



جهاز الضرائب
TAX AUTHORITY

Oman SMP API Specifications – v1.0 (4-6-2026)

@omantax 

www.taxoman.gov.om 



Table of Contents

1. Oman SMP API Specifications – v1.0 (4-6-2026)	3
1.1 Version Control	3
1.2 SMP Service Description	3
1.3 Table of URLs	3
1.4 For Developers	3
1.5 API Routes	4
1.5.1 API Routes - Table of API Routes	4
1.5.2 API Routes – Source Code	4
2. Schemas	6
2.1 Schemas – Table of Schemas.....	6
2.1.1 Table of Schemas	6
2.2 Schemas – Source Code.....	7
3. Full SMP API Specification Source Code.....	13

1. Oman SMP API Specifications – v1.0 (4-6-2026)

1.1 Version Control

Version	Date	Description	Detailed Changelog
1.0	4-6-2026	First version	-

1.2 SMP Service Description

The Oman SMP REST API provides secure, standards-based access to Service Metadata Publisher (SMP) resources for participant SML, participant service metadata, and participant Peppol Directory publication/amendment/removal.

1.3 Table of URLs

API Category	Security	Path Prefix	Environments	
			T-SMP URL	Production SMP URL
POST/PUT/DELETE APIs	mTLS Secured	/v1/secure/smp/wrapper	test-smp-ext-api.taxoman.gov.om	To be provided upon go live

1.4 For Developers

For an interactable view please utilize the following editor: <https://editor.swagger.io/>, Copy and Paste the full API Specification (see below) on the editor's text box.



1.5 API Routes

1.5.1 API Routes - Table of API Routes

API Routes	Available Methods	Version	Description	Request Security
/{participantId}/services/{documentTypeId}	PUT	v1	Add participant (if not created before) Add service meta data including process list for the documents	Authenticated using certificate
	DELETE	v1	Delete a document type from the participant's list of applicable documents	Authenticated using certificate
/{participantId}	DELETE	v1	Deletes the participant from the SMP, SML & Directory	Authenticated using certificate

1.5.2 API Routes – Source Code

/{participantId}
<pre> /{participantId}: parameters: - name: participantId in: path required: true description: The participant identifier (e.g. iso6523-actorid-upis::0088:123456). schema: type: string example: iso6523-actorid-upis::9915:test-participant description: The identifier must be URL-encoded when used in the path. For example, the '#' character should be encoded as '%23', and ':' should be encoded as '%3A'. delete: tags: - ServiceGroup summary: Delete Service Group description: Removes a participant ID and issues a deletion command to the SML. operationId: deleteServiceGroup security: - mTLS: [] responses: '200': description: Service Group successfully deleted. '204': description: Idempotent delete success (if configured in server properties). '404': description: Service Group does not exist. '500': description: Internal server error. </pre>



{/participantId}/services/{DocumentTypeId}

```

/{participantId}/services/{documentTypeId}:
parameters:
  - name: participantId
    in: path
    required: true
    schema:
      type: string
      example: iso6523-actorid-upis::9915:test-participant
      description:
        The identifier must be URL-encoded when used in the path. For example, the '#' character should be encoded
        as '%23', and ':' should be encoded as '%3A'.
  - name: documentTypeId
    in: path
    required: true
    description: The document type identifier representing the standard and format.
    schema:
      type: string
      example: busdcox-docid-qns::urn:oasis:names:specification:ubl:schema:xsd:Invoice-
      2::Invoice%23%23urn:cen.eu:en16931:2017%23compliant%23urn:fdc:peppol.eu:2017:poacc:billing:3.0::2.1
      description: The identifier must be URL-encoded when used in the path. For example, the '#' character should be encoded
        as '%23', and ':' should be encoded as '%3A'.
put:
  tags:
    - ServiceMetadata
  summary: Create or Update Service Metadata & Optionally the business card
  description: |
    As per wrapper definition, this method does the following:
    - Creates the AP endpoint URLs, transport profiles, and certificates. Overwrites existing.
    - Creates the participant if not already existing.
    - Pushes participant to SML
    - Pushes participant to Peppol Directory
    Note that validations and restrictions can happen on above actions.
  operationId: putServiceMetadataAndBusinessCard
  security:
    - mTLS: []
  requestBody:
    required: true
    content:
      application/xml:
        schema:
          $ref: '#/components/schemas/ParticipantDetailsType'
        example: |
          <ParticipantDetails>
            <smp:ServiceMetadata
xmlns:wsa="http://www.w3.org/2005/08/addressing" xmlns:smp="http://busdcox.org/serviceMetadata/publishing/1.0/" xmlns:id="http://busdcox.org/transport/identifiers/1.0/">
              <smp:ServiceInformation>
                <id:ParticipantIdentifier scheme="iso6523-actorid-upis">9876:participantid</id:ParticipantIdentifier>
                <id:DocumentIdentifier scheme="busdcox-docid-qns">urn:oasis:names:specification:ubl:schema:xsd:Invoice-
                2::Invoice##urn:cen.eu:en16931:2017#compliant#urn:fdc:peppol.eu:2017:poacc:billing:3.0::2.1</id:DocumentIdentifier>
              <smp:ProcessList>
                <smp:Process>
                  <id:ProcessIdentifier scheme="cenbii-procid-ubl">urn:fdc:peppol.eu:2017:poacc:billing:01:1.0</id:ProcessIdentifier>
                  <smp:ServiceEndpointList>
                    <smp:Endpoint transportProfile="peppol-transport-as4-v2_0">
                      <wsa:EndpointReference>
                        <wsa:Address>http://test.smpserver/as4</wsa:Address>
                      </wsa:EndpointReference>
                      <smp:RequireBusinessLevelSignature>>false</smp:RequireBusinessLevelSignature>
                      <smp:Certificate>blacert</smp:Certificate>
                      <smp:ServiceDescription>Unit test service</smp:ServiceDescription>
                      <smp:TechnicalContactUrl>https://tms.taxoman.gov.om/portal/</smp:TechnicalContactUrl>
                    </smp:Endpoint>
                  </smp:ServiceEndpointList>
                </smp:Process>
              </smp:ProcessList>
            </smp:ServiceInformation>
          </smp:ServiceMetadata>
          <BusinessCard xmlns="http://www.peppol.eu/schema/pd/businesscard/20180621/">
            <BusinessEntity registrationDate="2026-01-01">
              <Name language="en">Test entity Oman</Name>
              <Name language="ar">عمان</Name>
              <CountryCode>OM</CountryCode>
            </BusinessEntity>
          </BusinessCard>
        </ParticipantDetails>
  responses:
    '200':
      description: Successful update.
    '400':
      description: Bad request.
    '401':
      description: Unauthorized.
    '500':
      description: Internal server error.
delete:
  tags:
    - ServiceMetadata
  summary: Delete Service Metadata
  description: |
    As per wrapper definition, this method does the following:
    - Removes specific routing configurations for a document type.
  operationId: deleteServiceMetadata
  security:
    - mTLS: []
  
```

```
responses:
  '200':
    description: Successfully deleted.
  '404':
    description: Metadata not found.
  '500':
    description: Internal server error.
```

2. Schemas

2.1 Schemas – Table of Schemas

Schemas are defined, referenceable objects with fields that can have rules such as "maxLength". They are reusable and can reference/nest each-other.

2.1.1 Table of Schemas

Schema Name	Description
ParticipantDetailsType	Used for when the service provider does an SMP /services request for a participant
ServiceMetadataType	Contains either detailed service information (ServiceInformation) or a redirect to another SMP (Redirect).
ServiceInformationType	Contains the full details of a service, including participant and document identifiers, and a list of processes.
ProcessType	Defines a specific business process, identified by a ProcessIdentifier, and lists the service endpoints for that process.
EndpointType	Defines the technical details of a transport endpoint, including its address, transport profile, certificate, and other metadata.
RedirectType	Defines a redirect to another SMP for a specific document type, including the target SMP's URL and certificate UID.
ParticipantIdentifierType	A standardized identifier for a participant in the network, consisting of a scheme and a value.
DocumentIdentifierType	A standardized identifier for a document type, consisting of a scheme and a value.
ProcessIdentifierType	A standardized identifier for a business process, consisting of a scheme and a value.
ExtensionType	A generic object used for future extensions to the SMP specification.
BusinessCardType	Root element for a single Peppol Directory business card.
BusinessEntityType	Represents a single business entity as contained in a business card.
ContactType	Resembles the structure of a single business contact.
MultilingualNameType	Represents a name that can be provided in multiple languages. In XML, 'language' is an attribute.

2.2 Schemas – Source Code

ParticipantDetailsType

```
ParticipantDetailsType:
  type: object
  required:
    - ServiceMetadata
  xml:
    name: ParticipantDetails
  properties:
    ServiceMetadata:
      $ref: '#/components/schemas/ServiceMetadataType'
    BusinessCard:
      $ref: '#/components/schemas/BusinessCardType'
```

ServiceMetadataType

```
ServiceMetadataType:
  type: object
  xml:
    name: ServiceMetadata
  properties:
    ServiceInformation:
      $ref: '#/components/schemas/ServiceInformationType'
    Redirect:
      $ref: '#/components/schemas/RedirectType'
```

ServiceInformationType

```
ServiceInformationType:
  type: object
  xml:
    name: ServiceInformation
  required:
    - ParticipantIdentifier
    - DocumentIdentifier
    - ProcessList
  properties:
    ParticipantIdentifier:
      $ref: '#/components/schemas/ParticipantIdentifierType'
    DocumentIdentifier:
      $ref: '#/components/schemas/DocumentIdentifierType'
    ProcessList:
      type: object
      xml:
        name: ProcessList
      properties:
        Process:
          type: array
          minItems: 1
          xml:
            wrapped: false
            name: Process
          items:
            $ref: '#/components/schemas/ProcessType'
    Extension:
      $ref: '#/components/schemas/ExtensionType'
```

ProcessType

```
ProcessType:
  type: object
  xml:
    name: Process
  required:
    - ProcessIdentifier
    - ServiceEndpointList
```



```

properties:
  ProcessIdentifier:
    $ref: '#/components/schemas/ProcessIdentifierType'
  ServiceEndpointList:
    type: object
    xml:
      name: ServiceEndpointList
    properties:
      Endpoint:
        type: array
        minItems: 1
        xml:
          wrapped: false
          name: Endpoint
        items:
          $ref: '#/components/schemas/EndpointType'
  Extension:
    $ref: '#/components/schemas/ExtensionType'

```

EndpointType

```

EndpointType:
  type: object
  xml:
    name: Endpoint
  required:
    - transportProfile
    - EndpointReference
    - RequireBusinessLevelSignature
    - Certificate
    - ServiceDescription
    - TechnicalContactUrl
  properties:
    transportProfile:
      type: string
      xml:
        attribute: true
    EndpointReference:
      type: object
      xml:
        name: EndpointReference
      required:
        - Address
      properties:
        Address:
          type: string
          format: uri
    RequireBusinessLevelSignature:
      type: boolean
    MinimumAuthenticationLevel:
      type: string
    ServiceActivationDate:
      type: string
      format: date-time
    ServiceExpirationDate:
      type: string
      format: date-time
    Certificate:
      type: string
    ServiceDescription:
      type: string
    TechnicalContactUrl:
      type: string
      format: uri
    TechnicalInformationUrl:
      type: string
      format: uri
  Extension:
    $ref: '#/components/schemas/ExtensionType'

```

RedirectType



```
RedirectType:
  type: object
  required:
    - CertificateUID
    - href
  properties:
    CertificateUID:
      type: string
    href:
      type: string
      format: uri
```

ParticipantIdentifierType

```
ParticipantIdentifierType:
  type: object
  xml:
    name: ParticipantIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
      xml:
        attribute: true
    value:
      type: string
```

DocumentIdentifierType

```
DocumentIdentifierType:
  type: object
  xml:
    name: DocumentIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
      xml:
        attribute: true
    value:
      type: string
```

ProcessIdentifierType

```
ProcessIdentifierType:
  type: object
  xml:
    name: ProcessIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
      xml:
        attribute: true
    value:
      type: string
```



ExtensionType

ExtensionType:
type: object

BusinessCardType

BusinessCardType:
type: object
description: Root element for a single Peppol Directory business card.
xml:
name: BusinessCard
properties:
ParticipantIdentifier: # Must not be included in case of using the /{pid}/services/{docId} route
\$ref: '#/components/schemas/ParticipantIdentifierType'
BusinessEntity:
type: array
description: "A list of business entities (companies, public bodies) reachable via the ServiceGroup. In XML, this will be represented by one or more <BusinessEntity> elements."
items:
\$ref: '#/components/schemas/BusinessEntityType'
xml:
wrapped: false

BusinessEntityType

BusinessEntityType:
type: object
description: Represents a single business entity as contained in a business card.
required:
- Name
- CountryCode
xml:
name: BusinessEntity
properties:
Name:
type: array
description: "The multilingual name of the business entity. At least one name must be provided."
minItems: 1
items:
\$ref: '#/components/schemas/MultilingualNameType'
xml:
wrapped: false
CountryCode:
type: string
description: "A 2-letter ISO 3166-1 alpha-2 country code"
minLength: 2
maxLength: 2
xml:
name: CountryCode
registrationDate:
type: string
format: date
description: "The date when the participant was registered for Peppol."
xml:
attribute: true
GeographicalInformation:
type: string
description: "Specific geographic information, such as an address or a region."
xml:
name: GeographicalInformation
Identifier:
type: array
description: "Additional identifiers for this entity, like VAT number, national organisation number, etc."
items:
properties:



```

value:
  type: string
  description: "The value of the identifier (the element's text content)."
```

```

scheme:
  type: string
  description: "The scheme of the identifier. For Peppol, this is typically 'iso6523-actorid-upis'."
```

```

xml:
  attribute: true
```

```

xml:
  wrapped: false
```

```

WebsiteURI:
  type: array
  description: "URIs linking to the business entity's websites."
  items:
    type: string
    format: uri
    xml:
      name: WebsiteURI
```

```

xml:
  wrapped: false
```

```

Contact:
  type: array
  description: "Business contact points to get in touch with the business entity."
  items:
    $ref: '#/components/schemas/ContactType'
```

```

xml:
  wrapped: false
```

```

AdditionalInformation:
  type: string
  description: "Any other relevant information that does not fit into a structured field."
  xml:
    name: AdditionalInformation
```

ContactType

```

ContactType:
  type: object
  description: Resembles the structure of a single business contact.
  xml:
    name: Contact
```

```

properties:
  Type:
    type: string
    description: "The type of the contact (e.g., sales, support)."
```

```

xml:
  name: Type
```

```

Name:
  type: string
  description: "The name of the contact (individual or organisational unit)."
```

```

xml:
  name: Name
```

```

PhoneNumber:
  type: string
  description: "The public telephone number of the contact."
```

```

xml:
  name: PhoneNumber
```

```

Email:
  type: string
  description: "The public email address of the contact."
```

```

xml:
  name: Email
```



MultilingualNameType

```
MultilingualNameType:
  type: object
  description: "Represents a name that can be provided in multiple languages."
  required:
    - value
  xml:
    name: Name
  properties:
    value:
      type: string
      description: A string that must not be empty.
      minLength: 1
    language:
      type: string
      description: "A 2-letter ISO 639-1 language code (e.g., 'en' for English)."      minLength: 2
      maxLength: 2
    xml:
      attribute: true
```

3. Full SMP API Specification Source Code

Full SMP API Specification Source Code

```

openapi: 3.1.0

info:
  title: Oman Tax Authority SMP REST API
  description: |
    The Oman Tax Authority SMP REST API provides secure, standards-based
    access to Service Metadata Publisher (SMP) resources for participant
    service group management, service metadata publication, and
    peppol directory integration.

    mTLS Authentication is required for write operations.
  version: 1.0.0

servers:
  - url: https://{host}/v1/secure/smp/wrapper
    description: Base URL for the Wrapper SMP API. Replace {host} with the actual server address.
    variables:
      host:
        default: test-smp-ext-api.taxoman.gov.om
        description: The hostname of the SMP server.

tags:
  - name: ServiceGroup
    description: Operations related to managing Participant Identifiers and SML synchronization.
  - name: ServiceMetadata
    description: Operations defining document types, business processes, and AS4 transport
    endpoints.

paths:
  /{participantId}:
    parameters:
      - name: participantId
        in: path
        required: true
        description: The participant identifier (e.g. iso6523-actorid-upis::0088:123456).
        schema:
          type: string
          example: iso6523-actorid-upis::9915:test-participant
        description:
          The identifier must be URL-encoded when used in the path. For example, the '#' character
          should be encoded as '%23', and ':' should be encoded as '%3A'.
    delete:
      tags:
        - ServiceGroup
      summary: Delete Service Group
      description: Removes a participant ID and issues a deletion command to the SML.
      operationId: deleteServiceGroup
      security:
        - mTLS: []
      responses:
        '200':
          description: Service Group successfully deleted.
        '204':
          description: Idempotent delete success (if configured in server properties).
        '404':
          description: Service Group does not exist.
        '500':
          description: Internal server error.
  /{participantId}/services/{documentTypeId}:
    parameters:
      - name: participantId
        in: path
        required: true
        schema:
          type: string
          example: iso6523-actorid-upis::9915:test-participant
        description:

```



The identifier must be URL-encoded when used in the path. For example, the '#' character should be encoded as '%23', and ':' should be encoded as '%3A'.

```

- name: documentTypeId
  in: path
  required: true
  description: The document type identifier representing the standard and format.
  schema:
    type: string
    example: busdox-docid-qns::urn:oasis:names:specification:ubl:schema:xsd:Invoice-
2::Invoice%23%23urn:cen.eu:en16931:2017%23compliant%23urn:fdc:peppol.eu:2017:poacc:billing:3.0::2.1
  description: The identifier must be URL-encoded when used in the path. For
example, the '#' character should be encoded as '%23', and ':' should be encoded as '%3A'.
  put:
    tags:
      - ServiceMetadata
    summary: Create or Update Service Metadata & Optionaly the business card
    description: |
      As per wrapper defitintion, this method does the following:
      - Creates the AP endpoint URLs, transport profiles, and certificates. Overwrites existing.
      - Creates the participant if not already existing.
      - Pushes participant to SML
      - Pushes participant to Peppol Directory
      Note that validations and restrictions can happen on above actions.
    operationId: putServiceMetadataAndBusinessCard
    security:
      - mTLS: []
    requestBody:
      required: true
      content:
        application/xml:
          schema:
            $ref: '#/components/schemas/ParticipantDetailsType'
          example: |
            <ParticipantDetails>
            <smp:ServiceMetadata
xmlns:wsa="http://www.w3.org/2005/08/addressing" xmlns:smp="http://busdox.org/serviceMetadata/publis
hing/1.0/" xmlns:id="http://busdox.org/transport/identifiers/1.0/">
            <smp:ServiceInformation>
            <id:ParticipantIdentifier scheme="iso6523-actorid-
upis">9876:participantid</id:ParticipantIdentifier>
            <id:DocumentIdentifier scheme="busdox-docid-
qns">urn:oasis:names:specification:ubl:schema:xsd:Invoice-
2::Invoice##urn:cen.eu:en16931:2017#compliant#urn:fdc:peppol.eu:2017:poacc:billing:3.0::2.1</id:Docu
mentIdentifier>
            <smp:ProcessList>
            <smp:Process>
            <id:ProcessIdentifier scheme="cenbii-procid-
ubl">urn:fdc:peppol.eu:2017:poacc:billing:01:1.0</id:ProcessIdentifier>
            <smp:ServiceEndpointList>
            <smp:Endpoint transportProfile="peppol-transport-as4-v2_0">
            <wsa:EndpointReference>
            <wsa:Address>http://test.smpserver/as4</wsa:Address>
            </wsa:EndpointReference>
            <smp:RequireBusinessLevelSignature>>false</smp:RequireBusinessLevelSignat
ure>
            <smp:Certificate>blacert</smp:Certificate>
            <smp:ServiceDescription>Unit test service</smp:ServiceDescription>
            <smp:TechnicalContactUrl>https://tms.taxoman.gov.om/portal/</smp:Technic
alContactUrl>
            </smp:Endpoint>
            </smp:ServiceEndpointList>
            </smp:Process>
            </smp:ProcessList>
            </smp:ServiceInformation>
            </smp:ServiceMetadata>
            <BusinessCard xmlns="http://www.peppol.eu/schema/pd/businesscard/20180621/">
            <BusinessEntity registrationDate="2026-01-01">
            <Name language="en">Test entity Oman</Name>
            <Name language="ar">عمان</Name>
            <CountryCode>OM</CountryCode>
            </BusinessEntity>
            </BusinessCard>
            </ParticipantDetails>

```



```

responses:
  '200':
    description: Successful update.
  '400':
    description: Bad request.
  '401':
    description: Unauthorized.
  '500':
    description: Internal server error.
delete:
  tags:
    - ServiceMetadata
  summary: Delete Service Metadata
  description: |
    As per wrapper definition, this method does the following:
    - Removes specific routing configurations for a document type.
  operationId: deleteServiceMetadata
  security:
    - mTLS: []
  responses:
    '200':
      description: Successfully deleted.
    '404':
      description: Metadata not found.
    '500':
      description: Internal server error.
components:
  securitySchemes:
    mTLS:
      type: mutualTLS
      description: |
        Mutual TLS (mTLS) authentication.
        The client must present a valid X.509 certificate during the TLS handshake.
        The certificate will be validated by the server.
  schemas:
    ParticipantDetailsType:
      type: object
      required:
        - ServiceMetadata
      xml:
        name: ParticipantDetails
      properties:
        ServiceMetadata:
          $ref: '#/components/schemas/ServiceMetadataType'
        BusinessCard:
          $ref: '#/components/schemas/BusinessCardType'
    ServiceMetadataType:
      type: object
      xml:
        name: ServiceMetadata
      properties:
        ServiceInformation:
          $ref: '#/components/schemas/ServiceInformationType'
        Redirect:
          $ref: '#/components/schemas/RedirectType'
    ServiceInformationType:
      type: object
      xml:
        name: ServiceInformation
      required:
        - ParticipantIdentifier
        - DocumentIdentifier
        - ProcessList
      properties:
        ParticipantIdentifier:
          $ref: '#/components/schemas/ParticipantIdentifierType'
        DocumentIdentifier:
          $ref: '#/components/schemas/DocumentIdentifierType'
        ProcessList:
          type: object

```



```

xml:
  name: ProcessList
  properties:
    Process:
      type: array
      minItems: 1
      xml:
        wrapped: false
        name: Process
      items:
        $ref: '#/components/schemas/ProcessType'
  Extension:
    $ref: '#/components/schemas/ExtensionType'

ProcessType:
  type: object
  xml:
    name: Process
  required:
    - ProcessIdentifier
    - ServiceEndpointList
  properties:
    ProcessIdentifier:
      $ref: '#/components/schemas/ProcessIdentifierType'
    ServiceEndpointList:
      type: object
      xml:
        name: ServiceEndpointList
      properties:
        Endpoint:
          type: array
          minItems: 1
          xml:
            wrapped: false
            name: Endpoint
          items:
            $ref: '#/components/schemas/EndpointType'
  Extension:
    $ref: '#/components/schemas/ExtensionType'

EndpointType:
  type: object
  xml:
    name: Endpoint
  required:
    - transportProfile
    - EndpointReference
    - RequireBusinessLevelSignature
    - Certificate
    - ServiceDescription
    - TechnicalContactUrl
  properties:
    transportProfile:
      type: string
      xml:
        attribute: true
    EndpointReference:
      type: object
      xml:
        name: EndpointReference
      required:
        - Address
      properties:
        Address:
          type: string
          format: uri
    RequireBusinessLevelSignature:
      type: boolean
    MinimumAuthenticationLevel:
      type: string
    ServiceActivationDate:
      type: string
      format: date-time

```



```

ServiceExpirationDate:
  type: string
  format: date-time
Certificate:
  type: string
ServiceDescription:
  type: string
TechnicalContactUrl:
  type: string
  format: uri
TechnicalInformationUrl:
  type: string
  format: uri
Extension:
  $ref: '#/components/schemas/ExtensionType'

RedirectType:
  type: object
  required:
    - CertificateUID
    - href
  properties:
    CertificateUID:
      type: string
    href:
      type: string
      format: uri

ParticipantIdentifierType:
  type: object
  xml:
    name: ParticipantIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
    xml:
      attribute: true
    value:
      type: string

DocumentIdentifierType:
  type: object
  xml:
    name: DocumentIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
    xml:
      attribute: true
    value:
      type: string

ProcessIdentifierType:
  type: object
  xml:
    name: ProcessIdentifier
  required:
    - scheme
    - value
  properties:
    scheme:
      type: string
    xml:
      attribute: true
    value:
      type: string

```



```

ExtensionType:
  type: object

BusinessCardType:
  type: object
  description: Root element for a single Peppol Directory business card.
  xml:
    name: BusinessCard
  properties:
    ParticipantIdentifier:
      $ref: '#/components/schemas/ParticipantIdentifierType'
    BusinessEntity:
      type: array
      description: "A list of business entities (companies, public bodies) reachable via the
ServiceGroup. In XML, this will be represented by one or more <BusinessEntity> elements."
      items:
        $ref: '#/components/schemas/BusinessEntityType'
      xml:
        wrapped: false

BusinessEntityType:
  type: object
  description: Represents a single business entity as contained in a business card.
  required:
    - Name
    - CountryCode
  xml:
    name: BusinessEntity
  properties:
    Name:
      type: array
      description: "The multilingual name of the business entity. At least one name must be
provided."
      minItems: 1
      items:
        $ref: '#/components/schemas/MultilingualNameType'
      xml:
        wrapped: false
    CountryCode:
      type: string
      description: "A 2-letter ISO 3166-1 alpha-2 country code"
      minLength: 2
      maxLength: 2
      xml:
        name: CountryCode
    registrationDate:
      type: string
      format: date
      description: "The date when the participant was registered for Peppol."
      xml:
        attribute: true
    GeographicalInformation:
      type: string
      description: "Specific geographic information, such as an address or a region."
      xml:
        name: GeographicalInformation
    Identifier:
      type: array
      description: "Additional identifiers for this entity, like VAT number, national
organisation number, etc."
      items:
        properties:
          value:
            type: string
            description: "The value of the identifier (the element's text content)."
          scheme:
            type: string
            description: "The scheme of the identifier. For Peppol, this is typically 'iso6523-
actorid-upis'."
        xml:
          attribute: true
      xml:
        wrapped: false
  
```



```

WebsiteURI:
  type: array
  description: "URIs linking to the business entity's websites."
  items:
    type: string
    format: uri
    xml:
      name: WebsiteURI
  xml:
    wrapped: false
Contact:
  type: array
  description: "Business contact points to get in touch with the business entity."
  items:
    $ref: '#/components/schemas/ContactType'
  xml:
    wrapped: false
AdditionalInformation:
  type: string
  description: "Any other relevant information that does not fit into a structured field."
  xml:
    name: AdditionalInformation

ContactType:
  type: object
  description: Resembles the structure of a single business contact.
  xml:
    name: Contact
  properties:
    Type:
      type: string
      description: "The type of the contact (e.g., sales, support)."
```