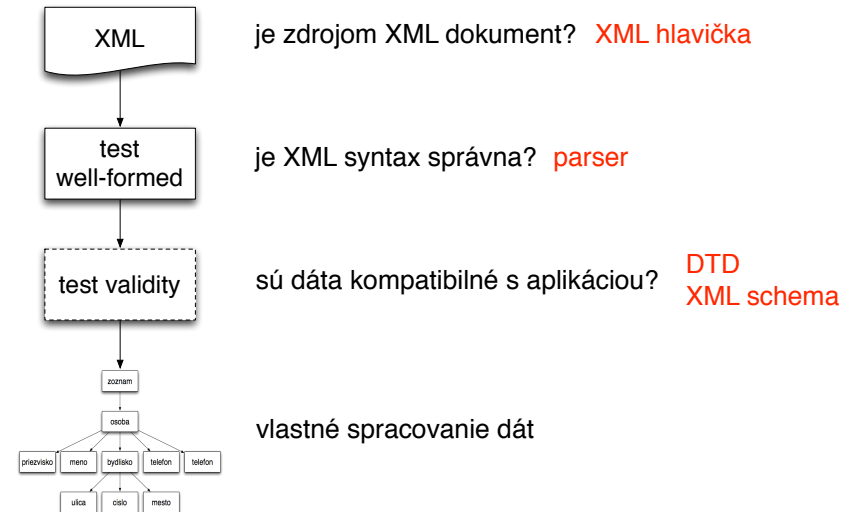


XML Schema

Michal Kvasnica

Spracovanie XML súborov



DTD: sumarizácia

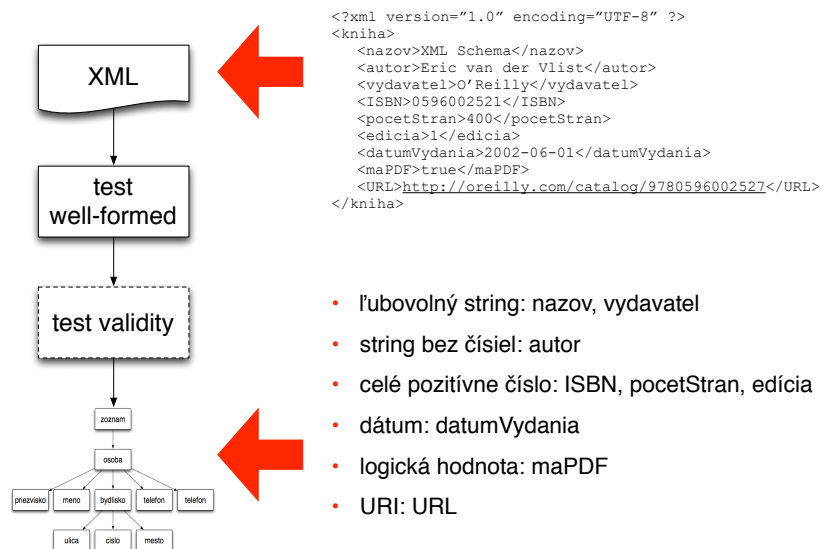
- DTD = spôsob vynútenia si istej štruktúry XML dokumentu
- DTD validuje správnosť daného XML (zmluva)
 - elementy
 - obsah elementov (#PCDATA, iné elementy)
 - atribúty elementov (CDATA, výber z možností, ID, IDREF)
- Nevýhody DTD:
 - #PCDATA nerozlišuje medzi reťazcom, číslom, dátumom, ...
 - iba 4 úrovne opakovania vnorených elementov:
 - práve raz
 - 0 alebo 1 raz
 - jedenkrát alebo viac
 - 0, 1 alebo viackrát

Príklad

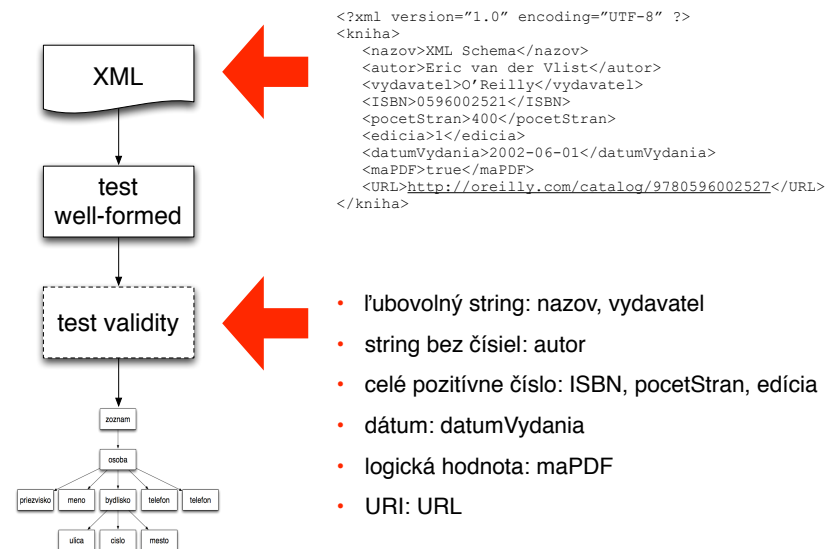
```
<?xml version="1.0" encoding="UTF-8" ?>
<kniha>
  <nazov>XML Schema</nazov>
  <autor>Eric van der Vlist</autor>
  <vydavatel>O'Reilly</vydavatel>
  <ISBN>0596002521</ISBN>
  <pocetStran>400</pocetStran>
  <edicia>1</edicia>
  <datumVydania>2002-06-01</datumVydania>
  <maPDF>>true</maPDF>
  <URL>http://oreilly.com/catalog/9780596002527</URL>
</kniha>
```

- ľubovlný string: nazov, vydavateľ
- string bez čísiel: autor
- celé pozitívne číslo: ISBN, pocetStran, edícia
- dátum: datumVydania
- logická hodnota: maPDF
- URI: URL

DTD



XML Schema

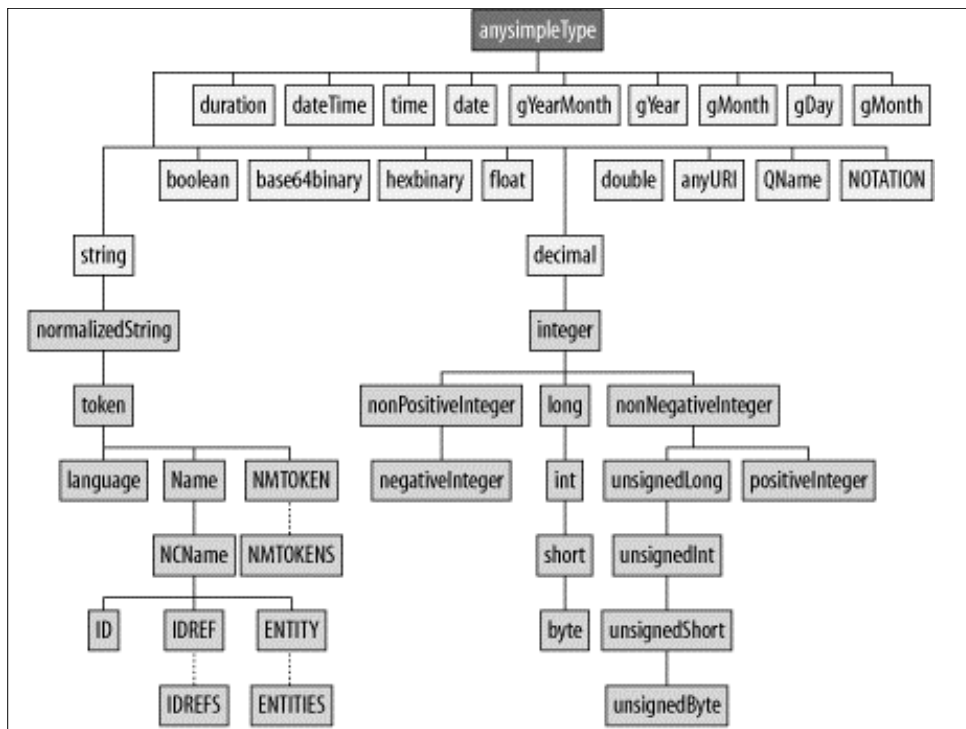


Čo je "XML Schema"

- XML Schema = DTD tak, ako má byť
- Spôsob vynútenia si istej štruktúry XML dokumentu
- Validuje správnosť daného XML (zmluva)
 - elementy
 - obsah elementov
 - atribúty elementov
- Hlavné výhody oproti DTD:
 - kontrola formátu obsahu (reťazec, číslo, boolean, dátum, čas, ...)
 - kontrola vlastností (min/max dĺžka, kladné/záporné hodnoty, ...)
 - flexibilnejšia definícia opakovaní (min/max počet opakovaní, ...)
- Hlavná nevýhoda oproti DTD:
 - komplikovanejšia syntax zápisu

Povolený obsah elementu v DTD

#PCDATA



DTD vs XML Schema

```
<?xml version="1.0" encoding="UTF-8" ?>
<!ELEMENT nazov (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
<!ELEMENT vydavatel (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT pocetStran (#PCDATA)>
<!ELEMENT edicia (#PCDATA)>
<!ELEMENT datumVydania (#PCDATA)>
<!ELEMENT maPDF (#PCDATA)>
<!ELEMENT URL (#PCDATA)>
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:string" />
  <xs:element name="autor" type="xs:token" />
  <xs:element name="vydavatel" type="xs:token" />
  <xs:element name="ISBN" type="xs:positiveInteger" />
  <xs:element name="pocetStran" type="xs:nonNegativeInteger"/>
  <xs:element name="edicia" type="xs:positiveInteger" />
  <xs:element name="datumVydania" type="xs:date" />
  <xs:element name="maPDF" type="xs:boolean" />
  <xs:element name="URL" type="xs:anyURI" />
</xs:schema>
```

Na ďalších prednáškach...

DTD:

```
<!ELEMENT kniha (nazov,autor,ISBN,...)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  ???
</xs:schema>
```

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:string" />
</xs:schema>
```

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

namespace

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

hlavný element
XML schémy

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

definícia nového
elementu

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

atribút "name" definuje
názov elementu

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

atribút "type" definuje
obsah elementu

Definícia elementu

DTD:

```
<!ELEMENT nazov (#PCDATA)>
```

XML Schema:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
</xs:schema>
```

prázdny element
<xs:element>

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string"></xs:element>  
</xs:schema>
```

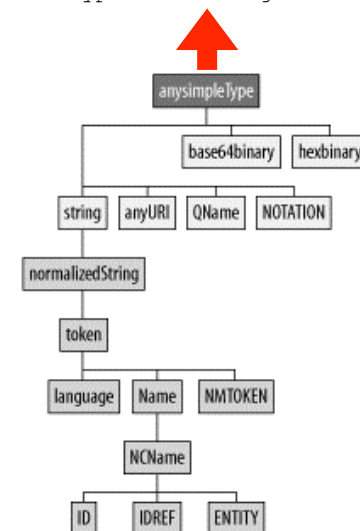
Príklad

```
<?xml version="1.0" encoding="UTF-8" ?>  
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string" />  
  <xs:element name="autor" type="xs:token" />  
  <xs:element name="vydavatel" type="xs:token" />  
  <xs:element name="ISBN" type="xs:positiveInteger" />  
  <xs:element name="pocetStran" type="xs:nonNegativeInteger"/>  
  <xs:element name="edicia" type="xs:positiveInteger" />  
  <xs:element name="datumVydania" type="xs:date" />  
  <xs:element name="maPDF" type="xs:boolean" />  
  <xs:element name="URL" type="xs:anyURI" />  
</xs:schema>
```

```
<nazov>XML Schema</nazov>  
<autor>Eric van der Vlist</autor>  
<vydavatel>O' Reilly</vydavatel>  
<ISBN>0596002521</ISBN>  
<pocetStran>400</pocetStran>  
<edicia>1</edicia>  
<datumVydania>2002-06-01</datumVydania>  
<maPDF>true</maPDF>  
<URL>http://oreilly.com/catalog/9780596002527</URL>
```

Stringový datatype v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:element name="nazov" type="xs:string"/>  
</xs:schema>
```



string

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:string"/>
</xs:schema>
```

XML	Výsledok
<nazov>XML Schema</nazov>	XML Schema
<nazov>XML Schema 1.0</nazov>	XML Schema 1.0
<nazov> XML Schema 1.0 </nazov>	XML Schema 1.0

výstup si uchováva všetky "biele" znaky

normalizedString

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:normalizedString"/>
</xs:schema>
```

XML	Výsledok
<nazov>XML Schema</nazov>	XML Schema
<nazov>XML Schema 1.0</nazov>	XML Schema 1.0
<nazov> XML Schema 1.0 </nazov>	XML Schema 1.0

TAB, CR, LF = príslušný počet medzier

token

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:token"/>
</xs:schema>
```

XML	Výsledok
<nazov>XML Schema</nazov>	XML Schema
<nazov>XML Schema 1.0</nazov>	XML Schema 1.0
<nazov> XML Schema 1.0 </nazov>	XML Schema 1.0

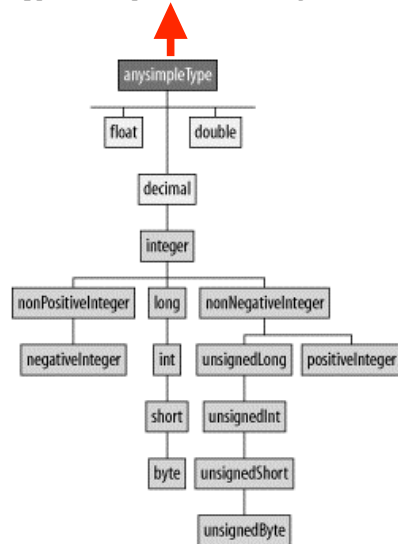
TAB, CR, LF = 1 medzera

Príklad

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:string" />
  <xs:element name="autor" type="xs:token" />
  <xs:element name="vydavatel" type="xs:token" />
  <xs:element name="ISBN" type="xs:positiveInteger" />
  <xs:element name="pocetStran" type="xs:nonNegativeInteger"/>
  <xs:element name="edicia" type="xs:positiveInteger" />
  <xs:element name="datumVydania" type="xs:date" />
  <xs:element name="maPDF" type="xs:boolean" />
  <xs:element name="URL" type="xs:anyURI" />
</xs:schema>
```

Číselný datatype v XML Schema

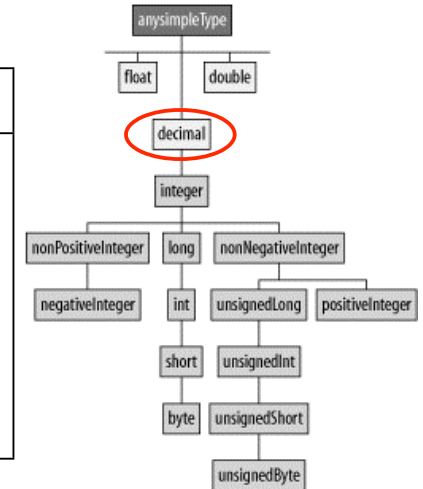
```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="ISBN" type="xs:positiveInteger"/>
</xs:schema>
```



"decimal": čísla s desatinnou čiarokou

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:decimal"/>
</xs:schema>
```

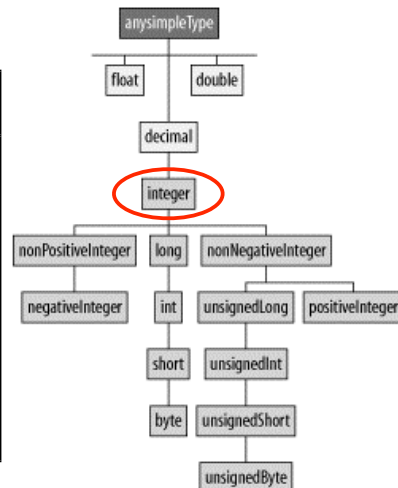
Valid	Non-valid
123	1 234
+123	1 234.56
-123	1,234.56
+000000012	1E2
123456789	- 123
123.	+ 1234.56
+123.456	
-.456	



"integer": celé čísla

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:integer"/>
</xs:schema>
```

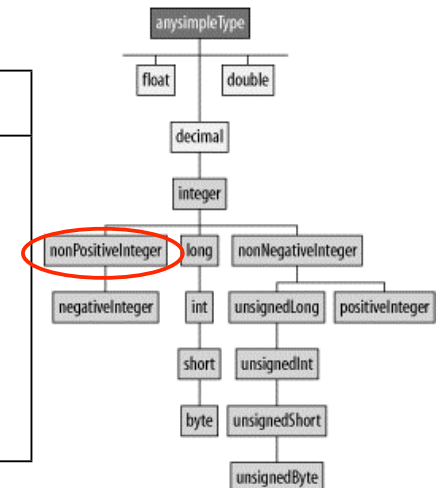
Valid	Non-valid
123	1 234
+123	1E2
-123	- 123
+000000012	1,234
123456789	123.
	123.456



"nonPositiveInteger" (<= 0)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:nonPositiveInteger"/>
</xs:schema>
```

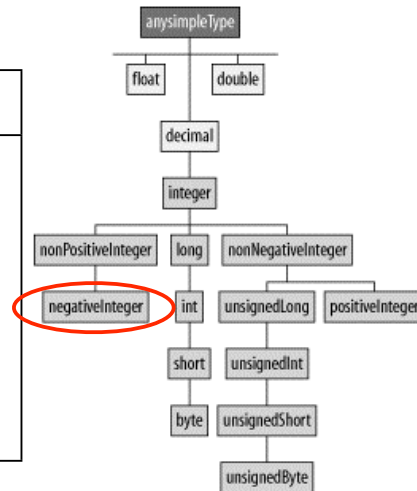
Valid	Non-valid
0	0.
-1	0.1
-123	-1.234
-0001	-1 000
	- 1
	1
	2
	3



“negativeInteger” (< 0)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:negativeInteger"/>
</xs:schema>
```

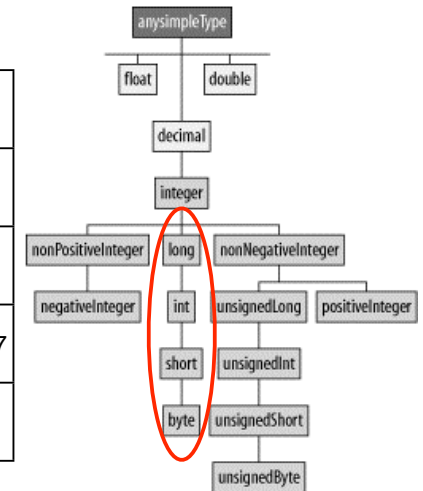
Valid	Non-valid
-1 -123 -0001	0 1 -1. -1.234 -1 000 - 1



“long”, “int”, “short”, “byte”: celé čísla

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:long"/>
</xs:schema>
```

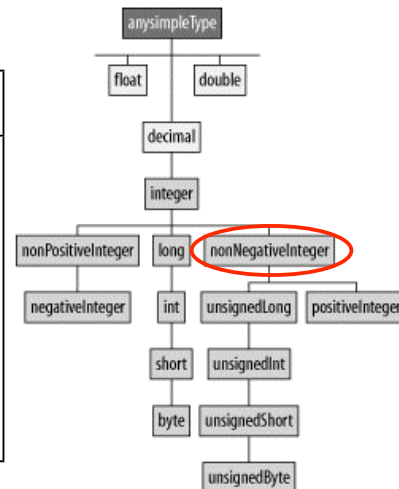
typ	bitov	rozsah
byte	8	-128 až 127
short	16	-32768 až 32767
int	32	-2147483648 až 2147483647
long	64	-9223372036854775808 až 9223372036854775807



“nonNegativeInteger” (>= 0)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:nonNegativeInteger"/>
</xs:schema>
```

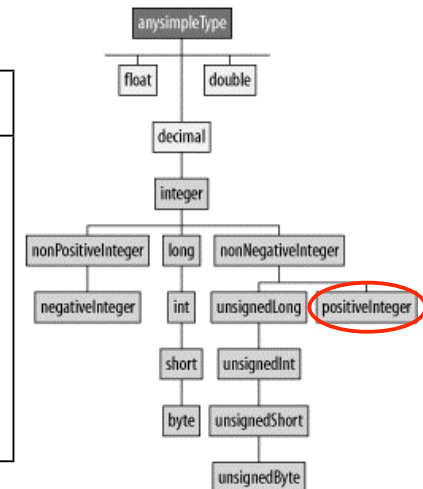
Valid	Non-valid
0 1 +123 000123 123456789	1.1 -123 +1 234 1,000 1E3



“positiveInteger” (> 0)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:positiveInteger"/>
</xs:schema>
```

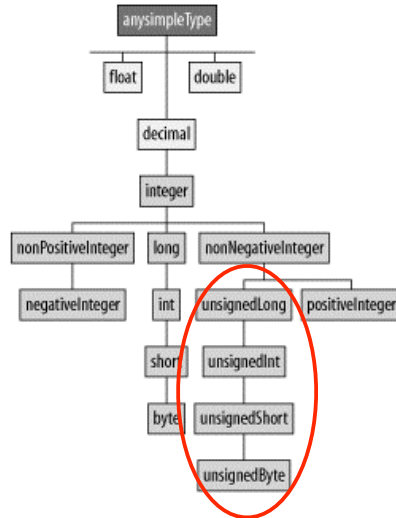
Valid	Non-valid
1 +123 000123 123456789	0 1.1 -123 +1 234 1,000 1E3



unsigned "long", "int", "short", "byte": celé čísla

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:unsignedLong"/>
</xs:schema>
```

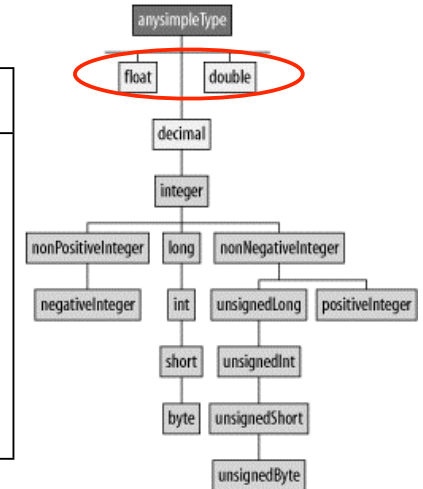
typ	bitov	rozsah
byte	8	0 až 255
short	16	0 až 65535
int	32	0 až 4294967295
long	64	0 až 18446744073709551615



"float" a "double": čísla s desatinnou číarkou

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="pocetStran" type="xs:float"/>
</xs:schema>
```

Valid	Non-valid
123.456	123.4E 56
+123.456	1E2.5
-123.456	+INF
1E2	NAN
1.23E+02	+NaN
-.45E-6	-NaN
INF	
-INF	
NaN	

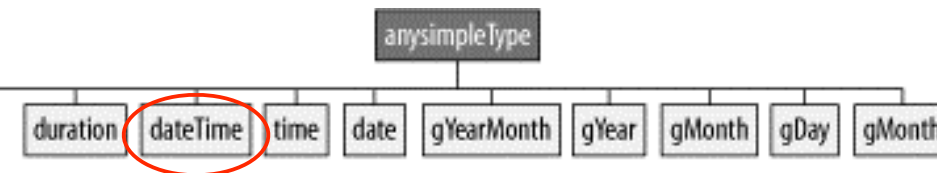


"boolean"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="verejny" type="xs:boolean"/>
</xs:schema>
```

Valid	Non-valid
<verejny>true</verejny>	<verejny></verejny>
<verejny>>false</verejny>	<verejny>True</verejny>
<verejny>1</verejny>	<verejny>FALSE</verejny>
<verejny>0</verejny>	<verejny>2</verejny>
	<verejny>-1</verejny>
	<verejny>1.0</verejny>

Dátumové a časové datatypy



```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="casIncidentu" type="xs:dateTime"/>
</xs:schema>
```

Valid	Non-valid
2001-10-26T21:32:52	2001-10-26
2001-10-26T21:32:52+02:00	2001-10-26T21:32
2001-10-26T19:32:52Z	2001-10-26T25:32:52+02:00
2001-10-26T19:32:52+00:00	01-10-26T21:32
-2001-10-26T21:32:52	26-10-2001T21:32:52
2001-10-26T21:32:52.12679	

Dátumové a časové datatypy

anySimpleType

duration dateTime **time** date gYearMonth gYear gMonth gDay gMonth

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="casIncidentu" type="xs:time"/>
</xs:schema>
```

Valid	Non-valid
21:32:52 21:32:52+02:00 19:32:52Z 19:32:52+00:00 21:32:52.12679	21:32 25:25:10 -10:00:00 1:20:10

Dátumové a časové datatypy

anySimpleType

duration dateTime time **date** gYearMonth gYear gMonth gDay gMonth

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="casIncidentu" type="xs:date"/>
</xs:schema>
```

Valid	Non-valid
2001-10-26 2001-10-26+02:00 2001-10-26Z 2001-10-26+00:00 -2001-10-26 -20000-04-01	26-10-2001 2001/10/26 2001/Oct/26 2001-10 2001-10-32 2001-13-1 01-10-26

Dátumové a časové datatypy

anySimpleType

duration dateTime time date **gYearMonth** **gYear** gMonth gDay gMonth

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="casIncidentu" type="xs:date"/>
</xs:schema>
```

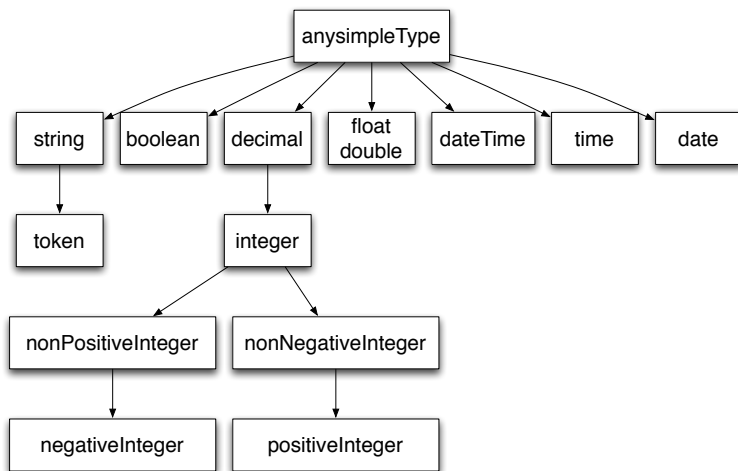
Typ	Valid	Non-valid
gYearMonth	2001-10, -2056-12, -20000-04	2001, 2001-13, 01-10, 10-2001, 2001/10, Oct/2001
gYear	2001, -2056, -20000, 32500	1

Príklad

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="nazov" type="xs:string" />
  <xs:element name="autor" type="xs:token" />
  <xs:element name="vydavatel" type="xs:token" />
  <xs:element name="ISBN" type="xs:positiveInteger" />
  <xs:element name="pocetStran" type="xs:nonNegativeInteger"/>
  <xs:element name="edicia" type="xs:positiveInteger" />
  <xs:element name="datumVydania" type="xs:date" />
  <xs:element name="maPDF" type="xs:boolean" />
  <xs:element name="URL" type="xs:anyURI" />
</xs:schema>
```

```
<nazov>XML Schema</nazov>
<autor>Eric van der Vlist</autor>
<vydavatel>O'Reilly</vydavatel>
<ISBN>-0596002521.23</ISBN>
<pocetStran>400</pocetStran>
<edicia>1.0</edicia>
<datumVydania>2002-06-01T20:00:00</datumVydania>
<maPDF>TRUE</maPDF>
<URL>http://oreilly.com/catalog/9780596002527</URL>
```

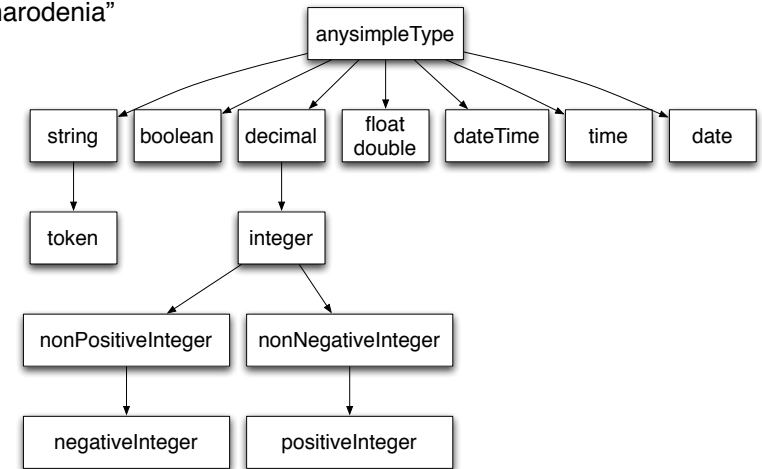
XML Schema datatype cheat sheet



Príklad: deti v pôrodnici

- “meno” a “priezvisko”
- “dátum narodenia”
- “miesto narodenia”
- “Žije”
- “váha”
- “výška”

aké datatypy použiť?



Príklad: deti v pôrodnici

- “meno” a “priezvisko”: tokeny
- “dátum narodenia”: dátum
- “miesto narodenia”: token
- “Žije”: boolean
- “váha”: kladný integer
- “výška”: kladný integer

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="meno" type="xs:token"/>
  <xs:element name="priezvisko" type="xs:token"/>
  <xs:element name="datumNarodenia" type="xs:date"/>
  <xs:element name="zije" type="xs:boolean"/>
  <xs:element name="vaha" type="xs:positiveInteger"/>
  <xs:element name="vyska" type="xs:positiveInteger"/>
</xs:schema>
```

Reštrikcie

- “meno” a “priezvisko”: tokeny (min. 3 a max. 20 znakov)
- “dátum narodenia”: dátum (> 1.1.2009)
- “miesto narodenia”: token (max. 40 znakov)
- “Žije”: boolean
- “váha”: kladný decimal (v rozmedzí od 1.0 do 7.0 kg)
- “výška”: kladný integer (v rozmedzí od 20 do 80 cm)

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"
- ktorý je reštrikciou štandardného datatypu "token"

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"
- ktorý je reštrikciou štandardného datatypu "token"
- 1. reštrikcia: minimálna dĺžka reťazca

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"
- ktorý je reštrikciou štandardného datatypu "token"
- 1. reštrikcia: minimálna dĺžka reťazca 3 znaky

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"
- ktorý je reštrikciou štandardného datatypu "token"
- 1. reštrikcia: minimálna dĺžka reťazca 3 znaky
- 2. reštrikcia: maximálna dĺžka reťazca 5 znakov

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

- definícia nového datatypu, ktorý sa bude volať "string_od_3_do_5"
- ktorý je reštrikciou štandardného datatypu "token"
- 1. reštrikcia: minimálna dĺžka reťazca 3 znaky
- 2. reštrikcia: maximálna dĺžka reťazca 5 znakov
- nový datatyp je globálny pre celú schému a môžeme ho teda použiť ako "typ" elementu "meno"

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

```
<meno>Jan</meno>
```

```
$ xmlLint --noout --schema stringy.xsd zoznam.xml
```

```
zoznam.xml validates
```

Reštrikcie v XML Schema

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
</xs:schema>
```

```
<meno>Michal</meno>
```

```
$ xmlLint --noout --schema stringy.xsd zoznam.xml
```

```
zoznam.xml:2: element meno: Schemas validity error : Element
'meno' [ST 'string_od_3_do_5', facet 'maxLength']: The value
has a length of '6'; this exceeds the allowed maximum length
of '5'.
```

```
zoznam.xml fails to validate
```

Reštrikcie pri báze "string" a "token"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="string_od_3_do_5">
    <xs:restriction base="xs:token">
      <xs:minLength value="3"/>
      <xs:maxLength value="5"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="string2">
    <xs:restriction base="xs:string">
      <xs:length value="2"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="meno" type="string_od_3_do_5"/>
  <xs:element name="okres" type="string2"/>
</xs:schema>
```

- xs:length
- xs:minLength
- xs:maxLength

Enumeratívna reštrikcia

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="okres">
    <xs:restriction base="xs:string">
      <xs:enumeration value="BA"/>
      <xs:enumeration value="PU"/>
      <xs:enumeration value="PB"/>
      <xs:enumeration value="KE"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="okres" type="okres"/>
</xs:schema>
```

```
$ xmlLint --noout --schema stringy.xsd zoznam.xml
```

```
zoznam.xml:2: element okres: Schemas validity error :
Element 'okres' [ST 'okres', facet 'enumeration']: The
value 'NZ' is not an element of the set {'PU', 'BA',
'PB', 'KE'}.
```

```
zoznam.xml fails to validate
```

Enumeratívna reštrikcia je pre všetky datatypy

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="prestupneRoky">
    <xs:restriction base="xs:gYear">
      <xs:enumeration value="2000"/>
      <xs:enumeration value="2004"/>
      <xs:enumeration value="2008"/>
      <xs:enumeration value="2012"/>
      <xs:enumeration value="2016"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="rokNarodenia" type="prestupneRoky"/>
</xs:schema>
```

Enumeratívna reštrikcia je pre všetky datatypy

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="mojeCisla">
    <xs:restriction base="xs:float">
      <xs:enumeration value="3"/>
      <xs:enumeration value="1.26"/>
      <xs:enumeration value="INF"/>
      <xs:enumeration value="-2.1E2"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="stavUctu" type="mojeCisla"/>
</xs:schema>
```

```
<stavUctu>      <stavUctu>      <stavUctu>      <stavUctu>
  3              3.0              -210             12.6E-1
</stavUctu>    </stavUctu>    </stavUctu>    </stavUctu>
```

Reštrikcie

- "meno" a "priezvisko": tokeny (min. 3 a max. 20 znakov) ✓
- "dátum narodenia": dátum (> 1.1.2009)
- "miesto narodenia": token (max. 40 znakov) ✓
- "Žije": boolean
- "váha": kladný decimal (v rozmedzí od 1.0 do 7.0 kg)
- "výška": kladný integer (v rozmedzí od 20 do 80 cm)

Reštrikcia typu "minimálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="novyDatum">
    <xs:restriction base="xs:date">
      <xs:minInclusive value="2009-01-01"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="narodeny" type="novyDatum"/>
</xs:schema>
```

```
<narodeny>2009-02-28</narodeny>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml validates
```

Reštrikcia typu "minimálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="novyDatum">
    <xs:restriction base="xs:date">
      <xs:minInclusive value="2009-01-01"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="narodeny" type="novyDatum"/>
</xs:schema>
```

```
<narodeny>2012-02-29</narodeny>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml validates
```

Reštrikcia typu "minimálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="novyDatum">
    <xs:restriction base="xs:date">
      <xs:minInclusive value="2009-01-01"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="narodeny" type="novyDatum"/>
</xs:schema>
```

```
<narodeny>2009-02-29</narodeny>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml:2: element narodeny: Schemas validity
error : Element 'narodeny' [ST 'mojDatum']: The
character content is not valid.
zoznam.xml fails to validate
```

Reštrikcia typu "minimálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="novyDatum">
    <xs:restriction base="xs:date">
      <xs:minInclusive value="2009-01-01"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="narodeny" type="novyDatum"/>
</xs:schema>
```

```
<narodeny>2008-12-31</narodeny>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml:2: element narodeny: Schemas validity
error : Element 'narodeny' [ST 'mojDatum', facet
'minInclusive']: The value is not facet-valid.
zoznam.xml fails to validate
```

Reštrikcie

- "meno" a "priezvisko": tokeny (min. 3 a max. 20 znakov) ✓
- "dátum narodenia": dátum (> 1.1.2009) ✓
- "miesto narodenia": token (max. 40 znakov) ✓
- "Žije": boolean
- "váha": kladný decimal (v rozmedzí od 1.0 do 7.0 kg)
- "výška": kladný integer (v rozmedzí od 20 do 80 cm)

Reštrikcia typu "maximálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="detskaVaha">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="7"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="vaha" type="detskaVaha"/>
</xs:schema>
```

```
<vaha>3.820</vaha>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml validates
```

Reštrikcia typu "maximálna hodnota"

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleType name="detskaVaha">
    <xs:restriction base="xs:decimal">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="7"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="vaha" type="detskaVaha"/>
</xs:schema>
```

```
<vaha>4800</vaha>
```

```
$ xmllint --noout --schema moja_schema.xsd zoznam.xml
```

```
zoznam.xml:2: element vaha: Schemas validity error :
Element 'vaha' [ST 'detskaVaha', facet
'maxInclusive']: The value is not facet-valid.
zoznam.xml fails to validate
```

Reštrikcie

- "meno" a "priezvisko": tokeny (min. 3 a max. 20 znakov) ✓
- "dátum narodenia": dátum (> 1.1.2009) ✓
- "miesto narodenia": token (max. 40 znakov) ✓
- "Žije": boolean
- "váha": kladný decimal (v rozmedzí od 1.0 do 7.0 kg) ✓
- "výška": kladný integer (v rozmedzí od 20 do 80 cm) ✓