

RaspConfiguration update for version 2.1.0

The purpose of the RASP version 2.1.0 is to allow PEPPOL BIS documents to be exchanged via the NemHandel infrastructure + remove trivial KeyType mappings.

DocumentType identification

Prior, a DocumentType was identified inside the `<DocumentTypeConfig>` using:

```
<RootNamespace>urn:oasis:names:specification:ubl:schema:xsd:Invoice-2</RootNamespace>
<RootName>Invoice</RootName>
<IdentifierDiscriminators>
  <XPathDiscriminatorConfig>
    <XPathExpression>/root:Invoice/cbc:CustomizationID</XPathExpression>
    <XPathExpectedResult>OIOUBL-2.0(1|2)</XPathExpectedResult>
  </XPathDiscriminatorConfig>
</IdentifierDiscriminators>
```

By utilizing the existing `<XPathDiscriminatorConfig>`, it is possible to discriminate the OIOUBL Invoice from the PEPPOL BIS Invoice.

Schematron validation

Prior, RASP only supported one Schematron/XSLT file per DocumentType using:

```
<SchematronValidationConfig>
  <SchematronDocumentPath>path/to/file/Schematron.xsl</SchematronDocumentPath>
  <ErrorXPath>/Schematron/Error</ErrorXPath>
  <ErrorMessageXPath>/Schematron/Error/Description</ErrorMessageXPath>
</SchematronValidationConfig>
```

In PEPPOL documents, multiple schematron files are provided per DocumentType.

This section must therefore now be wrapped in a container class called `<SchematronValidationConfigs>`:

```
<SchematronValidationConfigs>
  <SchematronValidationConfig>
    <SchematronDocumentPath>path/to/file/Schematron.xsl</SchematronDocumentPath>
    <ErrorXPath>/Schematron/Error</ErrorXPath>
    <ErrorMessageXPath>/Schematron/Error/Description</ErrorMessageXPath>
  </SchematronValidationConfig>
</SchematronValidationConfigs>
```

This allows for multiple `<SchematronValidationConfig>` elements. These elements are validated in the listed order.

KeyTypeMappingExpressions

The Identifier KeyType inside each DocumentTypeConfig is now directly supported in the NHR UDDI Inquiry interface.

Therefore many of the old mappings inside //MappingExpressions/KeyTypeMappingExpressions/Mappings can be removed. Examples of standard mappings that should be removed now:

```
<Mapping>
  <Value>ean</Value>
  <MapsTo>ean</MapsTo>
</Mapping>
<Mapping>
  <Value>cvr</Value>
  <MapsTo>cvr</MapsTo>
</Mapping>
<Mapping>
  <Value>IBAN</Value>
  <MapsTo>iban</MapsTo>
</Mapping>
```

For backward compatibility when sending to RASP Java 2.0.x endpoints, 6 old mappings are essential to keep in relation to OIOUBL:

```
<Mapping>
  <Value>GLN</Value>
  <MapsTo>ean</MapsTo>
</Mapping>
<Mapping>
  <Value>DK:CVR</Value>
  <MapsTo>cvr</MapsTo>
</Mapping>
<Mapping>
  <Value>DK:CPR</Value>
  <MapsTo>cpr</MapsTo>
</Mapping>
<Mapping>
  <Value>DK:P</Value>
  <MapsTo>p</MapsTo>
</Mapping>
<Mapping>
  <Value>DK:SE</Value>
  <MapsTo>se</MapsTo>
</Mapping>
<Mapping>
  <Value>DK:VANS</Value>
  <MapsTo>vans</MapsTo>
</Mapping>
```

In general, the exchanged KeyType value must be one of the following: EAN/GLN, DK:CVR, DK:SE, DK:VANS, DK:P, IBAN, DUNS and DK:CPR. Depending on exchanged document types, other mappings might be needed in customer specific setups.

Only required in C#- cache settings

In Schematron Store Cache configuration inside <ConfigurationSection xsi:type="CacheConfig">, the <ImplementationNamespaceClass> element must be changed to:

```
<ImplementationNamespaceClass>dk.gov.oiosi.common.cache.QuantityCache`2[[System.  
String, mscorlib],[dk.gov.oiosi.xml.schematron.CompiledXslt,  
dk.gov.oiosi.library]]</ImplementationNamespaceClass>
```