MUST at least carry the certificate to be validated in **ds:X590Data/ds:X509Certificate**. More information – e.g. certificate chains – MAY be supplied by the requestor. One **xkms**:**ValidateRequest** MUST carry only one end user certificate to be validated; multiple **xkms:ValidateRequest** elements SHOULD be grouped in a **xkms:CompoundRequest**, if validation of more then one certificate is required to be done within one request/response sequence (see R0110 above).

R0250 - **xkms:QueryKeyBinding** carries an optional element **xkms**:**TimeInstant**, the value outlined here is the requested time instant for which the requestor wants to check the certificate validity. In case of verifying digital signatures, as discussed in D1.3 part 3 a requestor has three options: Valid at time of signing, valid at time of first verification, valid at time of verification.

The element **xkms**:**TimeInstant** MUST be used for the "valid at time of signing" option, in which case the value of **xkms**:**TimeInstant** MUST be derived from the signing time instant, if available in the underlying signature.

With the "valid at time of first verification" option, the requestor MAY specify the desired time value in **xkms**:**TimeInstant.** The requestor MAY omit **xkms**:**TimeInstant** requesting verification according to responder's actual server time (see below); the time of this verification will then be the "time of first verification". With the "valid at time of first verification" option, the requestor SHOULD log verification information and refer to this information at later points as an alternative to calling the responder for each verification.

If **xkms**:**TimeInstant** is not supplied in the request, according to [XKMS] the responder has to validate the certificate on base of the responder's actual server time, corresponding to the "valid at time of verification" option.

3.3 ValidateResult

For the standard part of **xkms:ValidateResult**, following detailing is made here:

R0300 - The **xkms:ValidateResult** MUST carry elements **xkms:KeyBinding**, containg the validate certificate(s) itself the sub-element **ds:keyinfo/ds:X509Data/ds:X509Certificate**. In addition, the according gathered validation information must be included in the sub-elements of ds:keyinfo/**ds:X509Data**. If validation has been performed on base of CRL, this CRL MUST be provided in the sub-element **ds:keyinfo/ds:X509Data/ds:X509CRL**, in case of OCSP this OCSP response MUST be carried in **ds:X509Data** extension **##other** foreseen by [XMLDSIG]. Format and namespace of this extension MUST follow the ETSI XAdES specification [XAdES] for **xades:OCSPValuesType**; the element name has to be **xades:RevocationValues** containing **xades:OCSPValues** as only constituent of the sequence defined for **xades:RevocationValues.**

Following applies concerning message extension:

R0310 - If a request carries an extension with a namespace known by the contacted XKMS responder instance, the request message extension MUST be processed according to the rules defined for this extension set. Processing MAY lead to a corresponding message extension in the response.

If an extension contained in the request is bound to a namespace not known by the responder instance, processing MUST proceed ignoring this request extension; the generated response MUST outline this fact by setting the **`@ResultMinor`** attribute value of the response to **`"http://www.w3.org/2002/03/xkms#OptionalElementNotSupported"`**, even if the **`@ResultMinor`** attribute value may be set to **`"http://www.w3.org/2002/03/xkms#Success".`** In case different values for these attributes should be generated during processing covered by the XKMS standard part, these values dominate.

4 Mediating XKMS Requests and Responses

4.1 Preconditions

R1000 - If a XKMS responder instance fails to process a validate request because the issuer of the certificate to be validated is not known here, it MUST be able to forward the validate request to another instance able to process the request. It is an implementation detail how the appropriate routing information is made available to the forwarding responder. This information SHOULD be gathered on base of Trusted Lists (TL) (D1.3 part 4).

R1010 - Trust MUST been established between the forwarding XKMS responder and validate request destination on base of known signature certificates used for message signing by the involved XKMS responder instances. Again, TLs SHOULD serve as the anchor to establish trust.

R1020 - For the synchronous processing as restricted for this version (see R0100), all instances involved in a mediation scenario MUST NOT close network connections on application level until response delivery is acknowledged by the respective requesting instance.

4.2 Request Forwarding

R1030 - Before request forwarding, the original request has to be modified:

The **`@service`** attribute of the request message MUST set to the value of the URI to which the XKMS request is directed now.

The **`@Id`** attribute of the request message MUST reset to a newly generated value according to chapter [3.1.4]; the original value MUST be retained for further processing.

An element **`xkmsEU:RequestingNodeChain`** MUST be added to a **`xkms:ValidateRequest`** by the first XKMS responder performing a request forwarding.

An element **`xkmsEU:RequestingNode`** must be added to the original request in the sequence **`xkmsEU:RequestingNodeChain`**, outlining the URL of the forwarding XKMS responder instance (see chapter [5.1] for details).

A new …**`/ds:Signature`** element MUST be provided, the forwarding instance MUST resign the request message after eliminating the existing …**`/ds:Signature`** element.

4.3 Result Delivery

R1040 - The responder instance the XKMS request has been directed to MUST deliver the result message to the mediating responder instance.

R1050 - The mediating responder instance MUST verify the result message signature. In case of fault or missing trust to the result messages signature, this message MUST be discarded and a new result messages MUST be generated with following fault information attributes:

`@ResultMajor=http://www.w3.org/2002/03/xkms#Receiver`

`@ResultMinor=`

`http://uri.peppol.eu/xkmsExt/v2#reasonTrustViolation`

`@Service` MUST carry the URI of the responder instance the corresponding request message was directed to.

R1060 - Before the mediating responder is re-signing the result message (see R1070) and forwarding it to the initial requestor, the result message attribute

`@RequestId` MUST be set to the value of the initial request

(which MUST have been retained by the mediating instance, see R1030).

R1070 - To provide trust establishment for the initial requestor, a new …**`/ds:Signature`** element MUST be provided¸ mediating instance MUST resign the result message after eliminating the existing …**`/ds:Signature`** element.

R1080 - The mediating responder MUST NOT apply any other changes on the result message.

5 XKMS Extensions defined for PEPPOL

For XKMS messages an abstract extension point **`xkms:MessageExtension`** is foreseen to carry additional information. German regulations require detailed information on certificate quality and validity status as well as the validation process itself. Thus, a **`/xkms:ValidateResult`** SHOULD contain an extension block **`/xkmsEU:ValidateResultExtEU`** as defined here and if requested by a message extension in the respective validate request.

If a XKMS responder instance forwards a **`xkms:ValidateRequest`** to another responder instance, an extension block **`/xkmsEU:ValidateReqestExtEU`** as defined in following subchapter MUST be build up as extension block in the **`xkms:ValidateRequest`**.

5.1 Extension for Validate Request

Following **`xkms:MessageExtension`** is defined for the case of forwarding requests; other extensions going beyond the standard **`xkms:ValidateRequest`** are defined above with R0220 for **`xkms:RespondWith`** URIs.

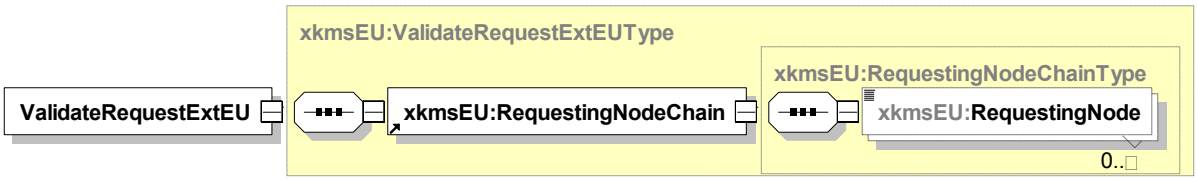

Figure 1: Validate Request extension scheme overview

```
<xkmsEU:ValidateRequestExtEU> ?

   <xkmsEU:RequestingNodeChain>
      <xkmsEU:RequestingNode>
          tsl:NonEmptyURIType
      </xkmsEU:RequestingNode> *
   <xkmsEU:RequestingNodeChain>

</xkmsEU:ValidateRequestExtEU>
```

Description of elements and attributes in the schema overview above:

`/xkmsEU:ValidateRequestExtEU ?`

Container element carrying all child elements explained below. This element MUST be added to a **`xkms:ValidateRequest`** by the first XKMS responder performing a request forwarding.

`/xkmsEU:RequestingNodeChain`

If the extension element is present, this enveloping element contains the URL's of all XKMS responder nodes which have forwarded this request.

`/xkmsEU:RequestingNodeChain/xkmsEU:RequestingNode *`

In case of forwarding a request, the XKMS responder has to add his service supply point in a form of a non-empty URI here. This information MUST be analysed by subsequent responder instances to detect possible chaining loops.

In case such a situation is detected, processing MUST be aborted with an XKMS error with **`@ResultMajor=http://www.w3.org/2002/03/xkms#Sender`**. An

xkmsEU:ErrorExtension MUST be generated, containing a reason URI of **xkmsEU:reasonResponderChainLoop**

5.2 Extension for Validate Result

Extended validation information is defined for

- the quality of a certificate and the issuing CSP according to the PEPPOL WP1 specifications (D1.3 part 7);
- as an alternative (or in addition) on request the ETSI TS 102 231 **tsl:ServiceInformation** element, containing the CSP quality rating defined for the EC TL;
- details for the validation processing done by a XKMS responder instance;
- details about the XKMS responder itself.

This is complemented by possible fault information concerning the processing of the extensions.

An overview is given in the following figure:

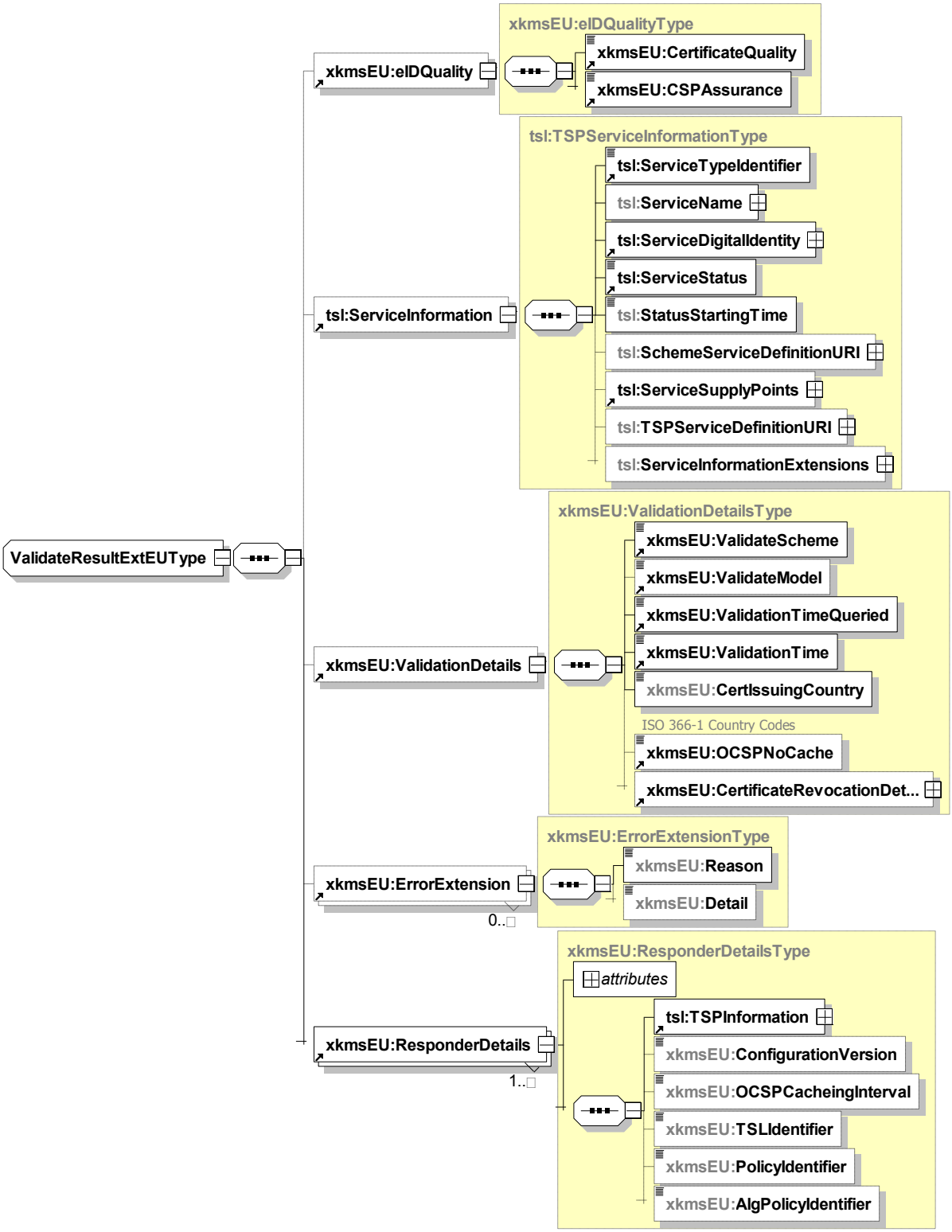

Figure 2: Validate Result extension scheme overview

Syntax for the **xkmsEU:ValidateResultExtEU** element:

```
<xkmsEU:ValidateResultExtEU>

   <xkmsEU:eIDQuality>
      <xkmsEU:CertificateQuality>
         http://uri.peppol.eu/xkmsExt/v2#certqualityUnknown |
         http://uri.peppol.eu/xkmsExt/v2#certqualityLow |
         http://uri.peppol.eu/xkmsExt/v2#certqualityLCP |
         http://uri.peppol.eu/xkmsExt/v2#certqualityNCP |
         http://uri.peppol.eu/xkmsExt/v2#certqualityNCPPLUS |
         http://uri.peppol.eu/xkmsExt/v2#certqualityQCP |
         http://uri.peppol.eu/xkmsExt/v2#certqualityQCPPLUS
      </xkmsEU:CertificateQuality> |

      <tsl:ServiceInformation>
            <!—TSP service information as specified for ETSI TSL -->
      </tsl:ServiceInformation>

      <xkmsEU:CSPAssurance>
         http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceNone |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 IndependentDocumentReview |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 InternalComplianceAudit |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 SupervisionWithComplianceAudit |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 ExternalComplianceAudit |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 ExternalComplianceAuditCertified  |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 SupervisionWithExternalComplianceAudit  |
         http://uri.peppol.eu/xkmsExt/v2#CSPAssurance
                                 AccreditationWithExternalComplianceAudit
      </xkmsEU:CSPAssurance>
   </xkmsEU:eIDQuality> ?

   <xkmsEU:ValidationDetails>
      <xkmsEU:ValidateScheme>
         http://uri.peppol.eu/xkmsExt/v2#valSchemeLOCAL |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeOCSP |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeOCSPCommonPKI |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeCRL |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeCRLLDAP |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeLDAP |
         http://uri.peppol.eu/xkmsExt/v2#valSchemeNONE
      </xkmsEU:ValidateScheme>

      <xkmsEU:ValidateModel>
         http://uri.peppol.eu/xkmsExt/v2#valModelPKIX |
         http://uri.peppol.eu/xkmsExt/v2#valModelChain |
         http://uri.peppol.eu/xkmsExt/v2#valModelEscapeRoute |
      </xkmsEU:ValidateModel>      ?

      <xkmsEU:ValidationTimeQueried> xs:dateTime
      </xkmsEU:ValidationTimeQueried>
      <xkmsEU:ValidationTime> xs:dateTime
      </xkmsEU:ValidationTime>
```

```
      <xkmsEU:CertIssuingCountry>
         isocc:ISOCountyCodeType
      </xkmsEU:CertIssuingCountry>

      <xkmsEU:CertificateRevocationDetails>
         <xkmsEU:RevocationTimeInstant> xs:dateTime
         </xkmsEU:RevocationTimeInstant>
         <xkmsEU:RevocationReason>
               http://uri.peppol.eu/xkmsExt/v2#reasonUnspecified |
               http://uri.peppol.eu/xkmsExt/v2#reasonKeyCompromise |
               http://uri.peppol.eu/xkmsExt/v2#reasonCACompromise |
               http://uri.peppol.eu/xkmsExt/v2#reasonAffiliationChanged |
               http://uri.peppol.eu/xkmsExt/v2#reasonSuperseded |
               http://uri.peppol.eu/xkmsExt/v2#reasonCessationOfOperation|
               http://uri.peppol.eu/xkmsExt/v2#reasonCertificateHold |
               http://uri.peppol.eu/xkmsExt/v2#reasonRemoveFromCRL |
               http://uri.peppol.eu/xkmsExt/v2#reasonPrivilegeWithdrawn |
               http://uri.peppol.eu/xkmsExt/v2#reasonAACompromise |
               http://uri.peppol.eu/xkmsExt/v2#reasonNone
         </xkmsEU:RevocationReason>
      </xkmsEU:CertificateRevocationDetails>     ?

      <xmksEU:OCSPNoCache> xs:Boolean </xkmsEU:OCSPNoCache>
   </xkmsEI:ValidationDetails> ?

   <xkmsEU:ResponderDetails chainingTo="tsl:NonEmptyURIType" ? >
      <tsl:TSPInformation> tsl:TSPInfomationType </tsl:TSPInformation>
      <xkmsEU:ConfigurationVersion> xs:string
               </xkmsEU:ConfigurationVersion> ?
      <xkmsEU:OCSPCacheingInterval> xs:duration
                  </xkmsEU:OCSPCacheingInterval> ?
      <xkmsEU:TSLIdentifier> tsl:NonEmtyURIType </xkmsEU:TSLIdentifier> ?
      <xkmsEU:PolicyIdentifier> xs:anyURI </xkmsEU:PolicyIdentifier> ?
      <xkmsEU:AlgPolicyIdentifier> tsl:NonEpmtyURIType
                           </xkmsEU:AlgPolicyIdentifier> ?
   </xkmsEU:ResponderDetails>

   <xkmsEU:ErrorExtension
      <xkmsEU:Reason=
         http://uri.peppol.eu/xkmsExt/v2#reasonOpaqueClientDataTooLong |
         http://uri.peppol.eu/xkmsExt/v2#reasonTrustCenterNotReachable |
         http://uri.peppol.eu/xkmsExt/v2#reasonWrongCertificateFormat |
         http://uri.peppol.eu/xkmsExt/v2#reasonWrongTimeInstant |
         http://uri.peppol.eu/xkmsExt/v2#reasonUnknownCA |
         http://uri.peppol.eu/xkmsExt/v2#reasonSignatureKeyTooShort  |
         http://uri.peppol.eu/xkmsExt/v2#reason
                                       NextResponderInChainNotReached |
         http://uri.peppol.eu/xkmsExt/v2#reasonResponderChainLoop  |
         http://uri.peppol.eu/xkmsExt/v2#reasonUnknown  |
         http://uri.peppol.eu/xkmsExt/v2#reasonNotUnderstood   |
         http://uri.peppol.eu/xkmsExt/v2#reason
                                       RevocationStatusNoLongerProvided  |
         http://uri.peppol.eu/xkmsExt/v2#reasonCertQualityNotConsistent
      </xkmsEU:Reason>
      <xkmsEU:Detail> xs:string </xkmsEU:Detail> ?
   </xkmsEU:ErrorExtension>     *

</xkmsEU:ValidateResultExtEU> ?
```

Description of elements and attributes in the schema overview above:

/xkmsEU:ValidateResultExtEU

Container element carrying all child elements explained below.

/tsl:ServiceInformation ?

Optional container element carrying assurances on certificate quality and issuing CSP TSP according to ETSI TSL specification. For further details see [ETSI102231], chapter 5.5. MUST be present if certificate validation could be processed, and MAY be present if certificate validation could not be processed, if this information was explicitly requested by a **xkms:RespondWith** value of **http://uri.etsi.org/02231/v2#ServiceInformation.**

/xkmsEU:eIDQuality ?

Optional container element carrying assurances on certificate quality and issuing CSP status. MUST be present with child elements if certificate validation could be processed, and MAY be present if certificate validation could not be processed, if this information was explicitly requested by a **xkms:RespondWith** value of **http://uri.peppol.eu/xkmsExt/v2#edIDQuality.**

/xkmsEU:eIDQuality/xkmsEU:CertificateQuality

Element of type **xs:anyURI** indicating the certificate quality. All values in the table below carry the prefix **http://uri.peppol.eu/xkmsExt/v2#certquality,** which is omitted here for readability. This table corresponds to D1.3 Part 7, "eID and eSignature Quality Classification", chapter 3.2.1. For further details, see ETSI specification [ETSI101456], [ETSI102042] referenced in this table.

CertificateQuality URI ending	Meaning
Unknown	Certificate quality can't be determined
Low	Low confidence in certificate but certificate policy exists or quality assessment is possible by other means
LCP	Certificate governed by a Certificate Policy in compliance with the ETSI TS 102 042 standard for LCP or a similar standard
NCP	Certificate governed by a Certificate Policy in compliance with the ETSI TS 102 042 standard for NCP or a similar standard
NCPPLUS	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)
QCP	Certificates governed by a Certificate Policy in compliance with the ETSI TS 101 456 standard for QCP or a similar standard
QCPPLUS	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)

Table 4: Quality of Certificate

/xkmsEU:eIDQuality/xkmsEU:CSPAssurance

Element of type **xs:anyURI** indicating the certificate issuing CSP status according to D1.3 Part 7, "eID and eSignature Quality Classification", chapter 3.2.3. All values in the table below carry the prefix **http://uri.peppol.eu/xkmsExt/v2#CSPAssurance,** which is omitted here for readability.

CSPAssurance URI ending	Meaning
None	Self assessment only
IndependentDocument Review	Statement of compliance issued by an independent, external unit based on document review only
InternalCompliance Audit	Internal audit carried out periodically concludes compliance to applicable requirements
SupervisionWithout ComplianceAudit	CA is supervised by a public, national or international authority according to applicable law to the CA
ExternalCompliance Audit	Audit carried out periodically by external, independent auditor concludes compliance to applicable requirements
ExternalCompliance AuditCertified	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 hierarchy as a result of appropriate assessment
SupervisionWith ExternalCompliance 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
AccreditationWith ExternalCompliance 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

Table 5: CA Independent Assurance

/xkmsEU:ValidationDetails ?

Optional container element carrying details on the certificate validation. MUST be present with child elements if certificate validation could be processed, and MAY be present if certificate validation could not be processed, if this information was explicitly requested by

a **xkms:RespondWith** value of **http://uri.peppol.eu/xkmsExt/v2#ValidationDetails.**

/xkmsEU:ValidationDetails/xkmsEU:ValidateScheme

Element of type **xs:anyURI** indicating the mechanism respective the protocol a certificate was validated. All values in the table below carry the prefix **http://uri.peppol.eu/2009/04/xkmsExt/v2#valScheme,** which is omitted here for readability.

ValidateScheme URI ending	Meaning
LOCAL	Only local checked by responder instance
OCSP	Request to CA OCSP responder following RFC2560 (only performing negative checks)
OCSP-CommonPKI	Request to CA OCSP responder, responder makes positive and negative OCSP check.
CRL	CRL used
CRL_LDAP	CRL and LDAP used
LDAP	Request to CA LDAP certificate directory
NONE	Validate scheme not determined

Table 6: Certificate Validation Schemes

/xkmsEU:ValidationDetails/xkmsEU:ValidateModel ?

Element of type **xs:anyURI** indicating the validation scheme used. All values in the table below carry the prefix **http://uri.peppol.eu/xkmsExt/v2#valModel,** which is omitted here for readability.

ValidateModel URI ending	Validation Process
PKIX	Validation PKIX-conformant (shell-model)
chain	Strict certificate chain validation processing
escapeRoute[13]	Mix of both above as described in [COMMPKI], part 9 "SigG Profile", chapter 6

Table 7: Certificate Validation Models

[13] Foreseen for future realisation, not used in this version.

/xkmsEU:ValidationDetails/xkmsEU:ValidationTimeQueried

Requested time of validation processing; element of type **xs:dateTime**. This time instant MUST be taken from the underlying **xkms:ValidateRequest**, attribute **@Time** of the element **/xkms:QueryKeyBinding/xkms:TimeInstant.**[14]

/xkmsEU:ValidationDetails/xkmsEU:ValidationTime

Time of validation processing; element of type **xs:dateTime**.

/xkmsEU:ValidationDetails/xkmsEU:CertIssuingCountry

The code of the country the certificate was issued; must be of type **isocc:ISOCountryCodeType** as defined in [ISOCC][15].

/xkmsEU:ValidationDetails/xkmsEU:OCSPNoCache ?

Optional element of type **xs:boolean.** MUST be reported as true, if the OSCP response was not taken from the cache.

/xkmsEU:ValidationDetails/xkmsEU:CertificateRevocationDetails ?

Container holding details in case of a certificate revoked status.

/xkmsEU:ValidationDetails/xkmsEU:CertificateRevocationDetails/ xkmsEU:RevocationTimeInstant

Time of revocation; type is **xs:dateTime**.

/xkmsEU:ValidationDetails/xkmsEU:CertificateRevocationDetails/ xkmsEU:RevocationReason

Element of type **xs:anyURI** indicating one of the following revocation reasons outlines in the table below. All values carry the prefix **http://uri.peppol.eu/xkmsExt/v2#revocationReason,** which is omitted here for readability.

RevocationReason URI ending	Meaning
Unspecified	It is unspecified as to why the certificate has been revoked
KeyCompromise	It is known or suspected that the user certificate subject's private key has been compromised
CACompromise	It is known or suspected that the issuer certificate subject's private key has been compromised
AffiliationChanged	The subject's name or other information has changed; certificate is not compromised

[14] To be able to proof the validation time originally queried, it must be outlined in the signed **xkms:ValidateResult**

[15] Actually, the ISO 3166 enumeration does not allow outlining the fact of an unknown issuer country. This should be foreseen for a future update of this specification.

RevocationReason URI ending	Meaning
`Superseded`	Certificate marked as superseded; certificate is not compromised
`CessationOfOperation`	Certificate marked as no longer needed; certificate is not compromised
`CertificateHold`	Certificate has been put on hold; certificate is not compromised
`RemoveFromCRL`	Certificate is withdrawn from CRL, reusable again
`PrivilegeWithdrawn`	A privilege documented in certificate is withdrawn
`AACompromise`	The private key of an Attribute Authority could be or is compromised
`None`	No revocation reason available

Table 8: Certificate Revocation Reasons

`/xkmsEU:ResponderDetails`

This container MUST be present, indicating details of the XKMS responder instance generating this validation result.

`xkmsEU:ResponderDetails/@chainingTo ?`

Optional attribute of type tsl:NonEmptyURIType, in case of chaining a request MUST be provided with the URL of the responder the request is forwarded to.

`/xkmsEU:ResponderDetails/tsl:TSPInformation ?`

Optional element of type **`tsl:TSPInformationType`** carrying information about the responder like name, address and other details as specified in [ETSI102231]. This element SHOULD be included unchanged from the according PEPPOL Public Registry Server (PPRS)[16] entry of this responder.

`/xkmsEU:ResponderDetails/xkmsEU:ConfigurationVersion ?`

Optional element of type **`xs:string`** carrying information about the responders configuration version.[17]

`/xkmsEU:ResponderDetails/xkmsEU:OCSPCacheingInterval ?`

Optional element of type **`xs:duration.`** If a responder uses cacheing for OSCP responses, the cacheing interval time SHOULD be reported here.

`/xkmsEU:ResponderDetails/xkmsEU:TSLIdentifier ?`

[16] PPRS is based on the TSL specification and described in [PEPPOL-1.3.4]

[17] Capabilities of a XKMS responder – i.e. OSCP-responders known by a responder instance - may depend on a concrete configuration version; this information may be helpful when checking for reasons of errors reported by a XKMS responder.

Optional element of type **`tsl:NonEmptyURIType`**, outlining the URI of the Trusted List instance used in this concrete validation process (may be exposed by the EU member states as machine readable XML file or in PDF format).

> **NOTE:** As soon as TL's will be generally made available by the EU member states, this element will be mandatory!
>
> If the certificate issuer cannot be found in the national TL (e.g. in case of non-qualified certificates), this MUST be expressed by the URI
>
> **`http://uri.peppol.eu/xkmsExt/v2#CertIssuerNotInTSL`**

`/xkmsEU:ResponderDetails/xkmsEU:AlgPolicyIdentifier ?`

Optional element of type **`tsl:NonEmptyURIType`**. EU member states may expose regulations on suitability of cryptographic algorithms with regard to underlying time scale online. If such a information was used as base of corresponding assertions made on algorithm strongness rating in the validate result, the value of this element MUST point to the respective document source[18].

`/xkmsEU:ResponderDetails/xkmsEU:Policy_Identifier ?`

Optional element of type **`xs:anyURI`**, pointing to the policy document of this XKMS responder instance.

`/xkmsEU:ErrorExtension *`

This optional element is used to report errors concerning the validation process in the attribute:

`/xkmsEU:ErrorExtension/Reason`

Element of type **`xs:anyURI`** with following possible values; all values carry the prefix **`http://uri.peppol.eu/2009/04/reason#`**, which is omitted here for readability.

ErrorExtension/Reason URI ending	Semantics
`OpaqueClientData TooLong`	Length of value of **`/xkms:OpaqueClientData`** exceeds 256 bytes
`TrustCenter NotReachable`	Responder of certificate issuer CA not reached - time-out limit reached or other technical reasons
`WrongCertificateFormat`	Certificate defect or wrong coded
`WrongTimeInstant`	Validation time instant not recognisable or in future
`UnknownCA`	Certificate issuer not known
`SignatureKeyTooShort`	Key length of signature certificate is too short

[18] A proposal for a XML "Data Structure for the Security Suitability of Cryptographic Algorithms" was published 2009 as RFC 5698: http://tools.ietf.org/html/rfc5698. Alike TSL's, such information may be exposed in machine-readable format in the future.

ErrorExtension/Reason URI ending	Semantics
`NextResponderInChainNotReached`	While chaining a request, the addressed next responder could not be reached or did not respond in time
`ResponderChainLoop`	While chaining a request, a loop in the responder chain was detected
`Unknown`	Error reason could not be determined
`RevocationStatusNoLongerProvided`	Certificate issuer no longer offers CRL or OCSP services (e.g. because of closing down business)
`CertQualityNotConsistent`	Different quality levels detected in the certificate chain
`NotUnderstood`	A request parameter could not be understood, but processing was (partially) possible. The indicated parameter SHOULD be outlined in the `xkmsEU:Detail` element of this `xkmsEU:ErrorExtension` entry.

Table 9: XKMS Error Extension: Reasons

6 Indices

6.1 Tables

6.2 Figures

6.3 References

[COMMPKI] Common PKI Specifications for interoperable Applications, Version 2.0, 20 January 2009; http://www.common-pki.org/uploads/media/Common-PKI_v2.0.pdf

[ETSI101456] ETSI: Electronic Signatures and Infrastructures (ESI); Policy Requirements for Certification Authorities issuing Qualified Certificates. ETSI TS 101 456 v1.4.1, 2006.

[ETSI102042] ETSI: Electronic Signatures and Infrastructures (ESI); Policy Requirements for Certification Authorities issuing Public Key Certificates. ETSI TS 102 042 v1.2.2, 2005

[ETSI102231] ETSI: Electronic Signatures and Infrastructures (ESI); Provision of harmonised Trust-service status information; ETSI TS 102 231, v3.1.2, 12-2009, http://www.etsi.org/deliver/etsi_ts/102200_102299/102231/03.01.02_60/ts_102231v030102p.pdf

[ISOCC] ISO 3166-1 Country Codes, Version2006, lat update 2009-10-23, http://www.tm-xml-wiki.org/wiki/TM-XML_ISO_3166_Country_Code_XSD

[PEPPOL-D1.1] PEPPOL project: Requirements for Use of Signatures in Public Procurement Processes. PEPPOL Deliverable D1.1, April 2009, http://www.peppol.eu/work_in_progress/wp-1-esignature/results/deliverable-1.1/first-deliverable-of-wp1-has-been-released

[PEPPOL-D1.2] PEPPOL project: Trans-national Verification Solution(s) – Prototype Documentation. PEPPOL Deliverable D1.2, April 2010, http://www.peppol.eu/work_in_progress/wp-1-esignature/results/deliverable-1.2/d1.2-trans-national-verification-solution-s-prototype-documentation

[PEPPOL-1.3.4] PEPPOL project: Architecture and Trust Models. PEPPOL Deliverable D1.3 Part 4, November 2010, http://www.peppol.eu/work_in_progress/wp-1-

esignature/results/deliverable-1.3-demonstrator-and-functional-specifications-for-cross-border-use-of-esignatures-in-public-procurement/d1.3-part-4-architecture-and-trust-models-v1.9.5.pdf/at_download/file

[RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, RFC 2119, Harvard University, March 1997, http://www.ietf.org/rfc/rfc2119.txt

[RFC4122] A Universally Unique Identifier (UUID) URN Namespace, The Internet Engineering Task Force July 2005, http://www.ietf.org/rfc/rfc4122.txt

[XAdES] European Telecommunications Standards Institute. ETSI TS 101 903: XML Advanced Electronic Signatures, V1.3.2 2006-03; http://webapp.etsi.org/action/PU/20060307/ts_101903v010302p.pdf

[XKMS] XML Key Management Specification (XKMS 2.0) Version 2.0, W3C Recommendation, 28 June 2005, http://www.w3.org/TR/2005/REC-xkms2-20050628/

[XKMSBIND] XML Key Management Specification (XKMS 2.0) Bindings Version 2.0, W3C Recommendation, 28 June 2005, http://www.w3.org/TR/2005/REC-xkms2-bindings-20050628/

[XMLDSIG] World Wide Web Consortium. XML-Signature Syntax and Processing (Second Edition), W3C Recommendation, 10 June 2008; http://www.w3.org/TR/xmldsig-core/

[XMLSchema] World Wide Web Consortium. XML Schema, Parts 0, 1, and 2 (Second Edition). W3C Recommendation, 28 October 2004; http://www.w3.org/TR/xmlschema-0/, http://www.w3.org/TR/xmlschema-1/, and http://www.w3.org/TR/xmlschema-2/

[XML 1.0] World Wide Web Consortium. Extensible Markup Language (XML) 1.0 (Fourth Edition), T. Bray, J. Paoli, C. M. Sperberg-McQueen, and E. Maler, Editors. 10 February 1998, revised 16 August 2006; http://www.w3.org/TR/2006/REC-xml-20060816/

[XPATH 1.0] W3C Recommendation, "XML Path Language (XPath) Version 1.0," 16 November 1999; http://www.w3.org/TR/xpath

7 Appendix: Extension Schema

Schema of PEPPOL XKMS Extensions

Note: In the following schema imports point to local copies in the file system, not to the http-resource as usual. This is necessary because the xkms schema held in http://www.w3.org/TR/xkms2/Schemas/xkms.xsd itself points to a local copy xmldsig-core-schema.xsd of http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/xmldsig-core-schema.xsd, which then leads to the effect of different instances of the schema definition for the namespace of http://www.w3.org/2000/09/xmldsig#

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xkms="http://www.w3.org/2002/03/xkms#"
xmlns:tsl="http://uri.etsi.org/02231/v2#"
xmlns:xkmsEU="http://uri.peppol.eu/xkmsExt/v2#"
xmlns:ds="http://www.w3.org/2000/09/xmldsig#" xmlns:isocc="http://www.tm-
xml.org/XMLSchema/common" targetNamespace="http://uri.peppol.eu/xkmsExt/v2#"
elementFormDefault="qualified" attributeFormDefault="unqualified" xml:lang="EN">
   <xs:annotation>
      <xs:documentation xml:lang="en">This schema serves the requirements of EC
PEPPOL Project regarding certificate validation as an extension to XKMS2 XKISS
ValidateResult</xs:documentation>
   </xs:annotation>
   <xs:import namespace="http://www.w3.org/2002/03/xkms#"
schemaLocation="xkms.xsd"/>
   <xs:import namespace="http://www.w3.org/2000/09/xmldsig#"
schemaLocation="xmldsig-core-schema.xsd"/>
   <xs:import namespace="http://uri.etsi.org/02231/v2#"
schemaLocation="ts_102231v030102_xsd.xsd"/>
   <xs:import namespace="http://www.tm-xml.org/XMLSchema/common"
schemaLocation="ISOCountryCodeType-V2006.xsd"/>
   <!-- ValidateResultExtEU -->
   <xs:element name="ValidateResultExtEU" type="xkmsEU:ValidateResultExtEUType"
substitutionGroup="xkms:MessageExtension"/>
   <xs:complexType name="ValidateResultExtEUType">
      <xs:complexContent>
         <xs:extension base="xkms:MessageExtensionAbstractType">
            <xs:sequence>
               <xs:element ref="xkmsEU:eIDQuality" minOccurs="0"/>
               <xs:element ref="tsl:ServiceInformation" minOccurs="0"/>
               <xs:element ref="xkmsEU:ValidationDetails" minOccurs="0"/>
               <xs:element ref="xkmsEU:ErrorExtension" minOccurs="0"
maxOccurs="unbounded"/>
               <xs:element ref="xkmsEU:ResponderDetails" maxOccurs="unbounded"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
   <!-- /ValidateResultExtEU -->
   <!-- ValidationDetails -->
   <xs:element name="ValidationDetails" type="xkmsEU:ValidationDetailsType"/>
   <xs:complexType name="ValidationDetailsType">
      <xs:sequence>
         <xs:element ref="xkmsEU:ValidateScheme"/>
         <xs:element ref="xkmsEU:ValidateModel" minOccurs="0"/>
         <xs:element ref="xkmsEU:ValidationTimeQueried"/>
         <xs:element ref="xkmsEU:ValidationTime"/>
         <xs:element name="CertIssuingCountry" type="isocc:ISOCountryCodeType">
            <xs:annotation>
               <xs:documentation>ISO 3166-1 Country Codes</xs:documentation>
            </xs:annotation>
```

```
            </xs:element>
            <xs:element ref="xkmsEU:OCSPNoCache" minOccurs="0"/>
            <xs:element ref="xkmsEU:CertificateRevocationDetails" minOccurs="0"/>
         </xs:sequence>
      </xs:complexType>
      <!-- /ValidationDetails -->
      <!-- ValidateScheme -->
      <xs:element name="ValidateScheme" type="xkmsEU:ValidateSchemeType"/>
      <xs:simpleType name="ValidateSchemeType">
         <xs:restriction base="xs:anyURI">
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeLOCAL"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeOCSP"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#valSchemeOCSPCommonPKI"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeCRL"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeCRLLDAP"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeLDAP"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valSchemeNONE"/>
         </xs:restriction>
      </xs:simpleType>
      <!-- /ValidateScheme -->
      <!-- ValidateModel -->
      <xs:element name="ValidateModel" type="xkmsEU:ValidateModelType"/>
      <xs:simpleType name="ValidateModelType">
         <xs:restriction base="xs:anyURI">
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valModelPKIX"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#valModelChain"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#valModelEscapeRoute"/>
         </xs:restriction>
      </xs:simpleType>
      <!-- /ValidateModel -->
      <!-- ValidationTime -->
      <xs:element name="ValidationTime" type="xs:dateTime"/>
      <!-- /ValidationTime -->
      <!-- ValidationTimeQueried -->
      <xs:element name="ValidationTimeQueried" type="xs:dateTime"/>
      <!-- /ValidationTimeQueried -->
      <!-- OCSPNoCache -->
      <xs:element name="OCSPNoCache" type="xs:boolean"/>
      <!-- /OCSPNoCache -->
      <!-- CertificateRevocationDetails -->
      <xs:element name="CertificateRevocationDetails">
         <xs:complexType>
            <xs:sequence>
               <xs:element name="RevocationTimeInstant" type="xs:dateTime"/>
               <xs:element ref="xkmsEU:RevocationReason"/>
            </xs:sequence>
         </xs:complexType>
      </xs:element>
      <xs:element name="RevocationReason" type="xkmsEU:RevocationReasonType"/>
      <xs:simpleType name="RevocationReasonType">
         <xs:restriction base="xs:anyURI">
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonUnspecified"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonKeyCompromise"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonCACompromise"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonAffiliationChanged"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonSuperseded"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonCessationOfOperation"/>
```

```
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonCertificateHold"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonRemoveFromCRL"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonPrivilegeWithdrawn"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonAACompromise"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#revocationReasonNone"/>
         </xs:restriction>
      </xs:simpleType>
      <!-- /CertificateRevocationDetails -->
      <!-- CertificateQuality -->
      <xs:element name="CertificateQuality" type="xkmsEU:CertificateQualityType"/>
      <xs:simpleType name="CertificateQualityType">
         <xs:restriction base="xs:anyURI">
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#certqualityUnknown"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#certqualityLow"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#certqualityLCP"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#certqualityNCP"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#certqualityNCPPLUS"/>
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#certqualityQCP"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#certqualityQCPPLUS"/>
         </xs:restriction>
      </xs:simpleType>
      <!-- /CertificateQuality -->
      <!-- CSPAssurance -->
      <xs:element name="CSPAssurance" type="xkmsEU:CSPAssuranceType"/>
      <xs:simpleType name="CSPAssuranceType">
         <xs:restriction base="xs:anyURI">
            <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceNone"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceIndependentDocumentReview"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceInternalComplianceAudit"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceSupervisionWithoutComplianceAudi
t"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceExternalComplianceAudit"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceExternalComplianceAuditCertified
"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceSupervisionWithExternalComplianc
eAudit"/>
            <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#CSPAssuranceAccreditationWithExternalComplia
nceAudit"/>
         </xs:restriction>
      </xs:simpleType>
      <!-- /CSPAssurance -->
      <!-- eIDQuality -->
      <xs:element name="eIDQuality" type="xkmsEU:eIDQualityType"/>
      <xs:complexType name="eIDQualityType">
         <xs:sequence>
            <xs:element ref="xkmsEU:CertificateQuality"/>
            <xs:element ref="xkmsEU:CSPAssurance"/>
         </xs:sequence>
      </xs:complexType>
      <!-- /eIDQuality -->
      <!-- ResponderDetails -->
```

```
   <xs:element name="ResponderDetails" type="xkmsEU:ResponderDetailsType"/>
   <xs:complexType name="ResponderDetailsType">
      <xs:sequence>
         <xs:element ref="tsl:TSPInformation" minOccurs="0"/>
         <xs:element name="ConfigurationVersion" type="xs:string" minOccurs="0"/>
         <xs:element name="OCSPCacheingInterval" type="xs:duration" minOccurs="0"/>
         <xs:element name="TSLIdentifier" type="tsl:NonEmptyURIType"
minOccurs="0"/>
         <xs:element name="PolicyIdentifier" type="xs:anyURI" minOccurs="0"/>
         <xs:element name="AlgPolicyIdentifier" type="tsl:NonEmptyURIType"
minOccurs="0"/>
      </xs:sequence>
      <xs:attribute name="ChainingTo" type="tsl:NonEmptyURIType" use="optional"/>
   </xs:complexType>
   <!-- /ResponderDetails -->
   <!-- ErrorExtension -->
   <xs:element name="ErrorExtension" type="xkmsEU:ErrorExtensionType"/>
   <xs:complexType name="ErrorExtensionType">
      <xs:sequence>
         <xs:element name="Reason" type="xkmsEU:ReasonType"/>
         <xs:element name="Detail" type="xs:string" minOccurs="0"/>
      </xs:sequence>
   </xs:complexType>
   <xs:simpleType name="ReasonType">
      <xs:restriction base="xs:anyURI">
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonOpaqueClientDataTooLong"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonTrustCenterNotReachable"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonWrongCertificateFormat"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonWrongTimeInstant"/>
         <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#reasonUnknownCA"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonSignatureKeyTooShort"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonNextResponderInChainNotReached"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonResponderChainLoop"/>
         <xs:enumeration value="http://uri.peppol.eu/xkmsExt/v2#reasonUnknown"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonNotUnderstood"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonRevocationStatusNoLongerProvided"/>
         <xs:enumeration
value="http://uri.peppol.eu/xkmsExt/v2#reasonCertQualityNotConsistent"/>
      </xs:restriction>
   </xs:simpleType>
   <!-- /ErrorExtension -->
   <!-- ValidateRequestExtEU -->
   <xs:element name="ValidateRequestExtEU" type="xkmsEU:ValidateRequestExtEUType"
substitutionGroup="xkms:MessageExtension"/>
   <xs:complexType name="ValidateRequestExtEUType">
      <xs:complexContent>
         <xs:extension base="xkms:MessageExtensionAbstractType">
            <xs:sequence>
               <xs:element ref="xkmsEU:RequestingNodeChain"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
   <!-- /ValidateRequestExtEU -->
   <!-- RequestingNodeChain -->
   <xs:element name="RequestingNodeChain" type="xkmsEU:RequestingNodeChainType"
substitutionGroup="xkms:MessageExtension"/>
```

```
   <xs:complexType name="RequestingNodeChainType">
      <xs:complexContent>
         <xs:extension base="xkms:MessageExtensionAbstractType">
            <xs:sequence>
               <xs:element name="RequestingNode" type="tsl:NonEmptyURIType"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
         </xs:extension>
      </xs:complexContent>
   </xs:complexType>
   <!-- /RequestingNodeChain -->
   <!-- /XKISS EU Extension  end schema -->
</xs:schema>
```