

# PEPPOL Access Point

## Hans Berg

PEPPOL infrastructure cloud services, OpenPEPPOL eDelivery community leader

[hans.berg@tickstar.com](mailto:hans.berg@tickstar.com)

## Mike Aksamit

Lead architect

[mikael.aksamit@tickstar.com](mailto:mikael.aksamit@tickstar.com)

Slides available on

[galaxygw.com/news](http://galaxygw.com/news)

[twitter.com/GalaxyGateway](https://twitter.com/GalaxyGateway)

## PEPPOL Technical key characteristics

- Technical architecture
  - Receiver discovery via SML/SMP
  - Security
  - Integrity (sending AP vouches for the identity of the sender)

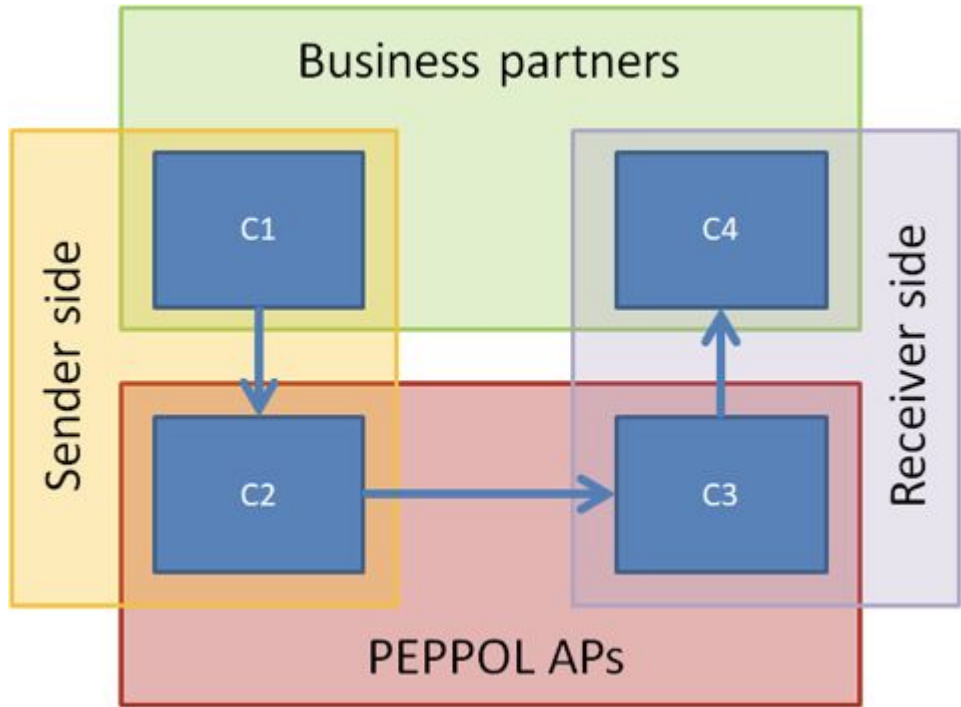
## Business key characteristics

- Anyone can become an access point provider
- No roaming fees allowed
- No roaming agreements between Access Point service providers needed
- A receiving AP (corner 3) is not allowed to reject an incoming transaction
- Sending AP (corner 2) vouches for the identity of the sender (corner 1)

## Technical Infrastructure

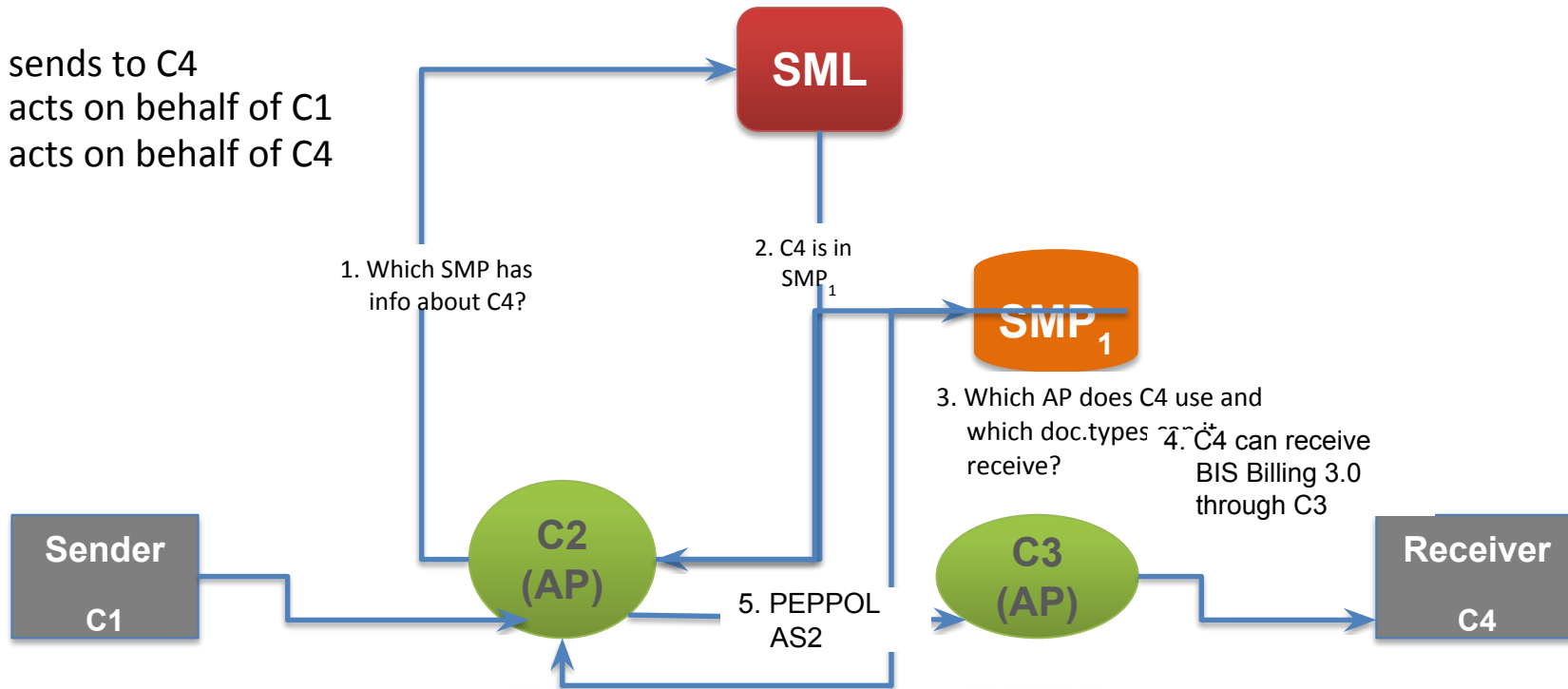
- SML (Service Metadata Locator)
  - A DNS storing information about where SMPs are located and which that contains metadata about a certain receiver. The SML is the only centrally operated component in the PEPPOL transport infrastructure.
  
- SMP (Service Metadata Publisher)
  - Publishes metadata and receiving capability about a receiver and which Access Point it uses to receive a particular document type.
  
- AP (Access Point)
  - Connects with other Access Points to receive and sends transaction with the AS2 Protocol.

# Role of Access Point in 4 corner model



# A PEPPOL Transaction

C1 sends to C4  
C2 acts on behalf of C1  
C3 acts on behalf of C4



# The SML is a DNS



Source: Policy 9 in "PEPPOL Policy for the use of identifiers"  
available on <https://github.com/OpenPEPPOL/documentation/tree/master/TransportInfrastructure>



# SMP Service Metadata



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns3:SignedServiceMetadata xmlns="http://busdox.org/transport/identifiers/1.0/" xmlns:ns2="http://www.w3.org/2005/08/addressing" xmlns:ns3="http://busdox.org/serviceMetadata/publishing/1.0/">
  <ns3:ServiceMetadata>
    <ns3:ServiceInformation>
      <ParticipantIdentifier scheme="iso6523-actorid-upis">0088:50512318800008</ParticipantIdentifier>
      <DocumentIdentifier scheme="busdox-docid-qns">urn:oasis:names:specification:ubl:schema:xsd:Invoice-2::Invoice##urn:www.cenbii.eu:transaction:biitrns010:ver2.0:extended:urn:www.cenbii.eu:profile:bii05:ver2.0</DocumentIdentifier>
    <ns3:ProcessList>
      <ns3:Process>
        <ProcessIdentifier scheme="cenbii-procid-ubl">urn:www.cenbii.eu:profile:bii05:ver2.0</ProcessIdentifier>
        <ns3:ServiceEndpointList>
          <ns3:Endpoint transportProfile="busdox-transport-as2-ver1p0">
            <ns2:EndpointReference>
              <ns2:Address>https://peppol.zzz.com/yyy/adapter/inbound/as2peppol</ns2:Address>
            </ns2:EndpointReference>
            <ns3:RequireBusinessLevelSignature>false</ns3:RequireBusinessLevelSignature>
            <ns3:MinimumAuthenticationLevel>1</ns3:MinimumAuthenticationLevel>
            <ns3:ServiceActivationDate>2017-03-13Z</ns3:ServiceActivationDate>
            <ns3:ServiceExpirationDate>2027-03-13Z</ns3:ServiceExpirationDate>
            <ns3:Certificate>MIIENjCCAx6gAwIBAgIOAovA/eZvvKqJmu+nv11PdDANBgkqhkiG9w0BAQsFADBx
```

- The Participant's identifier (PEPPOL id)
- Supported message type
- Business process
- Transport protocol to use for this message
- Technical endpoint/address to where the message should be sent

# How to become an Access Point provider

- Implement yourself using the specifications
  - <https://peppol.eu/downloads/the-peppol-edelivery-network-specifications/>
  
- Open source implementations
  - <https://peppol.eu/downloads/peppolimplementations/>
  - Oxalis, PH AP
  
- Buy an Access Point as a service
  - Consider white label vs branded

The used protocol during exchange between two Access Points

- AS2 (Current)
  - Secured by sending AP (C2) signing the contents, receiving AP (C3) calculates the signature of received contents and verifies against the signature to assert integrity of the exchanged information. Contents of transaction readable by Access Point.
  - MDN used in synchronous response to ACK/NACK the exchange.
- AS4 (Upcoming)
  - Specification currently under review.
  - Will be mandated not earlier than mid 2019.
- In all situations
  - Sending AP (C2) vouches for sending end user (C1)
  - Sending AP (C2) must ensure sent contents are valid

## SBDH - Standard Business Document Header

- XML
- Header with metadata
- One payload (XML) attached within the SBDH

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<StandardBusinessDocument xmlns="http://www.unece.org/cefact/namespaces/StandardBusinessDocumentHeader">
  <StandardBusinessDocumentHeader>
    <HeaderVersion>1.0</HeaderVersion>
    <Sender>
      <Identifier Authority="iso6523-actorid-upis">0007:SENDERPARTICIPANT</Identifier>
    </Sender>
    <Receiver>
      <Identifier Authority="iso6523-actorid-upis">0007:RECEIVERPARTICIPANT</Identifier>
    </Receiver>
    <DocumentIdentification>
      <Standard>urn:oasis:names:specification:ubl:schema:xsd:Invoice-2</Standard>
      <TypeVersion>2.1</TypeVersion>
      <InstanceIdentifier>f415743d-df26-4dea-a975-51b930c6ceba</InstanceIdentifier>
      <Type>Invoice</Type>
      <CreationDateAndTime>2018-07-03T10:00:10Z</CreationDateAndTime>
    </DocumentIdentification>
    <BusinessScope>
      <Scope>
        <Type>DOCUMENTID</Type>
        <InstanceIdentifier>urn:oasis:names:specification:ubl:schema:xsd:Invoice-2:Invoice#urn:www.cenbii.eu:transaction:biitrs010:ver2.0:extended:urn:www.peppol.eu:bis:peppol4a:ver2.0:2.1</InstanceIdentifier>
      </Scope>
      <Scope>
        <Type>PROCESSID</Type>
        <InstanceIdentifier>urn:www.cenbii.eu:profile:bi04:ver2.0</InstanceIdentifier>
      </Scope>
    </BusinessScope>
  </StandardBusinessDocumentHeader>
  <Invoice xmlns="urn:oasis:names:specification:ubl:schema:xsd:Invoice-2" xmlns:cac="urn:oasis:names:specification:ubl:schema:xsd:CommonAggregateComponents-2" xmlns:cbc="urn:oasis:names:specification:ubl:schema:xsd:CommonBasicComponents-2" xmlns:ccts="urn:un:unece:cefact:documentation:2" xmlns:cdt="urn:oasis:names:specification:ubl:schema:xsd:QualifiedDatatypes-2" xmlns:udt="urn:un:unece:cefact:data:specification:UnqualifiedDatatypesSchemaModule:2">
    ... cut ...
  </Invoice>
</StandardBusinessDocument>
```

## Error Handling

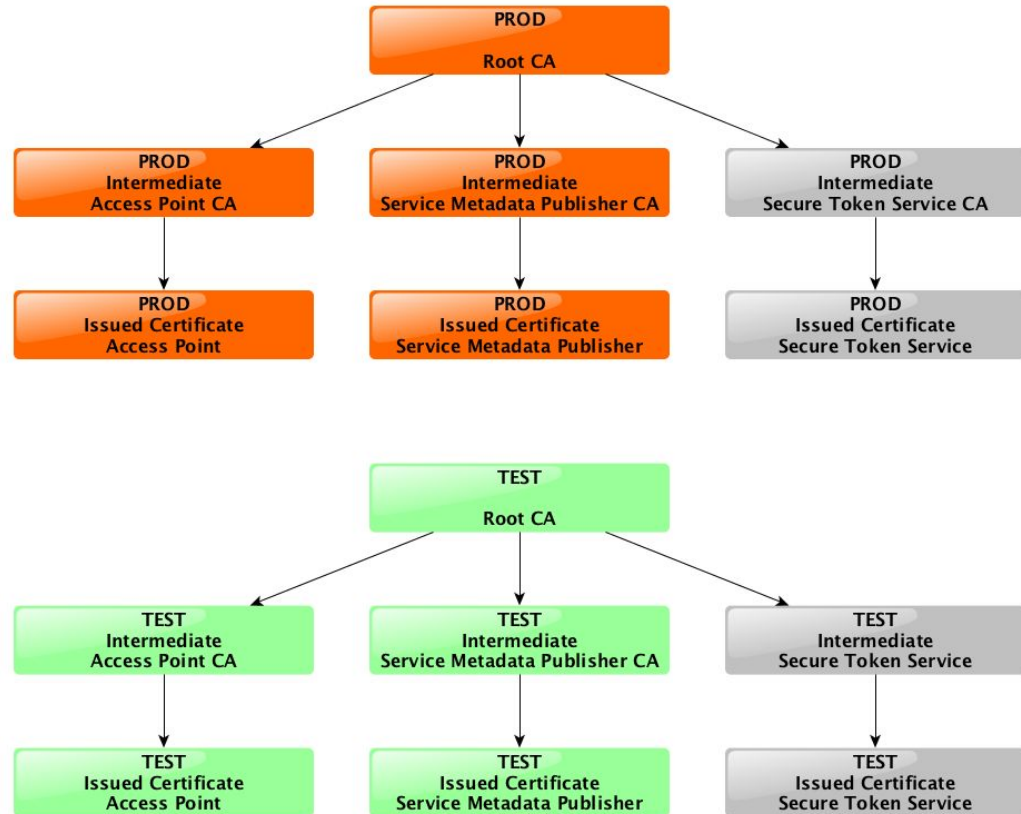
The following scenarios are important to deal with for a sending AP (C2)

- Receiving AP (C3) returns HTTP 404
- Receiving AP (C3) not responding
- SMP Response reports error

Consider a retry mechanism with a backoff policy for the listed scenarios.

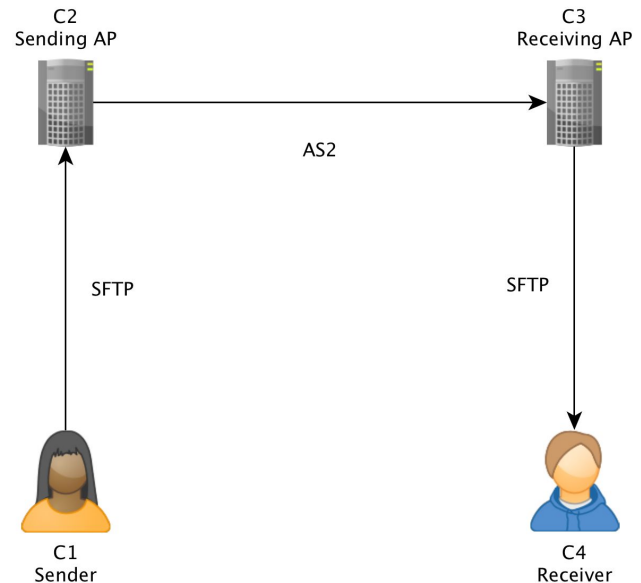
## PKI v.2 → v.3 migrator

- Starts **2018-09-03** (Sending AP can use either v2 or v3, receiving AP MUST accept both)
- Ends **2018-11-30** (All OpenPEPPOL transactions only to use v3)



## Access Point and SMP Lab

1. Create PEPPOL participant in SMP
2. Verify participant metadata with a lookup
3. Prepare a test file with an envelope
4. Upload file to sftp folder
5. Poll for receipt and payload



Source: [www.galaxygw.com/support/integration-test](http://www.galaxygw.com/support/integration-test)

## How to become an Access Point, step by step

1. Select your AP option
  - a. Develop yourself
  - b. Use open source options
  - c. buy as a service
2. [Join OpenPEPPOL](#) and sign the [Transport Infrastructure agreement](#) (TIA) with a [PEPPOL authority](#).
3. Learn about the current versions of the mandatory PEPPOL BIS message type.
4. (Deploy your access point and establish processes for operations, maintenance and SLA compliance)
5. (Perform the test procedure mandated by OpenPEPPOL.)
6. Test sign-off
7. Done! You can now exchange messages with all PEPPOL Access Points.

Source: [www.galaxygw.com/support/faq](http://www.galaxygw.com/support/faq)



## How to become an SMP, step by step

1. Select your SMP option
  - a. Develop yourself
  - b. Use open source options
  - c. buy as a service
2. [Join OpenPEPPOL](#) and sign the [Transport Infrastructure agreement](#) (TIA) with a [PEPPOL authority](#).
3. (Deploy SMP and establish processes for operations, maintenance and SLA compliance)
4. (Perform the test procedure mandated by OpenPEPPOL.)
5. Test sign-off
6. Done! You can now start to publish your PEPPOL receivers with your SMP

Source: [www.galaxygw.com/support/faq](http://www.galaxygw.com/support/faq)