XML Schema — пушкице

Милош Утвић, misko@fil.bg.ac.rs , 27. јун 2024.

Простори имена и коренски елемент шеме

`xmlns:xs = "http://www.w3.org/2001/XMLSchema"`
Користи се у дефиницији шеме.

`xmlns:xsi =`
`"http://www.w3.org/2001/XMLSchema-instance"`
Користи се за везу XML документа и шеме.

`xs:schema`
коренски елемент шеме. Атрибути:
`elementFormDefault`=
`qualified | unqualified`
`attributeFormDefault`=
`qualified | unqualified`
Име елемента или атрибута је квалификовано ако се испред њега обавезно наводи префикс простора имена (подразумевано не мора, тј. `unqualified`, али у пракси се увек користи `qualified` испред елемената и никад испред атрибута).

Елементи и атрибути

Дефиниције и референце

`xs:element`
дефиниција елемента или референца на дефиницију (име) елемента. Атрибути:
`name`= име елемента (у дефиницији)
`type`= тип елемента (у дефиницији)
`ref`= референца на дефиницију или групу
`use='prohibited'` укидање елемента.
`minOccurs`= најмањи број појављивања
`maxOccurs`= највећи број појављивања
Максимална вредност је `unbounded`.

`xs:attribute`
дефиниција атрибута. Атрибути су исти као за `xs:element` осим minOccurs, maxOccurs.

Елемент се дефинише једном у шеми, а на дефиницију се може реферисати произвољан број пута.

Основни уграђени типови података

Универзални тип

`xs:anyType`
тип који прихвата било који модел садржаја.

Текстуални типови

`xs:string`
основни текстуални тип (чува белине).

`xs:normalizedString`
нормализован текстуални тип (белине које нису размаци се током обраде замењују размацима).

`xs:token`
токенски тип. Исти као нормализован текстуални тип, али се почетне и крајње белине уклањају, а вишеструке узастопне белине замењују једним размаком.

`xs:language`
вредности су кодови језика дефинисани стандардом RFC 5646 (IETF BCP 47[1]), на пример `en` (енглески), `sr` (српски), `sr-Cyrl` (српски, ћирилица), `sr-Latn` (српски латиница), `sr-Latn-RS` (српски, латиница, Србија) итд.

`xs:anyURI`
произвољан URI (у пракси најчешће URL, односно апсолутна адреса ресурса на интернету).

Нумерички типови

`decimal`
децимални тип. Произвољни бројеви са коначним децималним записом, укључујући како реалне бројеве, тако и целе (без децимала) са необавезним знаком испред (+, −). Белине нису дозвољене унутар броја, док се околне белине игноришу. Нису дозвољени ни сепаратори хиљада ни научна нотација (са експонентом).

`xs:integer`
целобројни тип са истим ограничењима као `decimal`.
Примери вредности: `123 0 -123`. Подтипови: `xs:positiveInteger` (позитивни цели бројеви $n > 0$ или $n \geq 1$), `xs:negativeInteger` (негативни цели бројеви $n < 0$ или $n \leq -1$), `xs:nonPositiveInteger` (непозитивни цели бројеви $n \leq 0$), `xs:nonNegativeInteger` (ненегативни цели бројеви $n \geq 0$),

`xs:float`
реални тип једноструке тачности. Дозвољава и научну нотацију, али само целобројни експонент. Примери вредности: `123.45 0.0` `−123.45 0.3E+2` (= `30.0`) `45E-3` (=`0.045`)

`xs:double`
реални тип двоструке тачности.

`xs:bool`
логички (буловски) тип. Има вредности `true` и `false`, односно бинарне цифре 1 и 0 (зато је нумерички тип).

Типови за датум и време

`xs:dateTime`
временски тренутак описан датумом и временом у формату `YYYY-MM-DDThh:mm:ss` стандарда ISO 8601 (са необавезним делом за временску зону на крају записа), где сва слова (осим `T`) представљају декадне цифре са следећим значењем:

`YYYY` година (0000–9999)
`MM` месец (01–12)
`DD` дан (01–31)
`hh` час (00–23)
`mm` минути (00–59)
`ss` секунде (00–59)

Слово `T` је сепаратор записа датума и записа времена. На крају се може навести и часовна зона. Следећа три записа се односе на исти временски тренутак:
`2024-06-24T18:36:52+02:00`
`2024-06-24T16:36:52Z`

[1] https://www.iana.org/assignments/language-subtag-registry/language-subtag-registry

`2024-06-24T16:36:52+00:00` (последња два означавају време у западноевропској временској зони која је и нулта, односно почетна зона; први запис означава време у нашој, средњоевропској или првој зони у тренутку када је на снази летње рачунање времена, па је разлика између поменутих два сата уместо уобичајеног једног). Слично, записи `2023-12-24T17:36:52+01:00` `2023-12-24T16:36:52Z` `2023-12-24T16:36:52+00:00` су исти (у овом примеру разлика је један сат јер је у зимском периоду у нашој, средњоевропској или првој зони на снази зимско рачунање времена).

`xs:date`
: датум по грегоријанском календару у формату `YYYY-MM-DD` са необавезним делом за временску зону. На пример, `2024-06-24` `2024-06-24Z` `2024-06-24+00:00`

`xs:time`
: време у формату `hh:mm:ss` са необавезним делом за временску зону. На пример, `18:36:52` `18:36:52+02:00` `16:36:52Z` `16:36:52+00:00`

Модели садржаја

XML Schema другачије дефинише моделе садржаја елемената у односу на DTD:

`empty content`
: празан садржај. Исто као у DTD-ју, имају га само празни елементи.

`simple content`
: једноставан садржај. Имају га елементи који садрже искључиво текст (ово је уже у односу на једноставан садржај у DTD-ју).

`complex content`
: сложен садржај. Имају га они елементи који садрже искључиво (под)елементе као своју децу (у DTD-ју је такав садржај такође третиран као једноставан).

`mixed content`
: Имају га елементи који садрже и текст и (под)елементе као своју децу.

Разумевање модела садржаја је кључно за разумевања дефиниција простих и сложених корисничких типова података (`xs:simpleType` и `xs:complexType`).

Именовани кориснички типови података

Прости типови података

Прости типови података се користе искључиво као типови атрибута и оних елемената у XML документу који имају једноставан садржај (само текст за садржај) и немају атрибуте.

`xs:simpleType`
: дефиниција простог типа. Атрибут:
  `name` име типа.
  Дефиниција се заснива на наслеђивању (извођењу из) неког уграђеног или претходно дефинисаног корисничког типа. Наслеђивање или извођење се најчешће заснива на сужењу (рестрикцији), али постоје и други начини које нећемо помињати.

Рестрикција једноставног садржаја

`xs:restriction`
: дете елемента `xs:simpleType` помоћу којег се врши сужење постојећег типа. Атрибут:
  `base` име (базног) типа који се сужава.
  Базни тип може да буде или прост тип (xs:simpleType) или сложени тип (xs:complexType) настао проширивањем једноставног модела садржаја (`simple content`) атрибутима. Само сужавање (наслеђивање, извођење) се обавља помоћу неког фасета, елемента који је дете елемента `xs:restriction`.

Фасети

Сви фасети имају исти атрибут:
  `value` вредност фасета.
Вредност фасета зависи од његовог значења (имена). Следећи фасети се примењују на све базне типове:

`xs:enumeration`
: набрајање могућих вредности базног типа

`xs:length`
: тачан број карактера (дужина) базне вредности

`xs:minLength`
: најмањи број карактера (дужина) базне вредности

`xs:maxLength`
: највећи број карактера (дужина) базне вредности

`xs:pattern`
: регуларни израз који базна вредност мора да задовољи. Односи se на вредност у целини, те није потребно и погрешно је користити метакарактере који у Пајтону означавају почетак (^) и крај ниске ($). Може се користити више пута у оквиру исте рестрикције, довољно је да вредност задовољи један од наведених регуларних израза да би била валидна.

Следећи фасети се примењују **само** на нумеричке базне типове:

`xs:minInclusive`
: најмања достижна доња граница

`xs:minExclusive`
: недостижна доња граница

`xs:maxInclusive`
: достижна горња граница

`xs:maxExclusive`
: недостижна горња граница

`xs:totalDigits`
: највећи дозвољен број цифара

`xs:fractionDigits`
: највећи дозвољен број децимала

За тачан број цифара и децимала мора се користити `xs:pattern`.

Хијерархија типова

Сви типови података (уграђени и кориснички) су део једног дрвета у чијем корену је универзални тип `xs:anyType`. Основно дете корена типа је универзалан прост тип `xs:anysimpleType` из кога су изведени сви набројани уграђени типови, као и кориснички дефинисани прости типови. Остала деца корена су сви кориснички дефинисани сложени типови података.

Сложени (кориснички) типови података

Сложени типови података се користе за дефиницију типова који се не могу дефинисати као прост тип, тј. не односе се ни на вредности атрибута, нити на вредности елемената који садрже само текст и немају атрибуте. Према томе, **модел садржаја сложеног типа може бити једноставан или сложен**, иако његово име сугерише само овај други. Наиме, на основу дефиниција модела садржаја и простих типова следи да елементи који садрже само текст, а имају атрибуте, имају једноставан садржај који се не може описати простим типом. Исто важи и за празне елементе јер они, по правилу, морају да садрже атрибуте (иначе не би служили ничему), па се ни они не могу описати простим типом.

`xs:complexType`
: дефиниција простог типа. Атрибут:
  `name` име типа.
  `mixed` мешовит садржај? Вредности:
  `true | false`
Главни (под)елементи:
  `xs:simpleContent`
  `xs:complexContent`
  `xs:attribute` референца на атрибут.

`xs:simpleContent` се користи за опис елемента једноставног садржаја (садржај је једино текст) коме се додају само атрибути. Атрибути се додају механизмом проширивања (екстензије). Такође је дозвољен и механизам рестрикције једноставног садржаја сложеног типа на исти начин као за прости тип.

`xs:complexContent` се користи за опис елемента сложеног садржаја (тај садржај чине поделементи), али се у појединим случајевима може и изоставити. Наиме, када се креира нови сложени тип за који се наводи да је изведен из неког постојећег типа, подразумева се да је тај базни тип `xs:anyType` и тада је довољно у оквиру `xs:complexType` употребити само неки од помоћних поделемената (композитора) без навођења елемента `xs:complexContent`:
  `xs:sequence` уређена листа елемената
  `xs:choice` избор једног од наведених елемената
  `xs:all` неуређена листа елемената
  `xs:group` груписање елемената
Другим речима, следеће две дефиниције имају исто значење, али се користи само прва пошто је једноставнија за писање:

```
<xs:complexType name="t">
   <xs:sequence>
       ...
   </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="t">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:sequence>
           ...
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
</xs:complexType>
```

Исто важи и за остале помоћне поделементе елемента `xs:complexType` (`xs:sequence`, `xs:all`, `xs:group`) који се још називају композитори зато што сваки од њих може да буде поделемент било ког од њих (дозвољена је њихова произвољна композиција).
У случају када се сложени тип дефинише помоћу неког постојећег типа, примењује се рестрикција или проширивање (екстензија). Да ли нови сложени тип садржи `xs:simpleContent` или `xs:complexContent` зависи од тога да ли је модел садржаја једноставан (само текст) или сложен (садржи поделементе као децу) без обзира да ли садржи или не атрибуте.

Проширивање једноставног садржаја

`xs:extension`
: дете елемента `xs:simpleContent` или `xs:complexContent`. У првом случају проширује постојећи тип једностав- ног садржаја, у другом случају постојећи тип сложеног садржаја. Атрибут:
  `base` име (базног) типа који се проширује.
Само проширивање се врши набрајањем референци на додате или укинуте елементе и атрибуте у оквиру елемента `xs:extension`. Укидање елемента или атрибута се врши навођењем `use='prohibited'` у оквиру референце.

Сложени тип мешовитог садржаја

У питању је тип елемента који садржи и текст и поделементе као децу. У том случају је јасно да је у питању сложени модел садржаја (`xs:complexContent`) и потребно је навести `mixed='true'` у отвореној етикети елемента `xs:complexType`.