Peppol Authority

Koordinierungsstelle für IT-Standards

Testing in the Peppol Network: What should be done? What support is available?

Informationen for Service Providers

© 2026 Koordinierungsstelle für IT-Standard
(Coordination Office for IT-Standards)

Version 1.3

History

Version	Date	Autor	Kapitel	Status
1.0	2022/08/12	Philip Helger	-	Release (Translation: Robert Connor)
1.1	2023/01/11	Philip Helger	3.4 (Link XRechung)	Release
1.2	2023/08/17	Philip Helger	5.1.3 (Testbed link)	Release
1.3	2026/07/09	Philip Helger	Links and PASR	Release

Content

1 Introduction 3
2 Testing in general 3
3 Artefacts whose behaviour is tested 4
3.1 Software components to be tested 4
3.2 Data formats to be tested 5
3.3 Other components that should be tested 5
3.4 Supplementary requirements in Germany 6
4 Test methodology 7
4.1 Testing software components 7
4.2 Testing data formats 7
4.2.1 XML based data formats 7
4.2.2 Other data formats 8
4.3 Testing other artefacts 8
4.3.1 Testing X.509 certificates 8
4.4 Test methodology in the Peppol Network 9
5 Test support 10
5.1 Support via OpenPeppol 10
5.1.1 Peppol eDEC specifications 10
5.1.2 Peppol Business Interoperability Specifications (Peppol BIS) 10
5.1.3 Peppol Testbed 10
5.1.4 Peppol Service Desk 11
5.2 Support via the Community 11
6 Appendix A 13
6.1 XML Schema: Sources 13
6.2 Schematron: sources 13

1 Introduction

This document is to assist Peppol service providers in testing solutions and in operational / support activities. It is a supplement to the Peppol test functionalities and documents provided by OpenPeppol.

It is intended for service providers who have a good basic understanding of the subject.

The document refers at individual points to tools that may be suitable for support. No liability is accepted for their availability or correct functioning.

Delimitations:

- Generating or processing 'test results' or 'test reports' is not the focus of this document.
- The document concerns only scenarios from the post-award phase. The pre-award phase is not considered.
- The tests named below refer exclusively to the verification of the correct implementation of the corresponding Peppol specifications, i.e., in particular not to specific software products.

Note: The use of RFC 2119 terminology (SHALL, SHOULD, CAN , etc.) is omitted as this document is not a specification.

2 Testing in general

Within the Peppol network, the main purpose of testing is to confirm that a specification is or has been correctly implemented.

Testing differentiates between the following cases:

- **Good case** — no error (see below) is expected — and
- **Bad case** — at least one error is expected (see below)

Each error message must be distinguished by a weighting:

- **German: Fehler** (English "error" or "fatal error") — unable to continue processing — and
- **German: Warnung** (English "warning") — processing can be continued

When at least one error message with the weighting "error" is shown, a test is regarded as a bad case. If only error messages are shown with a weighting "warning" at most, the test is regarded as a good case.

Delimitation: This document does not cover non-discrete tests, such as the availability of an access point. It only concerns tests that have a clear assertion of the defined assumption at one point in time.

3 Artefacts whose behaviour is tested

Peppol uses the 4-corner model to enable an information exchange for business data, whereby Peppol communication takes place exclusively between nodes C2 (service provider of the sender) and C3 (service provider of the receiver). Specific software components are used as well as specific data formats.

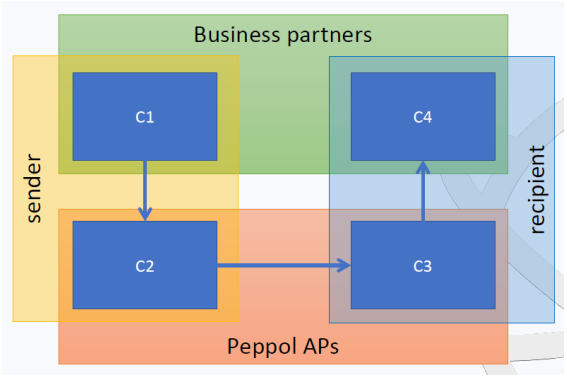


Diagram 1: Peppol 4-corner model

The components to be tested by a Peppol Service Provider depend on the respective Peppol membership, or the services to be offered via the Peppol network: Post-Award and / or Capability Lookup Service (see Service Provider Agreement, Annex 4).

This is referred to in the remainder of the document.

3.1 Software components to be tested

The following Peppol-specific software components must be tested by the service provider:

- Access Point (AP), for the exchange of business documents
  - C2 as sender or sending organisation
  - C3 as receiver or receiving organisation
- Capability Lookup Service or Service Metadata Publisher (SMP)[1] (only for C3): for providing routing information
- SMP Lookup Client (only for C2): to retrieve routing information on an SMP
- Service Metadata Locator (SML). This is a central component that is only used in conjunction with the SMP.
- Peppol Directory (PD) This is a central component that is only used in conjunction with the SMP.

Service providers that exclusively offer a post-award service, test only the Access Point and the SMP Lookup Client.

Service providers that only offer a Capability Lookup Service, test only the SMP, as well as the interfaces to the SML and Peppol Directory.

[1] The Capability Lookup Service is referred to in the following as SMP for short.

3.2 Data formats to be tested

The following Peppol-specific formats are to be tested:

In the context of the AP:

- AS4: for transferring documents between APs
- ebMS: defines the header data of an AS4 message (e.g., as AS4 UserMessage).
- SBDH: the Peppol Envelope Format "Standard Business Document Header" is XML-based. It encompasses the BIS data with routing-specific metadata according to the Peppol Envelope specification[2]
- SOAP: is used in an AS4 message as a container for AS4 user messages and WS-security data.
- WS-Security - used in an AS4 message

In the context of the SMP:

- PD: the Peppol Directory specific data formats, according to the Peppol Directory specification.[3]
- SML-data types: SML-specific data formats, according to the Peppol SML specification.[4]
- SMP-data types: SMP-specific data formats, according to Peppol SMP specification.[5]
- XMLDSig - used for the digital signature of SMP responses

In the context of document exchange:

- BIS (Peppol Business Interoperability Specification): the Peppol-specific standards for data exchange between C2 and C3. They are based exclusively on UBL.
- EN16931: EN16931 specifies the European standard that defines a uniform semantic data model for electronic invoices. It forms the basis of Peppol BISBilling3.0.
- UBL: a collection of XML-based document specifications

Almost all of the formats mentioned are XML-based. The only exception is the AS4 message format when exchanging via HTTPS, which follows the MIME Multipart specification.

Service Providers that only offer a post-award service, test only the formats in the contexts of Access Point and Document Exchange.

Service providers that only offer the Capability Lookup service, test only the formats in the SMP context.

3.3 Other components that should be tested

The following aspects should also be tested:

---

[2] 9.7.2026: https://docs.peppol.eu/edelivery/

[3] 9.7.2026: https://docs.peppol.eu/edelivery/

[4] 9.7.2026: https://docs.peppol.eu/edelivery/

[5] 9.7.2026: https://docs.peppol.eu/edelivery/

- Consistency of Peppol X.509 certificates
    - concerns all service providers
- Consistency of TLS certificates between access points
    - only concerns service providers that (also) provide a post award service
- Compliance with the "Peppol Policy for use of Identifiers".[6]

3.4 Supplementary requirements in Germany

In Germany the XRechnung format may be used for addressing public authorities or for B2B.

- XRechnung[7]: The CIUS defined by the public authorities, available as UBL and CII
- Schema Leitweg-ID (route ID): national identifier developed to address exclusively public entities in Germany.
- CII - Cross Industry Invoice: XML based e-invoice format. The public authorities shall in principle also be able to accept and process XRechnung in CII format.

All the formats mentioned are XML-based.

Service providers that offer a post-award service must test accordingly.

[6] 9.7.2026: https://docs.peppol.eu/edelivery/

[7] https://www.xeinkauf.de/xrechnung

4 Test methodology

Separate test cases shall be defined for each of the software components.

There are a number of basic tools available for testing data formats that can be used in virtually the same way everywhere.

4.1 Testing software components

In practice, the software provided/implemented by service providers is much more extensive than required by the underlying Peppol specifications. The tests listed below are for the sole purpose of determining the correctness of the relevant Peppol specifications.

The list of current Peppol eDelivery specifications can always be found at https://docs.peppol.eu/edelivery/.

4.2 Testing data formats

4.2.1 XML based data formats

There are official XML Schemas (XSDs) for all the formats mentioned, although for "profiles" such as BIS or XRechnung, only the higher-level ("profiled") XSDs should be used.

Only one XML Schema check is performed for each document to be validated, but this usually refers to more than one XML Schema file. Performing XML schema validations is an obligation that service providers must fulfil as part of the Peppol Architectural Framework.

Schematron is an additional tool used for validating an XML document and is based on an XML schema. XSD checks structure and data types, whereas Schematron can be used to check cross-references and other complex relationships within a document. Several Schematron rule sets can also be used for complete validation of a document, which build on each other.

Through the combination of XSD and Schematron a validation pyramid is created (see diagram 2). This shows that a file must be validated from the bottom up. Should an error be found during the XSD validation process, the Schematron rules should no longer be used, due to the high probability of subsequent errors occurring (Schematron errors that occur only due to the XSD error).

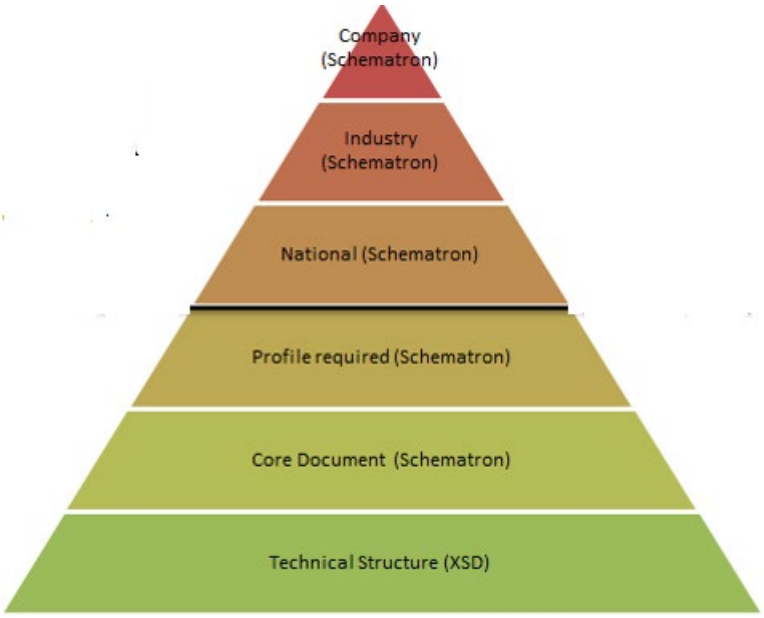


Diagram 2: XML Validation Pyramid

An important aspect of both XSD and Schematron rules is to ensure the correct version of the rules is used. In XSD, these rules are usually very stable, whereas Schematron rules can typically change.

Chapters 6.1 and 6.2 contain the references to / sources of all XML Schema and Schematron rules relevant in the context of Peppol in Germany.

4.2.2 Other data formats

Non-XML-based file formats should be tested in accordance with the relevant format specification.

4.3 Testing other artefacts

4.3.1 Testing X.509 certificates

X.509 certificates in the Peppol Network can be tested as follows:

- Are the data regarding the validity of the X.509 certificate within the certificate consistent? (validity period, issuer, serial number of the issuer)
- Was the certificate revoked? (OCSP query / CRL verification)
- Is the correlation between a Peppol X.509 certificate and a signature generated at a certain point in time from it coherent?
- Is the correlation between a Peppol X.509 certificate and a message encoded with it coherent?

4.4 Test methodology in the Peppol Network

Within the Peppol Network, testing should only be done in the Test Network. This includes the following components:

- The use of Peppol test certificates for the Access Point
- The use of Peppol test certificates for the SMP
- The use of the SMK for the registration of test participants
- The use of the domain `participant.sml.test.tech.peppol.org.`[8] for registering or looking up Peppol Participant IDs

Note: When sending messages via the Peppol test network, not only is the system that is being tested likely to be affected, but also external systems from OpenPeppol or other Peppol service providers. This should be taken into account especially for load tests or tests with large amounts of data.

[8] The domain acc.edelivery.tech.ec.europa.eu. may not be used anymore.

5 Test support

5.1 Support via OpenPeppol

OpenPeppol provides a variety of support services that are continuously expanded and improved in the Peppol network.

5.1.1 Peppol eDEC specifications

All components used in the Peppol network have either been specified or profiled by OpenPeppol. (Profiling indicates placing further boundaries on an existing specification). These specifications contain instructions on how a software component should be implemented. A complete list can be found at https://docs.peppol.eu/edelivery/.

5.1.2 Peppol Business Interoperability Specifications (Peppol BIS)

For the Peppol BIS formats there are Schematron rules that are continuously maintained by OpenPeppol (see chapter 6.2). These represent the reference implementation.

In the case of electronic invoicing (UBL Invoice or CreditNote), they are based on the EN16931 Schematron rules, which are available for download in the appropriate version of OpenPeppol.

Note: Since older rule versions of OpenPeppol are not available for download, local versioning is recommended. A collection of old rule versions is available here for information only and without responsibility: https://github.com/phax/phive-rules/tree/master/phive-rules-peppol/docs[9].

In addition to the Schematron rules, the BIS specifications also contain explanations of the individual XML elements and attributes as well as code lists for values to be used.

5.1.3 Peppol Testbed

The Peppol Testbed is the official onboarding tool of OpenPeppol. This must be used initially by all Peppol service providers with Peppol test certificates. The report generated there serves as proof of compatibility and is necessary in order to issue the productive certificates.

The Testbed can be found here: https://www.testbed.peppol.org/. To use it, the Peppol certificate must be imported into the browser, which is required as a client certificate in the TLS handshake.

[9] The collection is administered by Philip Helger. No liability is accepted for completeness and correctness.

5.1.4 Peppol Service Desk

Service Providers can contact the Peppol Service Desk at all times free of charge. Clarification and compliance questions can be addressed there. Communication language is English: Peppol Service Desk - Jira Service Management (atlassian.net)

5.2 Support via the Community

In the Peppol network there is a multitude of tools, most of which are also open source. The following list is not intended to be exhaustive. No guarantee is given for accessibility and functionality, nor is any review made. Component strings are in alphabetical order.

- Full Access Point solutions:
    - Domibus – Domibus - Digital – (EU-sample-implementation)
    - Holodeck – http://holodeck-b2b.org/download/
    - Oxalis – https://github.com/OxalisCommunity/
    - phoss-ap – https://github.com/phax/phoss-ap
- AS4 solutions:
    - phase4 – https://github.com/phax/phase4
- SMP Server solutions:
    - phoss SMP – https://github.com/phax/phoss-smp/
- Validation solutions for XML-Schema and Schematron:
    - ecosio Validator – https://ecosio.com/en/peppol-and-xml-document-validator/
    - KoSIT Validator – https://github.com/itplr-kosit/validator
        - Validator configuration BIS – https://github.com/itplr-kosit/validator-configuration-bis
        - Validator configuration XRechnung – https://github.com/itplr-kosit/validator-configuration-xrechnung
    - Peppol Practical – https://peppol.helger.com/public/menuitem-validation-ws2
    - phorm – https://github.comphax/phorm
        - phive – https://github.com/phax/phive
        - phive-rules – https://github.com/phax/phive-rules/
- Sub components for solving specific sub problems
    - peppol-commons – https://github.com/phax/peppol-commons
    - ph-cii – https://github.com/phax/ph-cii
    - ph-sbdh – https://github.com/phax/ph-sbdh/
    - ph-schematron – https://github.com/phax/ph-schematron/

- ph-ubl – https://github.com/phax/ph-ubl
- Unimaze Peppol Stylesheets – https://github.com/unimaze/unimaze-peppol-stylesheets
- Visualization of XML invoices: List of open source tools - https://github.com/phax/en16931-visualization/

All tools made available are, generally, subjected to extensive testing before they are published. However, since not all tools are subjected to the same tests, there may be some gaps in the tests.

In some instances, especially with the handling of data formats, due care and attention is not always paid and XML-schema checks are often not implemented or done incorrectly.

6 Appendix A

6.1 XML Schema: Sources

The sources of the XML schemas to be used for each format are as per July 2026:

- BIS: uses UBL 2.1 XSDs
- CII: Version D16B – the XSDs are in https://unece.org/DAM/cefact/xml_schemas/D16B_SCRDM__Subset__CII.zip
- ebMS: https://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/ebms-header-3_0-200704.xsd - important is to take account of the patch from https://issues.oasis-open.org/projects/EBXMLMSG/issues/EBXMLMSG-2?filter=allissues. A patched version of the XSD file can be found at https://github.com/phax/phase4/blob/master/phase4-lib/src/main/resources/external/schemas/ebms-header-3_0-200704.xsd .
  The file https://docs.oasis-open.org/ebxml-bp/2.0.4/OS/signalSchema/ebbp-signals-2.0.4.xsd for the „Non Repudiation“ data structures is also needed.
- EN16931:2017: uses UBL 2.1 XSDs bzw. die CII D16B XSDs
- SBDH: Version 1.3 – die XSDs can be downloaded at https://www.gs1.org/sbdh-xml-schema-files
- PD, SML and SMP data types: can be downloaded at https://docs.peppol.eu/edelivery/
- SOAP Version 1.2: is used in AS4. The XSD is available at https://www.w3.org/2003/05/soap-envelope/
- UBL Version 2.1: XSDs are in https://docs.oasis-open.org/ubl/os-UBL-2.1/UBL-2.1.zip
- Web Service Security, Version 1.1.1: the XSD can be found at https://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd
- XRechnung (based on EN 16931:2017): uses UBL 2.1 XSDs / CII D16B XSDs

6.2 Schematron: sources

The sources of the Schematron rules to be used for each format are as per July 2026:

- BIS
  - Invoice: The Schematron rules for Peppol BIS Billing 3.0 can be downloaded from https://docs.peppol.eu/poacc/billing/3.0/. These rules change at least twice per year – the validity rules are specified at https://peppol.org/documentation/technical-documentation/post-award-documentation/. The Billing 3.0 Schematron rules build on the EN16931 Schematron rules.
  - All areas except invoice: The Schematron rules can be downloaded at https://docs.peppol.eu/poacc/upgrade-3/. These rules are not based on other Schematron rules.
- CII: No Schematron rules defined.
- ebMS: No Schematron rules defined.

- EN16931: The official EN 16931 Schematron rules are updated by DG DIGIT and published at https://github.com/ConnectingEurope/eInvoicing-EN16931/releases . Ensure particular attention must be given to the version, as some rules are incompatible with each other.
- PD: No Schematron rules defined.
- SBDH: No Schematron rules defined.
- SML: No Schematron rules defined.
- SMP: No Schematron rules defined.
- SOAP: No Schematron rules defined.
- UBL: No Schematron rules defined.
- Web Service Security: No Schematron rules defined.
- XRechnung: The official Schematron rules are published, amongst others, at https://github.com/itplr-kosit/xrechnung-schematron/releases, and are also available in other download packages. XRechnung-Schematron rules build on the EN16931 Schematron rules. The Schematron rules must always match the XRechnung version.