Valsts reģionālās attīstības aģentūra



|  |  |
| --- | --- |
|  |  |

----

versija

Rīgā 2016

|  |  |  |
| --- | --- | --- |
|  | **IEGULDĪJUMS TAVĀ NĀKOTNĒ** |  |

Dokumenta identifikācija

|  |  |
| --- | --- |
| Dokumenta ID: | -----V- |
| Dokumenta nosaukums: | .  .  .  . |
| Dokumenta kods: | ---- |
| Versija: | Versija , Laidiens (saīsināti V ) |

Saskaņojumi

|  |  |  |  |
| --- | --- | --- | --- |
| Organizācija | Vārds, uzvārds, amats | Datums | Paraksts |
|  | Atbildīgā persona no Pasūtītāja puses |  |  |
|  | J.Korņijenko, projekta vadītājs par tehniskiem jautājumiem no Izpildītāja puses |  |  |
|  | M.Pētersons, projekta vadītājs par administratīviem jautājumiem no Izpildītāja puses |  |  |
|  | E.Blumberga, projekta kvalitātes kontroles vadītāja |  |  |

Izmaiņu vēsture

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versija | Datums | Apraksts | Organizācija | Autors |
| 0.08 | 24.02.2006. | Izveidota dokumenta sākotnējā versija | SIA „ABC software” | J.Korņijenko |
| 0.09 | 06.03.2006. | Veiktas šādas izmaiņas:   1. Papildināta jaunas XML shēmas publicēšanas kārtība. 2. Izlabota prasība PR-1. 3. Izlabota personas koda XML shēma. 4. Izlabota pasta indeksa XML shēma.   Izlabota URN XML shēma. | SIA „ABC software” | J.Korņijenko |
| 0.12 | 11.08.2006. | Dokumenta aktualizācija saskaņā ar EPAK ieviešanas pilotprojekta e-pakalpojumu arhitektūras vadlīniju izstrādi | SIA „ABC software” | J.Korņijenko |
| 1.00 | 07.12.2006. | Redakcionālas izmaiņas pirms dokumenta publicēšanas | SIA „ABC software” | J.Korņijenko |
| 1.01 | 30.05.2008. | Veiktas šādas izmaiņas:   1. Papildināta un izlabota XML shēmas ievietošanas un publikācijas kārtībā IVIS XML shēmas katalogā 2. Papildinātas un izlabotas prasības AR-03, AR-04, AR-05, NK-03, NK-04, SK-01, SK-06 3. Pievienota prasība SK-16   Citas redakcionālas un saturiskas izmaiņas | SIA „ABC software” | J.Korņijenko |
| 1.02 | 12.01.2012. | Redakcionālas izmaiņas | SIA „ABC software” | J.Korņijenko |
| 1.03 | 21.02.2012. | Iestrādātas izmaiņas saskaņā ar 23.01.2012. Nodevuma "Vadlīnijas" izvērtējumu Nr.11 (VRAA-6\_15\_11\_58-VISS\_2010-NI-11-VDL-V1.00-23.01.2012.). | SIA „ABC software” | M.Gasparoviča |
| 1.04 | 09.10.2015 | Ņemti vērā Pasūtītāja komentāri un aktualizētas 2.1. un 2.2. nodaļas, 3. un 4. attēli, kā arī 2.4.1. un 3.1.6. sadaļas. | SIA „ABC software” | J.Korņijenko |
| 1.05 | 04.12.2015 | Ņemti vērā Pasūtītāja komentāri veikti redakcionāli labojumi visā dokumentā. | SIA „ABC software” | J.Korņijenko |
| 2.00 | 01.07.2016 | Veikti redakcionāli labojumi 1.3. nodaļā. | SIA „ABC software” | J.Korņijenko |

**Satura rādītājs**

[Attēlu saraksts 7](#_Toc455392549)

[1. Ievads 8](#_Toc455392550)

[1.1. Dokumenta nolūks 8](#_Toc455392551)

[1.2. Termini un pieņemtie apzīmējumi 8](#_Toc455392552)

[1.2.1. Termini 8](#_Toc455392553)

[1.2.2. Apzīmējumi 9](#_Toc455392554)

[1.3. Saistība ar citiem dokumentiem 9](#_Toc455392555)

[1.4. Dokumenta pārskats 10](#_Toc455392556)

[2. XML shēmas jēdziens 11](#_Toc455392557)

[2.1. Citu shēmu lietošanas ierobežojumi 11](#_Toc455392558)

[2.2. Aprakstu uzmetums 11](#_Toc455392559)

[3. XML shēmu izstrādes process 12](#_Toc455392560)

[3.1. Jaunas XML shēmas izstrāde 12](#_Toc455392561)

[3.2. Jaunas XML shēmas ievietošana un publicēšana 13](#_Toc455392562)

[3.3. Esošas XML shēmas jaunas versijas izveide 18](#_Toc455392563)

[3.4. VISS XML resursu kataloga saskarne 18](#_Toc455392564)

[3.4.1. XML resursu tipi 18](#_Toc455392565)

[4. XML shēmu izstrādes vadlīnijas 20](#_Toc455392566)

[4.1. XML shēmu arhitektūras prasības 20](#_Toc455392567)

[4.1.1. Primārā shēmu valoda 20](#_Toc455392568)

[4.1.2. Shēmu sarežģītība 20](#_Toc455392569)

[4.1.3. Datu modelis, nevis formas 21](#_Toc455392570)

[4.1.4. XML shēmas arhitektūras dizains 21](#_Toc455392571)

[4.1.5. Import un include lietošana 23](#_Toc455392572)

[4.1.6. Arhitektūras XML shēmu lietošana 23](#_Toc455392573)

[4.2. Nosaukumu konvencija 24](#_Toc455392574)

[4.3. XML shēmas kodēšana 27](#_Toc455392575)

[4.3.1. Vārdtelpas (namespace) 27](#_Toc455392576)

[4.3.2. Versiju piešķiršana shēmām 28](#_Toc455392577)

[4.3.3. id atribūts schema elementā 29](#_Toc455392578)

[4.3.4. id atribūta izmantošana shēmas elementā 29](#_Toc455392579)

[4.3.5. elementFormDefault un attributeFormDefault lietošana 29](#_Toc455392580)

[4.3.6. Datu tipi .v. elementu deklarācija 31](#_Toc455392581)

[4.3.7. Globālās definīcijas 32](#_Toc455392582)

[4.3.8. Elementi .v. Atribūti 32](#_Toc455392583)

[4.3.9. Shēmu dokumentēšana 33](#_Toc455392584)

[4.3.10. Shēmu atkārtotās lietošanas īpatnības 33](#_Toc455392585)

[4.3.11. XML shēmu mantošana 33](#_Toc455392586)

[4.3.12. Elementu datu saturs 34](#_Toc455392587)

[4.3.13. Alternatīvo nosacījumu reprezentācija 34](#_Toc455392588)

[4.3.14. string, normalizedString, token un citi W3C XML shēmas tipi 34](#_Toc455392589)

[4.3.15. Saraksti 35](#_Toc455392590)

[4.3.16. XML shēmas satura ierobežojumi 36](#_Toc455392591)

[4.4. Atribūtu lietošana 36](#_Toc455392592)

[4.4.1. default un fixed atribūtu lietošana 36](#_Toc455392593)

[4.4.2. Lokalā .v. Globalā atribūtu definēšana 37](#_Toc455392594)

[4.5. Klasificējamās vērtības un pārskaitāmie tipi 37](#_Toc455392595)

[4.5.1. Klasificējamo vērtību kopas prezentācija 37](#_Toc455392596)

[4.5.2. Teksts .v. Kods 38](#_Toc455392597)

[4.6. XML shēmas publicēšana 38](#_Toc455392598)

[4.6.1. VISS XML shēmu kataloga izmantošana 38](#_Toc455392599)

[4.6.2. XML shēmas metadati 38](#_Toc455392600)

[4.6.3. XML shēmas metadatu elementi 40](#_Toc455392601)

[5. Pielikumi 42](#_Toc455392602)

[5.1. Pielikums Nr.1 42](#_Toc455392603)

# Attēlu saraksts

[1.attēls. Pilnteksta XML shēmas meklēšanas saskarne 11](#_Toc455392604)

[2. attēls. XML shēmas pievienošana 13](#_Toc455392605)

[3.attēls. VISS XML resursu kataloga diskusijas sadaļa 14](#_Toc455392606)

[4.attēls. XML shēmas izveidošana un pilnveidošana (vienas versijas ietvaros) 15](#_Toc455392607)

[5.attēls. XML shēmas publikācija 16](#_Toc455392608)

[6.attēls. XML shēmas standarta tipu hierarhijas diagramma 34](#_Toc455392609)

[7.attēls. Pieļaujamais elementu skaits tiek definēts xs:sequence sekcijā 34](#_Toc455392610)

[8.attēls. Pieļaujamais elementu skaits tiek definēts elementā 35](#_Toc455392611)

# Ievads

Informācijas sistēmu, e-pakalpojumu izstrādes un integrācijas uzdevumus nav iespējams iedomāties bez datu semantiskas un sintaksiskas saskaņošanas, kas mūsdienās ir pamats jebkuram veiksmīgam IT projektam valsts un pasaules mērogā. Šim mērķim Valsts informācijas sistēmu savietotāja (VISS) projekta ietvaros ir izvēlēta W3C XML shēmu valoda, kas atbalsta vārdtelpas, datu tipu definēšanu un moduļshēmu projektēšanu.

Dokumenta pēdējā versija ir pieejama VISS portālā adresē <https://viss.gov.lv>.

## Dokumenta nolūks

Dokuments satur vadlīnijas un noteikumus valsts integrācijas projektu ietvaros izmantojamo XML shēmu izstrādei. Aprakstītās vadlīnijas un noteikumi iekļauj obligātos nosacījumus, kuri tiek izvirzīti XML shēmu struktūrai un saturam, kā arī izstrādes rekomendācijas. Esošais dokuments ir domāts Web servisu un XML shēmu izstrādātājiem, kas darbojas Latvijas Republikas valsts un komercsektorā.

## [Termini](#_Toc65487056) un pieņemtie apzīmējumi

### Termini

Dokumentā izmantotie termini ir apkopoti 1.tabulā.

1.tabula

Termini

| Termins | Apraksts |
| --- | --- |
| Entītija | Šajā dokumentā ar entītiju ir domāta kāda fiziska persona, valsts iestāde, organizācija vai komercuzņēmums. |
| Informācijas sistēma | Datu bāzes, lietotāja programmatūras, procedūru un funkciju kopums un to savstarpējās attiecības, kas nodrošina sistēmas funkcionēšanu atbilstoši tās pamatmērķiem un pamatuzdevumiem. |
| XML dokuments | XML dokuments ir labi noformēts (*well-formed*) un pabeigts XML vienums, kā to definē W3C rekomendācijas. Sadarbspējas prasību dēļ daudzi dokumenti tiek sūtīti kā ziņojumi starp datoru sistēmām un tiek saukti par XML ziņojumiem. |
| XML shēmas elements | XML shēmas elements vai vienkārši shēmas elements ir abstrakta datu struktūra, kura nepieciešama, lai pārbaudītu (validētu) XML dokumenta elementu. XML shēmas elements ir izteikts XML valodā un aprakstīts saskaņā ar W3C rekomendācijām. |
| XML shēmas dokuments | XML shēmas dokuments, shēmas dokuments, XML shēma vai shēma ir XML shēmas datne (.xsd), kas satur vienu vai vairākus XML shēmas elementus un kas definē XML dokumenta vai ziņojuma saturu. XML shēmu izstrādes vadlīniju dokumenta ietvaros termini XML shēmas dokuments, shēmas dokuments, XML shēma vai shēma tiek lietoti ar vienu un to pašu nozīmi, norādot uz shēmas elementu kopu; tie izteikti XML valodā un aprakstīti saskaņā ar W3C rekomendācijām. |
| Arhitektūras XML shēma | Arhitektūras XML shēma vai arhitektūras shēma ir XML shēmas dokuments, kas nodrošina atkārtoti izmantojamo (*reusable*) saturu vēlreizējai lietošanai ziņojumu shēmās. Arhitektūras XML shēmas piemērs ir PersonCode - vienkāršais datu tips. |
| XML ziņojuma shēmas dokuments | XML ziņojuma shēmas dokuments ir shēmas dokuments, kas definē XML dokumenta struktūru un kontekstu. Šeit tiek lietots termins ”ziņojuma shēma” (nevis ”dokumenta shēma”), lai nejauktu ar ”shēmas dokumentu” un ”dokumenta shēmu”, kas nozīmē divas dažādas lietas. |
| XML elements vai XML dokumenta elements | XML elements vai XML dokumenta elements ir XML dokumenta fragments, kas ir validēts ar kādu ziņojuma shēmu. Ja šis fragments ir XML dokumenta elements, tad dokuments tiek saukts par XML dokumenta elementu. Jāpiezīmē, ka ar arhitektūras shēmām XML dokumenti netiek veidoti tieši, tās ir paredzētas atkārtotai izmantošanai ziņojumu shēmu dokumentos. |
| XML shēmas vārdtelpa (*namespace*) | XML shēmas vārdtelpa nosaka noslēgtu nosaukumu kopu vai vietu, kur shēma (nosaukumu kopa) tiek glabāta. Vārdtelpas definē ar URI (piemēram, URL) un, tās paredzētas ar nosaukumu, piešķiršanu saistītu problēmu risināšanai. Vienas vārdtelpas iekšienē visiem nosaukumiem jābūt unikāliem, lai gan citā vārdtelpā attiecīgajam nosaukumam var būt cita nozīmē. |
| XML shēmu dokumentu hierarhija vai XML shēmu hierarhija | Tas ir saistīto XML shēmu dokumentu un arhitektūras shēmu kopums, kam ir šādas pazīmes:   * shēmas elementu valodā apraksta kādu priekšmetisko sfēru; * identificē ar vienu vārdtelpas (namespace) identifikatoru. |

### Apzīmējumi

Dokumentā izmantotie apzīmējumi ir apkopoti 2.tabulā.

2.tabula

Apzīmējumi

|  |  |
| --- | --- |
| Apzīmējums | Apraksts |
| IR | PMLP „Iedzīvotāju reģistrs” |
| ISO | Starptautiskā standartizācijas organizācija (*International Organization for Standardization*) |
| PMLP | Pilsonības un migrācijas lietu pārvalde |
| VRAA | Valts reģionālās attīstības aģentūra (Bij,, ĪUMEPLS - Īpašu uzdevumu ministra e-pārvaldes lietās sekretariāts) |
| VISS | Valsts informācijas sistēmu savietotājs (bij. IVIS - integrētā valsts informācijas sistēma) |
| MIME | Internet pasta vairākmērķu paplašinājumi (*Multipurpose Internet Mail Extensions*) |
| UBL | Universālā biznesa valoda (*Universal Business Language*) |
| UN/CEFACT | United Nations Centre for Trade Facilitation and Electronic Business |
| URI | Vienotais resursa identifikators (*Uniform Resource Identifiers*) |
| W3C | Globālā tīmekļa konsorcijs (*World Wide Web Consortium*) |
| W3CDTF | W3C datuma un laika formāti (*W3C Date and Time Formats*) |
| WSDL | Tīmekļa pakalpju aprakstes valoda (*Web Services Description Language*) |
| XML | Paplašināmās iezīmēšanas valoda (*eXtensible Markup Language*) |
| XSD | XML dokumenta struktūras apraksts (*XML Schema Definition Language*) |
| XSL | Izvēršamā stila lapu valoda *(Extensible Stylesheet Language)*, skat. XSLT |
| XSLT | Paplašināmās stila lapas pārveidošanas valoda (*eXtensible Stylesheet Language (XSL) transformations*) |

## Saistība ar citiem dokumentiem

Dokuments ir izstrādāts, balstoties uz sekojošiem dokumentiem:

1. Metadatu un e-pakalpojumu identifikācijas standarts (URN:IVIS:100001:DOC-RCM-META).
2. e-GMS v3.0 Application Profile and Binding. XML Schemas (https://www.oasis-open.org/committees/download.php/7271/eGMS%20version%203.pdf).
3. Global versus Local. A Collectively Developed Set of Schema Design Guidelines (<http://www.xfront.com/GlobalVersusLocal.html>)
4. XML Schema Part 2: Datatypes Second Edition. W3C Recommendation 28 October 2004 (<http://www.w3.org/TR/xmlschema-2/>)
5. Code List Task Group Final Report (http://www.oasis-open.org/committees/download.php/3493/code-proposal2-20030911.html).
6. Multipurpose Internet Mail Extensions (MIME) (https://en.wikipedia.org/wiki/Media\_type)
7. ISO 639-2 (<http://www.loc.gov/standards/iso639-2>)
8. URI (<http://www.ietf.org/rfc/rfc2396.txt> vai <http://purl.org/dc/terms/URI>)
9. W3CDTF (<http://www.w3.org/TR/NOTE-datetime> vai (<http://dublincore.org/2003/03/24/dcq#W3CDTF>)
10. Vienotās veselības nozares informācijas sistēmas darbības paplašināšana. E-veselības ziņojumapmaiņā izmantojamo datu struktūru standarts. NVD.STD.HL7.

## Dokumenta pārskats

Dokumentu veido šādi nodalījumi:

* 1.nodalījumā – „Ievads” – aprakstīta dokumenta kopējā struktūra, nolūks, izmantotie termini un apzīmējumi, kā arī saistība ar citiem dokumentiem.
* 2.nodalījums – „XML shēmas jēdziens” - aprakstīts XML shēmas jēdziens un lietošanas ierobežojumi.
* 3.nodalījumā - „XML shēmu izstrādes process” - aprakstīts XML shēmu izstrādes process.
* 4.nodalījumā - „XML shēmu izstrādes vadlīnijas” - aprakstītas XML shēmu izstrādes vadlīnijas.
* 5.nodalījumā iekļauti pielikumi.

# XML shēmas jēdziens

XML shēmas dokuments, shēmas dokuments, XML shēma vai shēma ir XML shēmas datne ar paplašinājumu .xsd, kas satur vienu vai vairākus XML shēmas elementus un kas definē XML dokumenta vai ziņojuma saturu. XML shēmu izstrādes vadlīniju dokumenta ietvaros termini XML shēmas dokuments, shēmas dokuments, XML shēma vai shēma tiek lietoti ar vienu un to pašu nozīmi, norādot uz shēmas elementu kopu; tie izteikti XML valodā un aprakstīti saskaņā ar W3C rekomendācijām. XML datu datnēm ir plašinājums .xml.

Definīciju kopumu, kas ir specifisks kādai priekšmetiskai sfērai, jādefinē piemērotās vārdtelpas iekšienē. Integrācijas uzdevumos reti tiks izmantotas atsevišķas XML shēmas, parasti tiek lietots saistīto XML shēmu kopums, kas apraksta konkrētu priekšmetisko sfēru un ko identificē ar vienu vārdtelpas (*namespace*) identifikatoru. Tātad, veidojot XML shēmu hierarhiju, jāveic to veidojošo XML shēmas dokumentu versiju kopējā kontrole un identificēšana.

Hierarhijas ietvaros XML shēmu savstarpējā izmantošana (sasaiste) notiek ar xs:include palīdzību.

Jaunizveidotā hierarhija būs pieejama no citas arhitektūras shēmas hierarhijas vai ziņojumu shēmas, izmantojot xs:import mehānismu. Visas references uz hierarhiju lietos šīs shēmas vārdtelpu.

## Citu shēmu lietošanas ierobežojumi

Šobrīd labs izstrādāto vadlīniju dokumentu praktiskā pielietojuma piemērs ir VISS XML shēmu katalogā reģistrētās XML shēmas (https://ivis.eps.gov.lv/RC.WebApp/). Neskatoties uz to, ka šis dokuments varētu būt pretrunā ar citām eksistējošām XML shēmām, kas ir izveidotas pirms standarta izstrādes, izmaiņas šajās shēmās nav paredzēts realizēt, lai pilnībā nodrošinātu šo vadlīniju pamatprincipus. Sevišķos gadījumos ārēji definēto shēmu komponentes var tikt kombinētas ar VISS definētām komponentēm tādā veidā, ka dažas komponentes atbilst vadlīnijām, bet dažas nē.

## Aprakstu uzmetums

Vadlīnijas saistībā ar shēmu izveidošanas dažādiem aspektiem tiek iedalītas vairākās grupās:

* „XML shēmu hierarhijas izstrādes un publicēšanas kartība” – nosaka nepieciešamos XML shēmas izstrādes un publicēšanas posmus.
* "XML shēmu izstrādes vadlīnijas” – satur ieteikumus datu projektēšanai, prasības un konvencijas, kas attiecas uz shēmām kopumā.
* "XML shēmu komponenšu izstrādes vadlīnija” – satur prasības un konvencijas, kas attiecas uz individuālās shēmas komponenšu modelēšanu.
* "Metadati un shēmas” – satur prasības un konvencijas, kas attiecas uz metadatu elementu lietošanu shēmās. Metadati šajā kontekstā iekļauj metadatu elementus (tādus kā, shēmas elementa versijas atribūts), kuri ir iebūvēti XML shēmu valodā.

# XML shēmu izstrādes process

XML shēmu izstrādes process loģiski ir sadalāms trīs fāzēs:

1. Jaunas XML shēmas (hierarhijas) izstrāde.
2. Jaunas XML shēmas (hierarhijas) publicēšana.
3. Jaunas versijas izveide esošai XML shēmai (hierarhijai).

## Jaunas XML shēmas izstrāde

Jaunas XML shēmas izstrādes soļi:

1. Veidojot jaunu XML shēmu vai shēmas hierarhiju, tās izstrādātājam ir jāveic shēmas darbības apgabala analīze, t.sk., jāsavāc visa tehniskā informācija par uzdevumu, jāizveido sākotnējais datu modelis, jānosaka iespējamie XML shēmas elementi un atribūti, ja nepieciešams jādefinē dienesta lauki un XML shēmas.
2. Kad visa sākotnējā informācija par veidojamo shēmu ir savākta, nākamais solis ir līdzīgu XML shēmu meklēšana VISS XML shēmu katalogā (https://ivis.eps.gov.lv/RC.WebApp/) pēc atslēgvārdiem vai kategorijām, izmantojot XML shēmas kataloga meklēšanas saskarni (skat. 4.6.1.sadaļu) – pilnteksta vai strukturēto (skat. 1. attēlā). XML shēmu katalogs garantē, ka visas katalogā ievietotās XML shēmas būs pieejamas tiešsaistē un netiks izdzēstas vai pārvietotas citā vietā. Ja nepieciešamā vai tai līdzīgā XML shēma (hierarhija) ir atrasta, tad to būtu iespējams izmantot tieši.



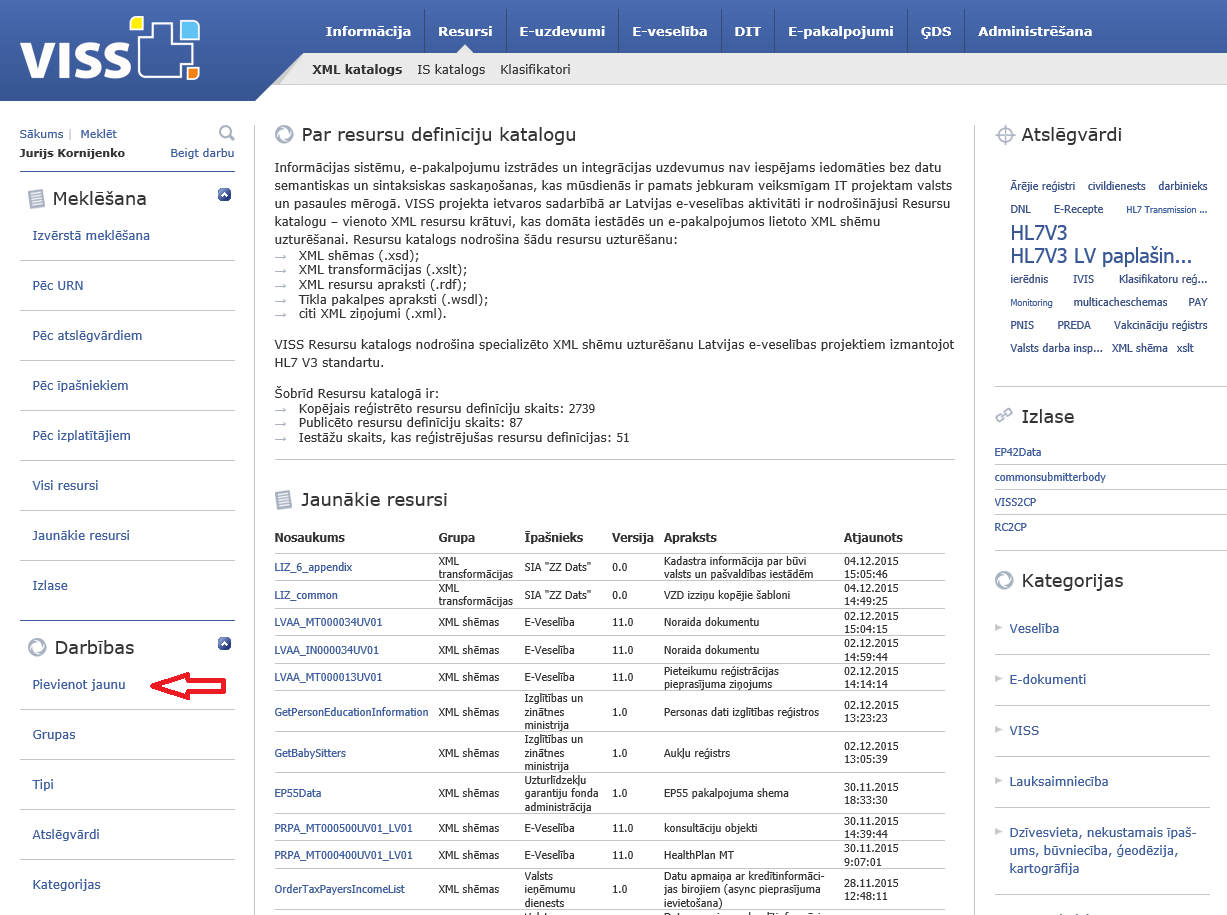
.attēls. Pilnteksta XML shēmas meklēšanas saskarne

1. Bet pastāv liela varbūtība (piemēram, kad mainās datu glabāšanas struktūra, datu formāts utml.), ka esošo shēmu izmantot tieši nevar, un izstrādātāja uzdevums tad būs, ņemot esošo shēmu (hierarhiju) par pamatu, veidot jaunu. Izstrādājot jauno XML shēmu, svarīgi atkal izmantot jau esošās arhitektūras shēmas, piemēram, personas kods, VISS URN, adresi vai citus (skat. 4.1.6. sadaļu).
2. Pēc shēmas sākotnējās versijas izstrādes jāveic tās ievietošana XML shēmu katalogā (skat. 3.2.nodaļu).

## Jaunas XML shēmas ievietošana un publicēšana

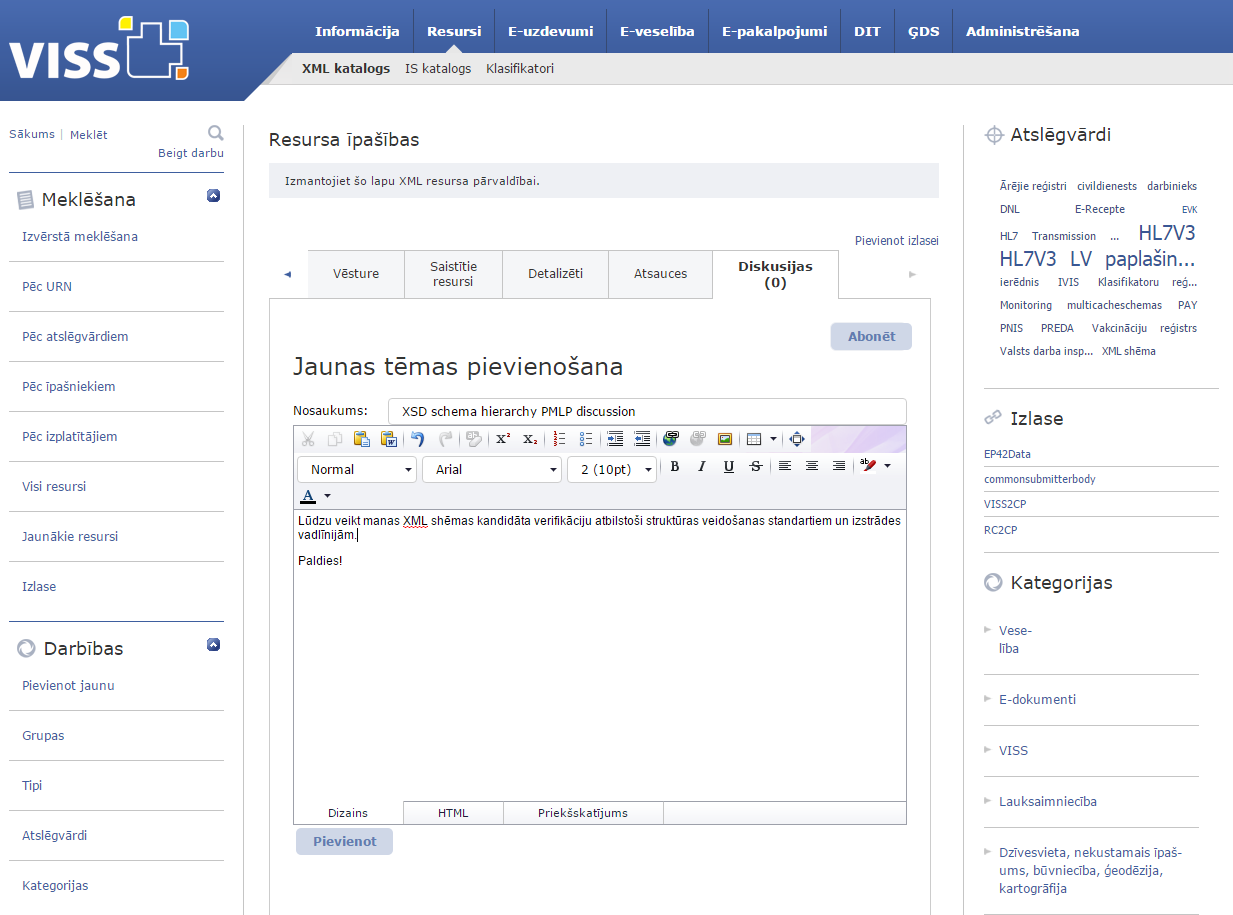
Jaunas XML shēmas (hierarhijas) publicēšana sastāv no vairākiem soļiem. Šie soļi galvenokārt nosaka XML shēmas izveidošanas un publicēšanas organizatoriskos posmus, kas ir nepieciešami, izstrādājot valsts mēroga integrētu informācijas sistēmu. Jaunas XML shēmas izveidošanas un pilnveidošanas kārtība ir redzama 4.attēlā, to veido šādi posmi:

1. Autorizācija VRAA VISS portālā, noklikšķinot “Ieiet” kreisajā augšējā stūrī un izvēloties kādu no piedāvātajām autentifikācijas metodēm (skat. .attēlu):
   1. Lietotāja vārda un paroles autentifikācija, kura piešķirta reģistrētajiem VISS portālā iestādēm un/vai lietotājiem.
   2. Autentifikācija, izmantojot kādu no latvija.lv autentifikācijas piegādātājiem, ir iespējama ja reģistrētajiem lietotājiem ir piešķirts autentifikācijas tips – „Cits IdP”.
2. Pēc autorizācijas, nepieciešams piereģistrēt XML shēmu noklikšķinot darbību “Pievienot jaunu”. Tālāk tiek piedāvātā ceļvežveida saskarne XML shēmas reģistrācijai.



2. attēls. XML shēmas pievienošana

1. XML shēma ir izveidota un saglabāta VISS XML shēmu katalogā ar pazīmi „CANDIDATE” (kandidāts). Šī posma darbība varētu būt atkārtotā vairakkārt – jo paralēli ar shēmas izstrādi (projektēšanu) varētu tikt veikta timekļa pakalpes vai e-pakalpojuma izstrāde, kur šī shēma ir izmantota.
2. Tiklīdz XML shēma ir pabeigta un atrodas jau stabilā stāvoklī, nepieciešams pieprasīt VISS moderatoru XML shēmas kandidāta verifikāciju atbilstoši struktūras veidošanas standartiem un izstrādes vadlīnijām. Tas tiek izdarīts izmantojot VISS XML resursu katalogā pieejamo diskusijas lapu:



.attēls. VISS XML resursu kataloga diskusijas sadaļa

1. Ar XML shēmas autoru tiek apspriesti iespējamie papildinājumi un uzlabojumi, lietojot VISS portāla diskusijas sadaļu, un autors veic nepieciešamos uzlabojumus. Rezultātā, autors izlaboto XML shēmu lejupielādē katalogā.
2. Kad XML shēmas autors novērš visus norādītos trūkumus, VISS moderators piešķir XML shēmai statusu „EDITED” (koriģēšana).



4.attēls. XML shēmas izveidošana un pilnveidošana (vienas versijas ietvaros)

Pēc statusa „EDITED” piešķiršanas XML shēmas autors ir tiesīgs mainīt shēmu, tai skaitā pievienot un labot elementus un tipus (visas darbības tiek veiktas vienas versijas ietvaros).

1. Tiklīdz visas nepieciešamās izmaiņas ir veiktas (parasti pēc attiecīgā integrācijas lietojuma vai e-pakalpojuma nodošanas), tiek pieprasīts XML shēmas publikācijas pieprasījums (skat. 5.attēlā) izmantojot VISS XML resursu katalogā pieejamo diskusijas lapu.



.attēls. XML shēmas publikācija

4. un 5.attēlos atspoguļotās biznesa procesa modelēšanas diagrammas notifikācijas elementi ir aprakstīti 5.1.nodaļā.

1. VRAA veic formālu XML shēmas atzinumu/akceptējumu. Tiklīdz XML shēma vai shēmas hierarhija ir akceptēta, VISS moderators to publicē un tās statuss ir „PUBLISHED” (publicēta).

Svarīgi:

VISS XML shēmu katalogā vienmēr pieejama tikai aktuālā shēma.

XML shēmas izstrādes un pieņemšanas procesā versijas komponente nemainās. Tas ir saistīts ar to, ka versijas komponente ir ievietota XML shēmas vārdtelpā.

Pēc publicēšanas XML shēmu ir aizliegts mainīt. Ja būs nepieciešams ieviest kādas izmaiņas, tas tiks nodrošināts, izveidojot XML shēmas jaunu versiju (jau mainot XML shēmas versijas komponenti).

XML shēmu publicēšanas kārtības ievērošana garantēs to kvalitāti un labas atkārtotas izmantošanas iespējas, kā arī nodrošinās XML struktūras stabilitāti un loģiskumu.

## Esošas XML shēmas jaunas versijas izveide

Versijas izveide esošai XML shēmai atbilst jaunas XML shēmas publicēšanas procesam, bet XML shēmas jaunā versija tiek veidota no jau esošas publicētās XML shēmas vai hierarhijas. Pavisam eksistē divas jaunās XML shēmas versijas izveides iespējas:

1. Pilnība tiek nodublēta visa hierarhija v1-1 ar vajadzīgajām izmaiņām. Šī paņēmiena mīnuss ir tāds, ka visi esošie tipi gan tie, kas tika mainīti, gan tie, kas netika mainīti, veidos jau pavisam citu hierarhiju, ne ar ko nesaistītu ar iepriekšējo. Tādā veidā ieteicams darīt, ja ir paredzēta nopietna pārstrāde.
2. Var v1-1 likt tikai tos tipus, kas pamainījās, veicot hierarhijas v1-0 importu. Skat. PMLP IR piemēru: http://ivis.eps.gov.lv/XMLSchemas/100010/PopulationRegister/v1-2/AddressResponse.xsd. Šeit arī ir shēmu atkal izmantošana no v1-0 un v1-1.

## VISS XML resursu kataloga saskarne

VISS XML resursu katalogs ir atrodams VISS portālā (<https://viss.gov.lv/>), izvēloties galvenajā izvēlnē *Resursi >> XML Katalogs*. Jaunas XML shēmas lejupielāde katalogā ir iespējama tikai pēc tās reģistrācijas, saņemot privilēģiju „XML shēmas uzturētājs”. Iestādes un lietotāja reģistrācijas forma ir pieejama VISS portālā tiešsaistes režīmā.

### XML resursu tipi

VISS XML resursu katalogs nodrošina dažādu resursu grupu uzturēšanu:

* XML – parastie XML ziņojumi un dokumenti, kas varētu kalpot par konkrētas XML shēmas piemēriem utt
* XSLT – XML ziņojumu transformācijas:
* “IVIS transformācija” - šis transformācijas tips tiek nodrošināts saderībai ar VISS XML resursu kataloga iepriekšējo versiju, jaunajām transformācijām jālieto "XSLT2.0 transformācija", "XSLT3.0 transformācija" vai “Cita transformācija” tipi;
* “XSLT2.0 transformācija” – uz XSLT 2.0 standarta bāzētā transformācija;
* “XSLT3.0 transformācija” – uz XSLT 3.0 standarta bāzētā transformācija (lieliem apjomiem);
* “Cita transformācija” – XSLT 1.0 un 1.1 transformācijas.
* WSDL – tīmekļa pakalpes apraksts;
* XSD – XML shēmas, izšķir vairākus XML shēmas tipus:
* "IVIS XML shēma” – šī vadlīniju pamata priekšmets, savukārt parasti dalās:
* arhitektūras XML shēmas – persona, adrese, juridiskā persona, identifikatori, kontaktinformācija un citas, kas apraksta plaši lietojamas entītijas un datu objektus un, kas parasti ir atkal izmantojamas citās XML shēmās;
* lietojumu XML shēmas – XML shēmas, kas orientētas uz konkrēto biznesa uzdevumu;
* e-pakalpojumu XML shēmas – speciāls XML shēmas veids, kas nodrošina VISS e-pakalpojumu darbību un parasti arī tiek mantots no e-pakalpojuma aploksnes XML shēmas.
* “HL7 XML schema” – uz e-veselības HL7 standarta bāzētas XML shēmas, skat. [11].
* “Cita XML shēma” – citas, nesavietojamas ar esošām vadlīnijām, XML shēmas.

# XML shēmu izstrādes vadlīnijas

Katras vadlīnijas apraksts ir sadalīts sešās sekcijās:

1. XML shēmu arhitektūras prasības;
2. Nosaukumu konvencija;
3. XML shēmas kodēšana;
4. Atribūtu lietošana;
5. Klasificējamās vērtības un pārskaitāmie (enumerated) tipi;
6. XML shēmas publicēšana.

## XML shēmu arhitektūras prasības

### Primārā shēmu valoda

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-01 |
| Ieteikuma būtība | W3C XML shēma jālieto kā galvenā shēmu valoda XML dokumentu aprakstīšanai. Visām XML shēmām jāizmanto UTF8 kodējums. |
| Obligātums | Obligāta |
| Paskaidrojums | Valsts iestāžu sadarbspējai ir svarīgi, lai visi lietotu vienādu shēmas valodu, pretējā gadījumā – vienādo komponenšu sintaksiski-specifiskās definīcijas nav iespējams, atkārtoti lietot, kas ir galvenais VISS mērķis. W3C XML shēmu (*XML Schema*) valoda ir izvēlēta par VISS XML dokumentu aprakstīšanas pamata valodu, jo tā atbalsta vārdtelpas, datu tipa definēšanu un moduļshēmu projektēšanu.  Ir paredzēti piemēri, kuros shēmas, saistībā ar to izstrādes mērķiem, tiks pielāgotas dažādiem lietošanas veidiem. Ja shēma ir izstrādāta vispārējai lietošanai, tad tā neierobežo XML dokumenta elementu atšķirībā no tādas, kas izstrādāta specifiskai lietošanai. Konkrētos lietošanas gadījumos nepieciešams vispārīgi lietojamai shēmai noteikt papildus ierobežojumus, lai pārliecinātos, ka shēmas elements veic visas paredzētās darbības. |
| Piemēri | <?xml version="1.0" encoding="utf-8"?>  <xs:schema xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0" id="Identification.xsd"> |

### Shēmu sarežģītība

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-02 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Nepieciešams atcerēties savu mērķauditoriju. Mazāk izplatītu XML shēmu nodrošinātās iespējas nerekomendē lietot tur, kur pastāv vienkāršākas alternatīvas. Shēmu izstrādātājiem jāvadās pēc citiem VISS XML shēmu piemēriem, īpaši tiem, kuri tika izstrādāti centralizēti, lai palīdzētu noteikt atbilstošu stilu. Shēmu izstrādātājiem jāņem vērā viņu izstrādāto shēmu testējamība. |
| Paskaidrojums | W3C XML shēmas valoda pieļauj milzīgas iespējas un elastīgumu shēmu definēšanas jomā. Vairumā gadījumu mērķa sasniegšanai shēmas varētu izstrādāt vienkāršā vai sarežģītā veidā. Bet, ņemot vērā, ka tās ir jaunas tehnoloģijas un izstrādātājiem, kuri skatīs šīs shēmas, būs maza pieredze, ir ieteicamāk tās veidot vienkāršas.  Shēmu izstrādes un testēšanas rīkiem piemīt kļūdas, pārsvarā - mazāk lietotajos XML shēmu aspektos.  Vienkāršās shēmas parasti ir arī vieglāk testēt. |

### Datu modelis, nevis formas

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-03 |
| Obligātums | Vēlams |
| Ieteikuma būtība | XML shēmām ir jāmodelē pamata dati, kas lietojumam nepieciešami vairāk nekā eksistējošās formas (veidlapas), eksistējošie ziņojumu apmaiņas formāti vai jāatkārto esošās relāciju datu tabulas struktūras. Lai gan eksistējošās formas un ziņojumu formāti bieži noder, kā labi shēmu veidošanas sākumpunkti, tie nevar dominēt fināla XML shēmas vai XML shēmas hierarhijas projektējumā. |
| Paskaidrojums | Tam ir trīs iemesli:   * labi projektēta forma (veidlapa) ir paredzēta lietošanai uz papīra, nevis datora ekrānā, līdz ar to, eksistējošam apmaiņas ziņojuma formātam varētu būt vai arī nebūt iespējams precīzi izpildīt informācijas prasības; * XML shēmas projektējumu nepieciešams veidot, ņemot vērā e-pakalpojumu informācijas, biznesa un darbības modeļus, nevis konkrēta procesa vai sistēmas prasības. * Pareizi uzprojektēta XML shēma (hierarhija) nodrošina hierarhisko tipu lietošanu: *Structure1* ietver sarakstu ar *Structure2*, kas savukārt varētu ietvert *Structure3* utt. Šajā gadījumā *Structure1, Structure2, Structure3* reprezentē attiecīgās datu entītijas, bet kopumā hierarhija apraksta lietojumapgabalu. |

### XML shēmas arhitektūras dizains

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-04 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Eksistē vairākas XML shēmu arhitektūras dizaina pieejas (skat. [4]), tai skaitā:   * *Russian Doll* („matreška”). XML shēmas dizaina struktūra atspoguļo XML dokumenta struktūru. XML shēmas saknes elements satur anonīmo kompleksa tipu, kas savukārt ietver citus elementus, anonīmos tipus utt. * *Salami Slice*. XML dokuments tiek sadalīts komponentēs, kur katra komponente tiek izpausta ar elementu. Ja nepieciešams piesaistīt kādu citu elementu, tiek lietotas references uz šiem elementiem. * *Venetian Blind*. Līdzīgs Salami *slice*, vienīgi elementu vietā tiek lietotas parasto un komplekso tipu deklarācijas.   Ja vien iespējams, būtu jālieto *Venetian Blind* XML shēmu izstrādes stils. Varētu pielietot arī vairāku metodoloģiju apvienojumu. XML shēmās, kas potenciāli varētu tikt sauktas par Arhitektūras XML shēmām (skat. 3.4.nodaļu), jālieto *Venetian Blind* izstrādes stils. |
| Paskaidrojums | *Venetian Blind* XML shēmas dizains nodrošina maksimālu XML shēmas tipa atkārtotu izmantojamību. Komponentes ir viegli saprotamas un grupē vairākus saistītus datus.  Jāizvairās no *Russian Doll* plašas izmantošanas, jo sistēmas, kas XML shēmas lieto, lai veidotu klašu definīcijas, visus anonīmos tipus automātiski identificē kā Tips1, *Enum2* utt., kas ļoti apgrūtina koda lasāmību un saprašanu. Arī tad, ja XML shēmu papildina ar jaunu tādu tipu, nav garantijas, ka Tips1 paliks un netiks aizvietots pavisam ar citu tipu.  Jāizvairās no *Salami Slice*, kur svarīgi atdalīt publicējamus un nepublicējamus tipus (skat. 4.3.7.sadaļu). |
| Piemēri | XML dokumenta fragments:  <Book>  <Title>Illusions</Title>  <Author>Richard Bach</Author>  </Book>  Attiecīgais Russian Doll XML shēmas fragments:  <xsd:element name="Book">  <xsd:complexType>  <xsd:sequence>  <xsd:element name="Title" type="xsd:string"/>  <xsd:element name="Author" type="xsd:string"/>  </xsd:sequence>  </xsd:complexType>  </element>  Attiecīgais Salami Slice XML shēmas fragments:  <xsd:element name="Title" type="xsd:string"/>  <xsd:element name="Author" type="xsd:string"/>  <xsd:element name="Book">  <xsd:complexType>  <xsd:sequence>  <xsd:element ref="Title"/>  <xsd:element ref="Author"/>  </xsd:sequence>  </xsd:complexType>  </xsd:element>  Venetian Blind XML shēmas fragments:  <xsd:simpleType name="TitleType">  <xsd:restriction base="xsd:string">  <xsd:enumeration value="mr"/>  <xsd:enumeration value="mrs"/>  <xsd:enumeration value="dr"/>  </xsd:restriction>  </xsd:simpleType>  <xsd:simpleType name="NameType">  <xsd:restriction base="xsd:string">  <xsd:minLength value="1"/>  </xsd:restriction>  </xsd:simpleType>  <xsd:complexType name="PublicationStructure">  <xsd:sequence>  <xsd:element name="Title" type="TitleType"/>  <xsd:element name="Author" type="NameType"/>  </xsd:sequence>  </xsd:complexType>  <xsd:element name="Book" type="PublicationStructure"/> |

### Import un include lietošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-05 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Atsaucoties uz XML shēmas hierarhijas citu failu, jālieto include, kas parasti ir faila nosaukums – relatīvā atsauce, jo saskaņā ar hierarhijas definīciju – to veidojošie faili tiek saglabāti vienā direktorijā.  Attiecīgi importējot citu hierarhiju, jālieto import izteiksme, norādot absolūtās references. Par citiem ierobežojumiem skatīt 4.3.16.sadaļu. |
| Paskaidrojums | Shēmas dokumentus ir iespējams pārvietot no vienas vietas uz citu, piemēram, hierarhijas jaunas versijas izveidē. Šajos gadījumos jānodrošina atsauces uz citiem shēmas dokumentiem. Ja dokumenti ir pārvietoti kā grupas, tad, izmantojot relatīvās atsauces starp tiem, tiks nodrošināta korekta saistība. Shēmas dokumentiem, kuri būs ārpus grupas (citu hierarhiju lietošana), jāizmanto absolūtās atsauces, lai nodrošinātu saistību.  Hierarhijas uzturēšanas kārtība VISS XML shēmu katalogā aprakstīta 3.nodaļā. |
| Piemēri | <?xml version="1.0" encoding="utf-8"?>  <xs:schema xmlns= "http://ivis.eps.gov.lv/XMLSchemas/100005/WorldWeather/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ww= "http://ivis.eps.gov.lv/XMLSchemas/100005/WorldWeather/v1-0" xmlns:iinfr= "http://ivis.eps.gov.lv/XMLSchemas/100000/IVISInfrastructure/v1-0" targetNamespace= "http://ivis.eps.gov.lv/XMLSchemas/100005/WorldWeather/v1-0" elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0" id="WeatherParameters.xsd">  <xs:include schemaLocation="WeatherDescription.xsd" />  <xs:import namespace= "http://ivis.eps.gov.lv/XMLSchemas/100000/IVISInfrastructure/v1-0" schemaLocation=" http://ivis.eps.gov.lv /XMLSchemas/100000/IVISInfrastructure/v1-0/Notification.xsd" /> |

### Arhitektūras XML shēmu lietošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | AR-06 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Jālieto XML shēmu katalogā publicētās arhitektūras XML shēmas, kas nodrošina pamatu semantikas ziņojumu savietojamībai. |
| Paskaidrojums | * Ar VISS saistīto arhitektūras XML shēmu kopums:  |  |  | | --- | --- | | Identifikācija | <http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0/Identification.xsd> | | e-pakalpojumi | <http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0/IVIS.xsd> | | metadatu apraksta standarts | <http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0/Metadata.xsd> |  * Personas dati:  |  |  | | --- | --- | | Personas kods | <http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/PersonCode.xsd> | | Vārds, uzvārds | <http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/FullName.xsd> | | Cita ar personu saistīta informācija | <http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/Person.xsd> |  * Adrese un kontakta informācija  |  |  | | --- | --- | | Latvijas adrese | <http://ivis.eps.gov.lv/XMLSchemas/100001/Address/v1-1/LivingPlaceLVA.xsd> | | E-pasts, telefons | <http://ivis.eps.gov.lv/XMLSchemas/100001/ContactInfo/v1-0/ContactInfo.xsd> | | Adrešu reģistrs | <http://ivis.eps.gov.lv/XMLSchemas/100007/AddressRegistry/v1-0/ARCode.xsd>  <http://ivis.eps.gov.lv/XMLSchemas/100007/AddressRegistry/v1-2/AddressSearch.xsd> |  * Valsts iestāde, uzņēmums  |  |  | | --- | --- | | Valsts iestāde | <http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0/Authority.xsd> | | Uzņēmums | <http://ivis.eps.gov.lv/XMLSchemas/100679/URTypes/v1-0/URSimpleCommonTypes.xsd>  <http://ivis.eps.gov.lv/XMLSchemas/100679/URTypes/v1-0/URCommonTypes.xsd> |  * Transportlīdzeklis utt  |  |  | | --- | --- | | Transportlīdzekļa apraksts | <http://ivis.eps.gov.lv/XMLSchemas/100025/CSDD/v1-0/CSDD.xsd> | | Traktortehnika | <http://ivis.eps.gov.lv/XMLSchemas/100745/VTUA/v1-0/VTUAData.xsd> | | Cita ar personu saistīta informācija | <http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/Person.xsd> |  * Maksājumi, bankas  |  |  | | --- | --- | | FiDAViSta | <http://ivis.eps.gov.lv/XMLSchemas/100017/fidavista/v1-2/fidavista.xsd> | |  |  | |

## Nosaukumu konvencija

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-01 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Komplekso (xs:complexType) datu tipu nosaukumiem jābeidzas ar teksta rindu Structure. Vienkāršu (xs:simpleType) datu tipu nosaukumiem jābeidzas ar teksta rindu Type. Šī ieteikuma dēļ, kur vien iespējams, jāizvairās no tādiem nobeigumiem elementu nosaukumos. |
| Paskaidrojums | Tas nodrošina iespēju saskaņot visus nosaukumus, jo pieļauj vienkāršas atšķirības starp vienkāršo datu tipu nosaukumiem, komplekso datu tipu nosaukumiem un elementu nosaukumiem. |
| Piemēri | [<xs:simpleType name="PersonCodeType">](http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/PersonCode.xsd)  …  </xs:annotation>  [<xs:simpleType name="ForeignPersonCodeType">](http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0/PersonCode.xsd)  …  </xs:annotation>  <xs:complexType name="PagedSearchInfoStructure" abstract="true">  …  </xs:complexType>  <xs:complexType name="DatePagedSearchInfoStructure" abstract="true">  …  </xs:complexType> |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-02 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Nav ieteicams lietot abreviatūras. No pārāk gariem nosaukumiem jāizvairās, izmantojot īsus un informatīvus nosaukumus. Izmantot iespējams vienīgi labi zināmas abreviatūras, kurās iekļauti tikai sākumburti. Tomēr, vienai sabiedrībai labi pazīstamas abreviatūras var būt nezināmas citām sabiedrībām, kurām jālieto tāds pats ziņojums. Tādās situācijās abreviatūras jāatšifrē elementa dokumentācijā. |
| Paskaidrojums | Ir iecerēts nosaukumus veidot tā, lai tie būtu saprotami visām ieinteresētajām pusēm, jo tas palīdzētu labāk saprast shēmu. |
| Piemēri | <xs:simpleType name="URNType"  <xs:annotation>  <xs:documentation xml:lang="lv">IVIS URN identifikators</xs:documentation>  </xs:annotation>  <xs:restriction base="xs:string">  <xs:pattern value="URN:IVIS:[0-9]{6}:[A-Za-z0-9\_()+,-.=@;$!\*]{0,133}"/>  </xs:restriction>  </xs:simpleType>  <xs:simpleType name="UUIDType">  <xs:annotation>  <xs:documentation xml:lang="lv">Universālais unikālais identifikators</xs:documentation>  </xs:annotation>  <xs:restriction base="xs:string">  <xs:pattern value="[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}"/>  </xs:restriction>  </xs:simpleType> |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-03 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Ikvienam nosaukumam (izņemot NK-04 un 4.3.1.sadaļā aprakstītajos gadījumos, t. i., uzskaitījuma vērtībās un vārdtelpas prefiksos) jāsākas ar lielo sākumburtu un ikvienam vārdam nosaukumā jāsākas ar lielo burtu (*PascalCase*). Ja nosaukumā ir abreviatūra, kurā visi ir lielie burti (piemēram, PMLP) vai cipars, nākamajam vārdam arī jāsākas ar lielo burtu. |
| Paskaidrojums | Tā ir viena no daudzām iespējamām nosaukumu veidošanas metodēm, bet tās pieņemšana nodrošina nepretrunīgumu. Tas palīdz, kad norāda uz nosaukumiem, kad ir zināma burtu pārveidošana no mazajiem par lielajiem („kapitalizācija”), un, līdz ar to, nav nepieciešams par to atgādināt. |
| Piemēri | <xs:complexType name="DatePagedSearchInfoStructure" abstract="true">  <xs:annotation>  <xs:documentation xml:lang="lv">Pēc lappusem datu meklēšanas struktūra ar datumiem</xs:documentation>  </xs:annotation>  <xs:complexContent>  <xs:extension base="PagedSearchInfoStructure">  <xs:sequence>  <xs:element type="xs:dateTime" name="StartTime" minOccurs="0"/>  <xs:element type="xs:dateTime" name="EndTime" minOccurs="0"/>  </xs:sequence>  </xs:extension>  </xs:complexContent>  </xs:complexType> |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-04 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Uzskaitījuma (saraksta) vērtībām visur jālieto mazie burti. Ja vērtība ir īsts nosaukums, abreviatūra vai akronīms, jālieto nepieciešamā „kapitalizācija”. Ja saraksta vērtība sastāv no vairākiem vārdiem, tad pirmais tiek rakstīts ar mazo burtu, bet citi ar lielo (camelCase). |
| Paskaidrojums | Uzskaites vērtību galvenā nozīme ir nepretrunīgums lietošanā. Tāpēc visur jālieto mazie burti, ja vien tie nav nosaukumi, kuri parasti sākas ar lielo burtu. |
| Piemēri | <xs:simpleType name="AnswerTypeType">  <xs:restriction base="xs:string">  <xs:enumeration value="email"/>  <xs:enumeration value="letter"/>  </xs:restriction>  </xs:simpleType> |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-05 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Visiem XML shēmas elementiem jābūt nosauktiem angļu valodā. Pastāv iespēja lietot abreviatūras latviešu valodā.  Atsevišķos gadījumos ir pieļaujams shēmu elementus nosaukt latviski (rakstot elementu nosaukumu bez garumzīmēm), bet nav pieļaujams valodu sajaukums vienas XML shēmas vai hierarhijas ietvaros, izņemot citu hierarhiju lietošanas gadījumā. |
| Paskaidrojums | Angļu nosaukumi un abreviatūras ir plaši lietoti un labāk saprotami vairākumam cilvēku. Abreviatūrām latviešu valodā jābūt paskaidrotām (piemēram, anotācijas sekcijā). |
| Piemēri | Elementa nosaukums angļu valodā:  <xsd:scheme . . .>  . . .  <Metadata>  <Audience>visiem</Audience>  <Contributor Href="http://viss.gov.lv" CodeListCodeValue="100001">Valsts informācijas sistēmu savietotājs</Contributor>  </Metadata>  **. . .**  </xsd:scheme> |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-06 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Izlaisti shēmas dokumenti (XSD faili) jāapzīmē ar anglisko nosaukumu, kas definē tā satura būtību vienskaitlī. Veidojot nosaukumu, jālieto esošā nosaukumu konvencija. Atšķirībā no W3C rekomendācijas VISS XML shēmas versija tiek definēta XML shēmas vārdtelpā. |
| Paskaidrojums | Šīs vadlīnijas garantē lietojuma integritāti, ja kāds shēmas dokuments atsaucas uz citu shēmas dokumentu, izmantojot include vai import. Tādā veidā izmantotus shēmas dokumentus nebūs iespējams izmainīt, nemainot shēmas versiju. |
| Piemēri | Personas vārda un uzvārda XML shēmas nosaukums:  FullName.xsd  Personas koda XML shēmas nosaukums:  PersonCode.xsd |

|  |  |
| --- | --- |
| Ieteikuma Nr. | NK-07 |
| Obligātums | Vēlama |
| Ieteikuma būtība | Datu tipu un elementu, kas satur citus elementus (sarakstu), nosaukumiem jābeidzas ar teksta rindu List. |
| Paskaidrojums | Tas nodrošina iespēju pēc nosaukuma saprast, ka elements ir citu elementu masīvs un tam nepieciešama konkrēta apstrāde. |
| Piemēri | <xs:element name="PersonCodeValidityList" type="PersonCodeValidityListStructure"/>  <xs:complexType name="PersonCodeValidityListStructure">  <xs:sequence minOccurs="0" maxOccurs="unbounded">  <xs:element name="PersonCodeValidity" type="pr:PersonCodeValidityStructure" />  </xs:sequence>  </xs:complexType> |

## XML shēmas kodēšana

### Vārdtelpas (namespace)

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-01 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Ieteikumu būtību veido šādas obligātas prasības:   * Shēmas dokumentam jāsatur mērķa (target) vārdtelpa, un pēc noklusējuma (default) vārdtelpai ir jābūt tādai pašai kā mērķa vārdtelpai. Arī shēmas definīcijas elementā jāliek vārdtelpas vēlamais prefikss un vārdtelpai, uz kuru norāda prefikss, ir jāsakrīt ar mērķa un ar noklusēto vārdtelpu. Prefiksam jāsastāv no maziem burtiem (var lietot „\_” vārdu atdalīšanai) un tā garums nevar pārsniegt 10 simbolus. * W3C XML shēmas vārdtelpai jābūt kvalificētai ar prefiksu xsd vai xs (vēlams xs). * Citām vārdtelpām jāizmanto piemērots prefikss, kas ir atrodams to definējošās shēmās.   Vārdtelpas struktūra:  http://ivis.eps.gov.lv/XMLSchemas/<IVIS 6-zīmju iestādes identifikators>/<namespace nosaukums, unikāls IVIS ietvaros>/v<versijas major un minor komponentes atdalītas ar „-”>  Izvēloties XML shēmas (hierarhijas) vārdtelpas nosaukumu un tās prefiksu, jāpārbauda, vai XML shēmu katalogā jau nav reģistrēta kāda cita hierarhija ar tādu pašu nosaukumu. Visi vārdtelpas nosaukumi un prefiksi VISS XML shēmu kataloga ietvaros ir unikāli! |
| Paskaidrojums | Shēmu izstrādāšanas rīki pēc noklusējuma vienmēr lieto xs vai xsd par XML shēmu vārdtelpu prefiksu, tādēļ, tās tiek piedāvātas kā opcijas. |
| Piemēri | <xs:schema  xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0"  elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0" id="Identification.xsd"> |

### Versiju piešķiršana shēmām

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-02 |
| Ieteikuma būtība | Atbilstoši W3C praksei shēmas ir jāveido ar versijas numuru, izmantojot schema elementa version atribūtu. Šī versija tiek pievienota visām komponentēm, kuras ir definētas shēmas dokumentā. |
| Paskaidrojums | Shēmu versiju norāde ir lietderīga un ļauj pasargāt no problēmām, kuras var izraisīt cilvēki, kuri gadījuma pēc strādā ar nepareizu shēmas versiju.  Vairumā gadījumu ir pietiekami versiju apzīmēt, izmantojot version atribūtu. Kaut gan ir gadījumi, kad gatavam XML dokumentam ir grūti noteikt, kuras versijas shēmas dokumentu izmantot. Tādējādi, piešķirot versiju vārdtelpām, tiek nodrošināts, ka specifiskās komponentes versija ir nepārprotama. |
| Piemēri | <?xml version="1.0" encoding="UTF-8"?>  <xs:schema  xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0"  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" elementFormDefault="qualified" attributeFormDefault="unqualified"  version="1.0" id="Identification.xsd">  ...  </xs:schema> |

### id atribūts schema elementā

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-03 |
| Ieteikuma būtība | schema elementa id atribūts jālieto, lai definētu shēmas identitāti, un tajā jāraksta XML shēmas faila nosaukums. |
| Paskaidrojums | Shēmas identifikators (faila nosaukums) tiek lietots, lai automātiskas sistēmas varētu noteikt shēmas atrašanos, lietojot shēmas saturu: vārdtelpa ir fiziska XML shēmas adrese, bet id atribūts ir shēmas faila nosaukums. |
| Piemēri | Skat. piemēru 4.3.2.sadaļā |

### id atribūta izmantošana shēmas elementā

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-04 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Shēmas elementa identificēšanai pieļaujams izmantot īpašu ID atribūtu. |
| Paskaidrojums | ID atribūtam nozīme ir tikai konkrētās shēmas kontekstā, un tas dod iespēju ātrāk atrast shēmas elementu. |
| Piemēri | <xs:element name="ArrestDate" type="xs:date" id="F1\_7"> |

### elementFormDefault un attributeFormDefault lietošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-05 |
| Obligātums | Obligāta |
| Ieteikuma būtība | elementFormDefault jābūt qualified un attributeFormDefault jābūt unqualified.  Izņēmums: ja definē atribūtus, kuri tiks pievienoti citu vārdtelpu elementiem. Labs piemērs tam ir XLink – saistītā informācija tiek nodrošināta XLink vārdtelpas atribūtos, kuri tiek pievienoti attiecīga dokumenta vārdtelpas elementiem. |
| Paskaidrojums | Tas nodrošina, ka izstrādātājs, nolasot vai atkārtoti lietojot shēmu, var paļauties uz redzamiem prefiksiem un vārdtelpām, nevis izsekot detalizētu shēmas iekšējo struktūru. |
| Piemēri | Atribūti netiks pievienoti citas vārdtelpas elementiem, un tāpēc tiem jābūt kvalificētiem, ja ir šāds gadījums:  <xs:schema  xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0"  elementFormDefault="qualified"  attributeFormDefault="unqualified"  version="1.0"  id="Identification.xsd">  Šādā gadījumā atribūti varētu būt pievienoti citas vārdtelpas elementiem:  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema-instance"  xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0"  elementFormDefault="qualified"  attributeFormDefault="qualified"  ... |

### Datu tipi .v. elementu deklarācija

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-06 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Vairumā gadījumu pastāv izvēles iespēja - atkārtoti lietoto komponenti definēt kā datu tipu, vai kā elementu (skat. arī 4.1.4.sadaļu). Komponentei jābūt definētai kā datu tipam, ja:   * tā tiks lietota ar dažādiem elementu nosaukumiem dažādos kontekstos vai * ir sagaidāms, ka nākamie datu tipi tiks atvasināti no tās.   Savukārt komponente kā elements jādefinē, ja:   * nav nolūka atvasināt jaunas komponentes no tās un * elements tiks lietots ar savu nosaukumu nemainīgi. |
| Paskaidrojums | Ir daudz gadījumu, kad elementam jābūt lietotam ar nemainītu nosaukumu. Piemēram, ja personas kodam vienmēr ir nosaukums PersonCode, tad tā semantika vienmēr būs zināma, un divas sistēmas, izmantojot vienu un to pašu elementu, zinās, ka jālieto tā pati definīcija.  Tomēr ir gadījumi, kad elementam piešķirt nosaukumu arhitektūras shēmas izstrādes laikā ir nelietderīgi. Piemēram, adresei var būt dažādas nozīmes, un tāpēc tā tiks lietota ar dažādiem nosaukumiem, tādiem kā CorrespondenceAddress, HomeAddress, BusinessAddress, ElectoralAddress utt.  Cits gadījums, kad jāizvēlas starp elementu un datu tipu, lai definētu komponenti, ja pastāv nolūks atvasināt no tās citas komponentes. Šajā gadījumā, izmantojot vienīgi datu tipus, iespējams atvieglot shēmas izprašanu, izmantojot vienkāršu mantošanas mehānismu, un izvairoties no xs:redefine lietošanas šim nolūkam.  Dažos gadījumos shēmā ir piemēroti definēt gan datu tipu, gan elementu (ja tas nav pretrunā ar 4.1.4.sadaļu). Tad elements ar zināmo fiksēto semantiku ir pieejams atkārtotai lietošanai, un datu tips ir pieejams atbilstošai modifikācijai. Tomēr jāņem vērā, ka jebkuru globāli definētu elementu var izmantot par cita dokumenta elementu. |
| Piemēri | Elementa, kas vienmēr tiks lietots ar vienu un to pašu nosaukumu un no kura nebūs atvasināti citi elementi, deklarācija:  <xs:element name="WeatherParameters">  <xs:complexType>  <xs:sequence>  <xs:element name="Country" type="xs:string" />  <xs:element name="City" type="xs:string" />  <xs:element name="CityCode" type="xs:string" />  <xs:element name="PartnerID" type="xs:string" />  <xs:element name="LicenseKey" type="xs:string" />  <xs:element name="DaysForward" type="xs:byte" />  <xs:element ref="ww:weather" />  <xs:element name="Notification" type="iinfr:NotificationDescriptionStructure" />  </xs:sequence>  </xs:complexType>  </xs:element>  Tipa, kas tiks lietots ar dažādiem nosaukumiem, deklarācija:  <xs:element name="PersonCode" type="PersonCodeType" />  <xs:simpleType name="PersonCodeType">  <xs:annotation>  <xs:documentation xml:lang="lv">Tips personas kods</xs:documentation>  </xs:annotation>  <xs:restriction base="xs:string">  <xs:pattern value="[0-9]{11}" />  </xs:restriction>  </xs:simpleType> |

### Globālās definīcijas

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-07 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Shēmas dokumentiem jānodrošina, lai globāli pieejamas būtu komponentes definīcijas, kuras:   * tiek atkārtoti lietotas shēmas iekšienē; * ir pieejamas atkārtotai lietošanai citās shēmās vai * ir paredzēts izmantot par cita XML dokumenta elementu.   Parasti šādi tipi tiek izpausti ar elementu palīdzību. |
| Paskaidrojums | Galvenais šādas pieejas iemesls ir ierobežot izmaiņas efektu. Ja tiek atstātas lokālas komponenšu definīcijas, ir viegli kontrolēt, kas vēl izmanto šīs definīcijas, un, tādējādi, ierobežot izmaiņu ietekmi. |
| Piemēri | – |

### Elementi .v. Atribūti

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-08 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Shēmas jāprojektē tā, lai elementi būtu galvenie informācijas satura glabātāji XML instancēs. Atribūti ir vairāk piemēroti, lai glabātu speciālus metadatus – vienkāršas vienības, kuras nodrošina vairāk informācijas par elementa saturu.  Atribūtus nevar izmantot, lai kvalificētu citus atribūtus, kur tas var radīt neskaidrības. |
| Paskaidrojums | Atribūti, atšķirībā no elementiem, nevar glabāt strukturētus datus. Sakarā ar to elementi tiek izvēlēti par informācijas satura pamata glabātājiem. Tomēr, pieļaujot atribūtu izmantošanu elementa satura metadatu glabāšanai (piemēram, datuma formāts), instances dokumentu varētu padarīt vienkāršāku un vieglāk saprotamu. |
| Piemēri | Dzimšanas datums var būt izteikts XML ziņojumā šādi:  <DateOfBirth>1975-06-03</DateOfBirth>  Tomēr varētu būt nepieciešams vairāk informācijas, piemēram, kā šis dzimšanas datums ir apstiprināts. Tas var būt izteikts ar atribūta palīdzību šādā veidā:  <DateOfBirth VerifiedBy="View of Birth Certificate">  1975-06-03  </DateOfBirth>  Zemāk esošais pieraksts nav piemērots:  <DateOfBirth  VerifiedBy="View of Birth Certificate"  ValueSet="ISO 8601"  Code="2">1975-06-03</DateOfBirth>  Nav pilnīgas skaidrības, vai Code kvalificē VerifiedBy atribūtu, vai ValueSet atribūtu. Piemērotāks atveidojums varētu būt šāds:  <DateOfBirth>  <VerifiedBy Code="2">View of Birth Certificate</VerifiedBy>  <Value ValueSet="ISO 8601">1975-06-03</Value>  </DateOfBirth> |

### Shēmu dokumentēšana

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-09 |
| Obligātums | Obligāta |
| Paskaidrojums | Elementa annotation apakšelements documentation eksistē, lai palīdzētu dokumentēt shēmas. Priekšrocība, izmantojot šo elementu, nevis iekļaujot tekstu komentāros, ir tā, ka elements, būdams daļa no XML dokumenta satura, atvieglo iespēju apstrādāt tekstu ar stila lapu (*stylesheet*), piemēram, lai sagatavotu lietotāja dokumentāciju. Komentāru informāciju nevajag pārnest no XML procesora uz lietojumu, tādu kā, XSLT procesors, jo tā var tikt pazaudēta.  Informācija par shēmas elementiem latviešu valodā tiek iekļauta documentation elementā ar xml:lang=”lv” atribūtu.  Shēmas elementu dokumentāciju, jāieliek uzreiz pēc elementa deklarācijas. Ja elementa dokumentācija satur vienu teikumu, tad „.” galā nevajag, ja vairākus, tad vajag. |
| Piemēri | Jāizmanto documentation tags dokumentācijas elementā:  <xs:simpleType name="PersonCodeType">  <xs:annotation>  <xs:documentation xml:lang="lv">Tips personas kods</xs:documentation>  </xs:annotation>  <xs:restriction base="xs:string">  <xs:pattern value="[0-9]{11}" />  </xs:restriction>  </xs:simpleType> |

### Shēmu atkārtotās lietošanas īpatnības

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-10 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Jāizvairās no xs:redefine lietošanas. |
| Paskaidrojums | Tas ir nepieciešams, lai paaugstinātu XML shēmas skaidrību un lasāmību. xs:import nevar būt lietots bez namespace atribūta (skat. arī 4.1.5.sadaļu).  Šī īpatnība pieļauj nekvalificētu referenci uz citām komponentēm bez mērķa vārdtelpas. Tas var novest pie shēmām, kuras ir sarežģīti uzturēt un atjaunot – kurām nav redzamas atkārtoti lietotās saites. |
| Piemēri | – |

### XML shēmu mantošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-11 |
| Obligātums | Vēlams |
| Ieteikuma būtība | Ja pastāvošā definīcija neatbilst konkrētām prasībām, tad var izmantot XML shēmas mantošanas mehānismu, lai definētu jaunu datu tipu, kas lielā mērā bāzēts uz kāda jau eksistējoša datu tipa.  Dažos gadījumos datu tips satur visas atļautās vērtības vai definē standartizēto datu formātu (piemēram, pasta indekss). Šādā gadījumā mantošana jālieto vienīgi, lai ierobežotu sākotnējo datu tipu (piemēram, visas pasta nodaļas, kas ir reģistrētas Rīgas rajonā). Citiem vārdiem, jāpārliecinās, ka modificētais datu tips ir pilnīgi savietojams ar bāzes tipu. |
| Paskaidrojums | Mantošanas mehānisms atļauj atvasināt jaunus tipus, kuri lietotājiem būs nepārprotami un dos iespēju identificēt atkarības starp definīcijām.  Ir četri mantošanas tipi, izmantojot extension un restriction. Tie ir:   * vienkāršā datu tipa ierobežošana; * vienkāršā datu tipa paplašināšana (lai izveidotu sarežģītu tipu); * kompleksā datu tipa ierobežošana; * kompleksā datu tipa paplašināšana.   Jāatceras, ja tiek ierobežots kompleksais tips, XML shēmas sintakse divkāršo pamattipa pilno definīciju. Tas var radīt grūtības, lai identificētu atvasinātam tipam nepieciešamās izmaiņas, ja bāzes tips ir mainīts. Šajā nolūkā, kompleksā datu tipa ierobežošanai jāpiemēro stingra pārraudzība. |

### Elementu datu saturs

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-12 |
| Obligātums | Obligāts |
| Ieteikuma būtība | Neobligātie elementi, kuri ir izveidoti, lai ietvertu saturu, nevar būt tukši. Shēmai jānodrošina, lai elementi vai nu nav, vai ir aizpildīti. |
| Paskaidrojums | Neobligāto elementu gadījumā, datu tukšumu varētu izteikt, izlaižot elementu no XML dokumenta. Dažreiz vienkāršāk ir elementu atstāt tukšu, nekā to izlaist, piemēram, ja viena un tā pati procedūra tiek izmantota aizpildītam un tukšam vienumam, kas ir daudz vieglāk no implementācijas viedokļa. Tomēr tas nav primārais mērķis, daudz svarīgāk ir sadarbspējas slāni (t.i., XML dokumentus) saglabāt tīru un kodolīgu. No otras puses, arī klātesošie neobligātie elementi aizņem sistēmas resursus. |

### Alternatīvo nosacījumu reprezentācija

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-13 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Šis ieteikums ir 4.3.12.sadaļas ieteikuma turpinājums. Alternatīvie nosacījumi jāreprezentē, izmantojot elementa vai atribūta vērtības, nevis elementa klātbūtni vai neesamību. |
| Paskaidrojums | XML stila dēļ dažreiz tiek lietota tukša elementa klātbūtne, kas nozīme “jā”, un elementa neesamība, kas nozīme “nē”. Tiek rekomendēts vienmēr prezentēt elementa vai atribūta vērtību (ar vērtībām “yes” vai “no”), lai norādītu nosacījuma klātbūtni vai neesamību. Tas lasītājiem sniedz iespēju vieglāk saprast XML dokumentu. |

### string, normalizedString, token un citi W3C XML shēmas tipi

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-14 |
| Obligātums | Obligāts |
| Ieteikuma būtība | Definējot elementus, datu tipus, jāsaprot, ko nozīmē un ar ko atšķiras konkrētie vienkāršie datu tipi, tai skaitā, xs:string no xs:normalizedString, xs:int no xs:integer utt. |
| Paskaidrojums | Labākais paskaidrojošais un aprakstošais materiāls par XML shēmas tipiem ir atrodams dokumentā [5]. Standarta XML shēmas tipu hierarhijas diagramma ir parādīta 6.attēlā.  type-hierarchy  .attēls. XML shēmas standarta tipu hierarhijas diagramma |

### Saraksti

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-15 |
| Obligātums | Vēlama |
| Ieteikuma būtība | Veidojot saraksta tipus un elementus, sarakstā pieļaujamais elementu skaits tiek definēts xs:sequence sekcijā. |
| Paskaidrojums | Sarakstā pieļaujamo elementu skaitu ir iespējams noteikt ar diviem paņēmieniem, turklāt ģenerētais XML fragments nemainās. Pirmais – sarakstā pieļaujamo elementu skaitu definē xs:sequence sekcijā (skat. 7.attēlu), otrais – sarakstā pieļaujamo elementu skaitu definēt apakšējā elementā (skat. 8.attēlu).  List2  .attēls. Pieļaujamais elementu skaits tiek definēts xs:sequence sekcijā  List1  .attēls. Pieļaujamais elementu skaits tiek definēts elementā |
| Piemēri | Pieļaujams elementu skaits tiek definēts xs:sequence sekcijā:  <xs:complexType name="ItemListStructure">  <xs:sequence maxOccurs="unbounded">  <xs:element name="Item" type="xs:string"/>  </xs:sequence>  </xs:complexType> |

### XML shēmas satura ierobežojumi

|  |  |
| --- | --- |
| Ieteikuma Nr. | SK-16 |
| Obligātums | Vēlama |
| Ieteikuma būtība | Dažādu papildu rīku (Microsoft BizTalk) lietošanu nosaka virkni ierobežojumu XML shēmas izstrādātājiem. Tos ir svarīgi ievērot arī tiem izstrādātājiem, kas nelieto minētos rīkus, jo potenciāli ir liela varbūtība, ka kādā integrācijas procesā vai e-pakalpojumā tiktu izmantota attiecīgā XML shēma (hierarhija). |
| Paskaidrojums | Veidojot XML shēmu, jāpievērš uzmanība šādām prasībām:   * XML shēmas faila nosaukumam jāatšķiras no jebkāda elementa nosaukuma, kas ir ievietots šajā shēmā * Nav ieteicams veikt vairāku shēmu importu (ar import operatoru) no vienas vārdtelpas. Ja iespējams, jāimportē hierarhijas vienu shēmu, kurā ir ietvertas visas citas hierarhijas shēmas ar include lietošanu. |
| Piemēri | Failā PersonCode.xsd nav vēlams veidot elementu PersonCode:  <xs:element name="PersonCode" type="PersonCodeType"/>  Hierarhija, kas apvieno personas tipus, ir definēta XML shēma Person.xsd, kas ietver šādus include operatorus:  <xs:include schemaLocation="DeclaredPersonList.xsd"/>  <xs:include schemaLocation="Citizenship.xsd"/>  <xs:include schemaLocation="PersonLivingPlaceInfo.xsd"/>  <xs:include schemaLocation="LifeEvent.xsd"/>  <xs:include schemaLocation="Nationality.xsd"/> |

## Atribūtu lietošana

### default un fixed atribūtu lietošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | AL-01 |
| Obligātums | Vēlams |
| Ieteikuma būtība | default atribūts netiek izmantots, lai pievienotu svarīgu informāciju shēmas elementam vai atribūtam.  fixed atribūts netiek izmantots, lai pievienotu svarīgu informāciju shēmas elementam vai atribūtam, izņemot atribūtus, ja tie tiek lietoti kopā ar use=”required” atribūtu. |
| Paskaidrojums | Šie divi atribūti ļauj shēmas apstrādes procesā pievienot informāciju XML dokumentam, izmantojot shēmas saturu. Piemēram, tas shēmai ļauj ierakstīt tās versijas numuru vai identifikatoru XML dokumentā, lai identificētu XML shēmu. |

### Lokalā .v. Globalā atribūtu definēšana

|  |  |
| --- | --- |
| Ieteikuma Nr. | AL-02 |
| Obligātums | Obligāts |
| Ieteikuma būtība | Atribūtiem jābūt definētiem lokālā redzesloka ietvaros, ja tas nav arhitektūras atribūts, kas tiks lietots specifiskam mērķim (skat. 4.5.1.sadaļu). |
| Paskaidrojums | Tas vienkāršo shēmas saprašanu, jo ļauj izvairīties no iespējamām strīdīgām vārdtelpām. Ja atribūts ar līdzīgu definīciju tiek lietots dažādās vietās, jādefinē datu tips vai atribūtu grupa, kas jāizmanto atkārtoti. Ja situācija ir vēl sarežģītāka, tad atribūtu datus labāk mēģināt reprezentēt kā elementus. |

## Klasificējamās vērtības un pārskaitāmie tipi

### Klasificējamo vērtību kopas prezentācija

|  |  |
| --- | --- |
| Ieteikuma Nr. | CL-01 |
| Obligātums | Obligāta |
| Ieteikuma būtība | Klasificējamo vērtību kopas reprezentācijai jālieto mehānisms, kuru pieņem UBL un UN/CEFACT un apraksta dokuments [6]. Implementācijas piemērus var atrast pēc adreses <http://www.unece.org/etrades/unedocs/codelist.htm>.  Definējot klasificēto vērtību, jālieto ivis:CodeListDescriptionGroup atribūtu grupa vai atsevišķi atribūti (skat. http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0/Identification.xsd). |
| Paskaidrojums | Vērtību kopas, kas ir starptautiski akceptētas (piemēram, valsts kodi, valūtas kodi, mērvienības) un pārvaldes definētas, bieži tiek lietotas, lai atvieglotu sadarbspēju.  Jāņem vērā, ka UBL vērtību kopu reprezentācijai izmanto fixed atribūta vērtības bez required atribūta.  Dažos gadījumos var būt nepieciešams saglabāt vērtību kopas ar shēmas dokumentu, kur tas ir pielietots, lai pārliecinātos, ka dokuments būs saprotams turpmāk. |
| Piemēri | Instance bez skaidri izteikta atribūta:  <CurrencyCode>EUR</CurrencyCode>  Klasificējamas instances piemērs:  <CurrencyCode ivis:ListAgencyID="6" ivis:ListID="ISO 4217" ivis:ListVersionID="0.3">EUR</CurrencyCode>  vai  <Category ivis:CodeListID="XMLSchemaType" ivis:CodeListAgencyID="100001" ivis:CodeListAgencyName="VRAA" ivis:CodeListLanguageID="lv" ivis:CodeListCodeValue="1">Arhitektūras XML shēma</ivis:Category> |

### Teksts .v. Kods

|  |  |
| --- | --- |
| Ieteikuma Nr. | CL-03 |
| Obligātums | Obligāts |
| Ieteikuma būtība | Kodus nevar lietot tekstā bez paskaidrojuma, kas identificē koda būtību. To iespējams realizēt – vai nu, iekļaujot gan tekstu, gan kodu, vai nodot norādi uz dokumentu, kas satur papildinformāciju. Otrajā gadījumā, papildinformācijai jābūt pieejamai gan lietotājam, gan automatizētajai sistēmai. Skatīt arī šī dokumenta 4.5.1.sadaļu. |
| Paskaidrojums | Katram XML dokumentam ir jāizpilda divi pretrunīgi mērķi - jābūt gan sistēmai, gan cilvēkam lasāmai. Nodrošinot mašīnai lasāmu kodu un cilvēkam lasāmu tekstu (vai referenci uz šo tekstu), iespējams ne tikai pasargāt no interpretācijas kļūdām, bet ļauj arī vienkāršāk demonstrēt dokumentu, izmantojot dokumenta stila lapas. |

## XML shēmas publicēšana

### VISS XML shēmu kataloga izmantošana

|  |  |
| --- | --- |
| Ieteikuma Nr. | SP-01 |
| Obligātums | Vēlams |
| Ieteikuma būtība | VISS XML shēmu katalogs jālieto kā primārā XML shēmu krātuve, kamēr specifiskā shēma tiks lietota tikai konkrētas iestādes ietvaros. Jālieto centralizēti definēti VISS XML shēmu kataloga datu tipi un jau izveidotās XML struktūru hierarhijas. |
| Paskaidrojums | VISS XML shēmu kataloga lietošana nodrošinās sadarbspēju un XML shēmu atkārtotu izmantojamību. |

### XML shēmas metadati

|  |  |
| --- | --- |
| Ieteikuma Nr. | SP-02 |
| Obligātums | Obligāts |
| Ieteikuma būtība | Veidojamajām shēmām jāsatur visi metadatu standartā aprakstītie pieejamie lauki. Metadatu aizpildīšana notiek, izmantojot XML shēmas pievienošanas ievadformu VISS XML shēmu katalogā (skat. 3.4.sadaļu), tie nav jāaizpilda manuāli! |
| Paskaidrojums | Gan shēmām, gan individuālām deklarācijām shēmas iekšienē ir vajadzīgi metadati, lai tās varētu indeksēt un meklēt. Šīs iespējas nodrošina VISS „Metadatu un e-pakalpojumu identifikācijas standarts” [1]. |
| Piemēri | <xs:schema xmlns="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:ivis="http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" targetNamespace= "http://ivis.eps.gov.lv/XMLSchemas/100001/IVIS/v1-0" elementFormDefault="qualified" attributeFormDefault="unqualified" version="1.0" id="v1-0-Identification">  <xs:annotation>  <xs:appinfo>  <ivis:Metadata xmlns:pers="http://ivis.eps.gov.lv/XMLSchemas/100001/Person/v1-0">  <ivis:Audience>visiem</ivis:Audience>  <ivis:Contributor Href="http://www.abcsoftware.lv" CodeListCodeValue="100001">SIA "ABC Software"</ivis:Contributor>  <ivis:Creator Href="http://www.eps.gov.lv" CodeListID="Authority" CodeListAgencyName="ĪUMEPLS" CodeListURI="http://ivis.eps.gov.lv/" CodeListLanguageID="lv" CodeListCodeValue="100001">ĪUMEPLS</ivis:Creator>  <ivis:Date>  <ivis:Created>2006-12-05</ivis:Created>  <ivis:Modified>2006-12-05</ivis:Modified>  </ivis:Date>  <ivis:Description>  <ivis:Default>Elementu un atributu identifikācijas tipi</ivis:Default>  </ivis:Description>  <ivis:Format>  <ivis:Default>text/xml</ivis:Default>  </ivis:Format>  <ivis:Identifier Scheme="URN">URN:IVIS:100001:XSD-IVIS-Identification-v1-0</ivis:Identifier>  <ivis:Language>LV</ivis:Language>  <ivis:Publisher CodeListID="Authority" CodeListAgencyID="100001" CodeListAgencyName="ĪUMEPLS" CodeListURI="http://ivis.eps.gov.lv/" CodeListCodeValue="100001">ĪUMEPLS</ivis:Publisher>  <ivis:Relation>  <ivis:ConformsTo Href="http://www.ietf.org/rfc/rfc4617.txt?number=4617">IVIS:URN saskaņā ar RFC 4617</ivis:ConformsTo>  <ivis:ConformsTo>UUID saskaņā ar RFC 4122, ITU-T Rec. X.667 un ISO/IEC 11578:1996</ivis:ConformsTo>  <ivis:ConformsTo>XML shemu izstrades vadlinijas</ivis:ConformsTo>  <ivis:ConformsTo Href="http://www.iana.org/assignments/urn-namespaces">IVIS:URN according to Official IANA Registry of URN Namespaces</ivis:ConformsTo>  </ivis:Relation>  <ivis:Status>  <ivis:Default>EDITED</ivis:Default>  <ivis:Version>v1.0</ivis:Version>  </ivis:Status>  <ivis:Subject>  <ivis:Keyword>IVIS</ivis:Keyword>  <ivis:Keyword>XML shēma</ivis:Keyword>  <ivis:Keyword>identifikācija</ivis:Keyword>  <ivis:Keyword>URN</ivis:Keyword>  <ivis:Project>IVIS</ivis:Project>  </ivis:Subject>  <ivis:Title>  <ivis:Default>Elementu un atributu identifikācijas tipi</ivis:Default>  </ivis:Title>  </ivis:Metadata>  </xs:appinfo>  </xs:annotation>  ...  </xs:scheme> |

### XML shēmas metadatu elementi

Šajā sadaļā ir aprakstīti metadatu elementi, kas tiek pielietoti XML shēmas aprakstam. Papildinformācija par visiem minētiem metadatu elementiem ir pieejama dokumentā [1]. Konkrēta elementa apraksts ir dots 3.tabulā pēc šādas shēmas:

* metadatu elementa nosaukums;
* minimālais/ maksimālais reprezentējamo elementu skaits. Minimālai vērtībai parasti atbilst vērtības „0” vai „1”; maksimālai vērtībai parasti atbilst vērtība „1” vai neierobežots daudzums;
* kodēšanas shēma, kura definē elementam pieejamās vērtības;
* metadatu elementu pielietojuma komentāri XML shēmas aprakstam;
* nosacījumi, kuriem pakļaujas elementi šajā standartā (piemēram, formāts).

3.tabula

Pielietotie metadatu elementi XML shēmas aprakstam

| Nosaukums | Min/ Max | Kod. sh. | Komentāri par XML shēmu pielietojumu | Formāts |
| --- | --- | --- | --- | --- |
| Contributor | 0..n |  | Organizācija, kas piedalās XML shēmas pilnveidošanā un uzturēšanā, bet nav XML shēmas izveidotāja | string |
| Creator | 1..1 |  | Organizācija, kas ir XML shēmas izveidotāja (īpašniece) | string |
| Date.Created | 1..1 | W3CDTF | Datums, kad XML shēma ir izveidota jeb lejupielādēta XML shēmu katalogā | date |
| Date.Modified | 1..1 | W3CDTF | XML shēmas pēdējās modifikācijas datums | date |
| Date.Declared | 0..1 | W3CDTF | XML shēmas publikācijas datums | date |
| Description | 0..1 |  | XML shēmas apraksts | string |
| Format | 0..1 | MIME | XML shēmas formāts, parasti „text/xml” | string |
| Identifier | 1..1 | URN:IVIS | Shēma tiks identificēta atbilstoši URN:IVIS shēmai saskaņā ar Metadatu un identifikācijas standartu [1]. Identifikators tiks automātiski ģenerēts XML shēmu katalogā. | string |
| Language | 0..1 | ISO 639-2/T | Valoda, kurā ir uzrakstīti komentāri XML shēmā | string |
| Publisher | 1..1 |  | Organizācija, kas publicēja XML shēmu (parasti – VRAA) | string |
| Relation.ConformsTo | 0..n | URI | Shēmas atbilstība standartam vai vadlīnijai | string |
| Rights.Copyright | 0..n |  | Autortiesības uz XML shēmu. Parasti autortiesības pieder XML shēmas izveidotājam. | string |
| Status | 0..n |  | XML shēmas statuss: CANDIDATE, EDITED, PUBLISHED vai DELETED | string |
| Version | 0..n |  | XML shēmas versija | string |
| Subject.Category | 1..n | Atkarīgs no datu vārdnīcas | XML shēmas kategorijas raksturojums | string |
| Subject.Keyword | 0..n |  | Atslēgvārdi, kas ir saistīti ar XML shēmu | string |
| Subject.Project | 0..1 |  | Parasti XML shēmas hierarhijas nosaukums | string |
| Title | 1..1 |  | XML shēmas nosaukums | string |

# Pielikumi

Pielikumu saraksts:

1. Biznesa procesu modelēšanas valodas notācijas elementu apkopojums un apraksts.

## Pielikums Nr.1

Pielikums satur biznesa procesu modelēšanas valodas notācijas elementu apkopojumu un aprakstu. Biznesa procesu modelēšanai ir izmantota vienotas modelēšanas valodas *(UML- Unified Modelling Language*) stāvokļu diagramma (Statechart diagramm).

| **Elements** | **Nosaukums** | **Definīcija / apraksts** |
| --- | --- | --- |
|  | Sākumstāvoklis | Atspoguļo diagrammas sākumu. |
|  | Stāvoklis | Atspoguļo situācijas vai statusus aprakstāmā objekta (dotajā dokumentā aprakstāmais objekts ir XML shēma) dzīves gaitā. |
|  | Izvēle | Atspoguļo nosacījumu dinamiskam atzarojumam. Tas ļauj sadalīt pārejas dažādos izejas virzienos, atkarībā no izpildīta nosacījuma rezultāta. |
|  | Pāreja | Atspoguļo viena stāvokļa pāreju uz citu stāvokli, izpildoties konkrētam notikumam vai darbībai. |
|  | Sadalīšana | Atspoguļo pārejas sadalīšanu vairākās paralēlās pārejās. |
|  | Apvienošana | Atspoguļo vairākās paralēlas pārejas apvienošanu vienā. |
|  | Piezīmes | Atspoguļo piezīmes bloku, kuru var piesaistīt pie jebkura diagrammas elementa. Dotajā dokumentā ar piezīmēm tiek apzīmēti izvēles elementa nosacījumi. |
|  | Beigustāvoklis | Atspoguļo diagrammas beigas. |