

All Fields marked with * are mandatory.

Change Request #:	2
Assigned OGC Document #:	09-069
Name:	*Gerhard Sommer
Organization:	*City of Vienna
Email:	*gerhard.sommer@wien.gv.at
Document Name/Version:	*Web Feature Service (WFS) Implementation Specification / 1.1
OGC Project Document:	*04-094

If this is a revision of a previous submission and you have a Change Request Number, then check here:

Enter the CR number here:

Enter the Revision Number that you are revising here:

Title:	*Correct the base type and facet pattern for TypeNameListType in the WFS 1.1.0 Schema
Source:	*City of Vienna
Work item code:	
Category:	* C (Functional modification of feature)

Reason for change:	* GetFeature requests, which use typeNames in the Query element, that are correct according to the specification (i.e. containing underscores or making use of aliases), currently do not validate, if the implementation (i.e. GeoServer in strict cite-conformance mode) uses schema validation to verify the request.
---------------------------	---

Summary of change:	* Change the base type of TypeNameListType to xsd:string and the facet pattern to <code>([\i-[:]][\c-[:]]*)?[\i-[:]][\c-[:]]*(=\w+)?(,\s*([\i-[:]][\c-[:]]*)?[\i-[:]][\c-[:]]*(=\w+)?)*</code>
---------------------------	--

Consequences if not approved:	Schema validation incorrectly produces an error for certain values of the typeName attribute of the Query element. I.e. it is not possible to use a valid QName as typeName. No aliases or comma-separated lists of typeNames, as described in the annotation of TypeNameListType, can be used.
--------------------------------------	---

Clauses affected:	* 9.2 (page 34 in PDF document)
--------------------------	------------------------------------

Additional Documents affected:

WFS 1.1.0 schema file wfs.xsd

Supporting Documentation:

```
<xsd:simpleType name="Base_TypeNameListType">
  <xsd:list itemType="xsd:QName"/>
</xsd:simpleType>
<xsd:simpleType name="TypeNameListType">
  <xsd:restriction base="wfs:Base_TypeNameListType">
    <xsd:pattern value="((\w:)?\w(=\w?)){1,}">
      <xsd:annotation>
        <xsd:documentation>
          Example typeName attribute value might be:

          typeName="ns1:Inwatera_1m=A, ns2:CoastL_1M=B"

          In this example, A is an alias for ns1:Inwatera_1m
          and B is an alias for ns2:CoastL_1M.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:pattern>
  </xsd:restriction>
</xsd:simpleType>
```

should be

```
<xsd:simpleType name="TypeNameListType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="([\i-[:]][\c-[:]]*:?)[\i-[:]][\c-[:]]*(=\w+)?(\s*([\i-[:]][\c-[:]]*:?)[\i-[:]][\c-[:]]*(=\w+)?)*">
      <xsd:annotation>
        <xsd:documentation>
          Example typeName attribute value might be:

          typeName="ns1:Inwatera_1m=A, ns2:CoastL_1M=B"

          In this example, A is an alias for ns1:Inwatera_1m
          and B is an alias for ns2:CoastL_1M.
        </xsd:documentation>
      </xsd:annotation>
    </xsd:pattern>
  </xsd:restriction>
</xsd:simpleType>
```

Comments:

Explanation:

1) xsd:list would define a list of Qnames separated by whitespace, not ", ". Each part would have to be a QName and therefore "=" or "," would not be allowed. Base_TypeNameListType is not used anywhere else in the schema anyway.

2) `[\i-[:]][\c-[:]]*` is the facet of NCName in the XML Schema standard document.

3) QName is defined in the XML Schema standard document as "NCName" or "NCName:NCName". Therefore the pattern `([\i-[:]][\c-[:]]*:?)[\i-[:]][\c-[:]]*` defines a valid QName.

I have checked this with Altova XMLSpy. I can not check it with GeoServer (using "=" or ","), as it needs a QName as base type (the java code tries to cast the string to QName internally and produces an error if xsd:string is the base type).

There has already been a change request by Clemens Portele, but not specifying the correct pattern (08-042_WFS_1.1_Corrigendum_1.doc).

Status: ⓘ

New

Disposition: ⓘ

Accepted