



X-tee teenuste kirjeldamise parimad praktikad ja tüüpilised vead

Sisukord

1	Sissejuhatus.....	2
2	Andmekogu WSDL.....	2
2.1	X-tee nõuded.....	3
2.1.1	Nimeruumid.....	3
2.1.2	WSDLi stiil.....	3
2.1.3	Vastused sisaldavad päringut.....	3
2.1.4	Ühes andmekogus võivad olla vaid ühe stiiliga X-tee teenused.....	3
2.1.5	X-tee päised.....	4
2.1.6	Manused.....	4
2.1.7	Üheselt mõistetav struktuur.....	4
2.2	Valideerimine ja korrektne kood.....	4
2.3	Operatsioonide versioonid.....	4
2.4	WS-I.....	4
2.5	Skeemide kasutamine.....	5
2.6	Inimloetavus.....	5
2.6.1	Kommentaarisid.....	5
2.6.2	Treppimine.....	6
2.6.3	Elementide nimetused.....	6
2.6.4	Skeemide importimine.....	6
2.6.5	Nimeruumid.....	6
2.7	Kodeeringud.....	7
2.8	WSDLi dokumenteerimine.....	7
3	Teenustega liidestamise dokumentatsioon.....	8
4	Tüüpilised vead.....	8
4.1	WSDLi stiil.....	8
4.1.1	„Rpc/encoded” ja „document/literal wrapped” teenused ühes andmekogus.....	8
4.1.2	Vale stiili kasutamine.....	8
4.1.3	Atribuut „encodingStyle”.....	8
4.2	Vastused ei sisalda päringut.....	9
4.3	X-tee päised.....	9
4.3.1	Atribuut „part”.....	9
4.3.2	Päised puuduvad.....	10
4.3.3	Vigane päiste struktuur.....	10
4.4	Manused.....	11
4.5	Operatsioonide nimekiri.....	11
4.6	Skeemide kasutamine.....	11
4.6.1	Mitmekordselt defineeritud elemendid.....	11
4.6.2	Väliseid elemente on kasutatud ilma importimiseta.....	11
4.6.3	Kättesaamatud välised skeemid.....	11
4.6.4	Mitu skeemi WSDLi dokumendi sees.....	12
4.7	Kodeeringud.....	12

1 Sissejuhatus

Riigi infosüsteemide andmevahetuskiht X-tee on tehniline ja organisatsiooniline keskkond, mis võimaldab korraldada turvalist internetipõhist andmevahetust riigi infosüsteemide vahel. X-tee võimaldab asutustel/inimestel turvaliselt andmeid vahetada, samuti korraldada isikute juurdepääsu riigi andmekogudes säilitatavatele ja töödeldavatele andmetele.

Andmekogud pakuvad andmeid X-tee teenuste abil. Korrektselt kirjeldatud ja dokumenteeritud teenused parandavad andmeid vahetatavate infosüsteemide koosvõimet ning muudavad odavamaks liidestamise hinna.

Selles dokumendis on X-tee teenuste kirjeldamise ning dokumenteerimise parimad praktikad ning tüüpilised kirjeldamise vead.

Tüüpiliste vigade nimekiri põhineb RIHAsse sisestatud ning kasutusel olevate andmekogude kirjeldustel.

See dokument on mõeldud arendajatele, kes projekteerivad X-tee andmekogusid, valmistavad teenuste kirjelduste WSDL dokumente ja dokumenteerivad teenuseid.

2 Andmekogu WSDL

Iga X-tee andmekogu jaoks peab olema loodud X-tee teenuste masinloetav kirjeldus ehk andmekogu WSDL dokument. Seda võivad kasutada nii inimesed kui ka automatiseeritud protsessid ja tööriistad. Seetõttu peab WSDL dokument vastama kõigile standarditele, lisaks peab kood olema loetav ning piisaval määral dokumenteeritud.

WSDLi koostamisel tuleks eelkõige lähtuda ametlikest juhenditest. Alatest viiendast X-tee versioonist kehtib arendajatele mõeldud juhend „Nõuded infosüsteemidele ja adapterserveritele (doc/literal)“:

http://x-road.ee/docs/est/nouuded_infosusteemidele_ja_adapterserveritele.pdf

Selles versioonis kasutatakse „document/literal wrapped“ stiili, mis omakorda võimaldab tagada parema koosvõime võrreldes varem kasutatud stiiliga.

Varem loodud teenuseid kirjeldab juhend „Nõuded infosüsteemidele ja adapterserveritele (rpc/encoded)“:

http://x-road.ee/docs/est/nouuded_infosusteemidele_ja_adapterserveritele_legacy.pdf

Selles kasutatakse „rpc/encoded“ stiili.

Kuigi praegu toetavad kõik turvaserverid veel mõlema stiili kasutamist, peab arvestama, et „rpc/encoded“ stiili tugi kaob tulevikus. Seetõttu peab uute teenuste loomisel kasutama alati uut stiili. Vanu teenuseid täiendades tuleb kaaluda üleminekut uuele stiilile.

2.1 X-tee nõuded

X-tee teenuste kirjeldamiseks on kasutatud WSDL dokumente, millele on rakendatud teatud kitsendusi ning kirjeldustes peavad olema X-tee teenuste jaoks kohustuslikud elemendid.

Täielik nõuete nimekiri on vastavas X-tee juhendis. Järgnevalt on välja toodud ainult need punktid, millega on kõige rohkem probleeme.

2.1.1 Nimeruumid

X-tee nõuete kohaselt peavad andmekogude ning metateenuste nimeruumid olema kindla struktuuriga.

„Document/literal wrapped” stiiliga andmekogu nimeruum on <http://name.x-road.ee/producer/>, kus *name* on andmekogu nimi. X-tee metateenuste nimeruum on sel juhul <http://x-road.ee/xsd/x-road.xsd>.

„Rpc/encoded” stiili puhul on nimeruum <http://producers.name.xtee.riik.ee/producer/name>, kus *name* on andmekogu nimi. X-tee nimeruum on sel juhul: <http://x-tee.riik.ee/xsd/xtee.xsd>.

Teatud „document/literal wrapped” andmekogude puhul on kasutatud andmekogude nimeruumi kujul <http://name.ee.x-rd.net/producer/>, kus *name* on andmekogu nimi. X-tee nimeruum on siis <http://x-rd.net/xsd/xroad.xsd>. Selliste nimeruumidega andmekogud tekkisid siis, kui loodi algne spetsifikatsioon „document/literal wrapped” stiiliga X-tee teenuste jaoks. Mõne aja pärast spetsifikatsiooni muudeti ning praegu ei ole selliste nimeruumide kasutamine soovitatav. Turvaserverid küll toetavad veel neid nimeruume, kuid see tugi võib tulevikus kaduda.

Uute teenuste loomisel tuleks alati valida „document/literal wrapped” stiil ning kehtiv X-tee nimeruum <http://x-road.ee/xsd/x-road.xsd>.

2.1.2 WSDLi stiil

X-tee reeglite kohaselt võivad teenuste kirjeldused olla kas „document/literal wrapped” stiilis (uus stiil) või „rpc/encoded” stiilis (vana stiil). Siin tuleks pöörata tähelepanu asjaolule, et stiili valik ei ole ainuüksi WSDLi <binding> osas kasutatud atribuut, vaid see mõjutab ka teenuste kirjelduste struktuuri ning XML sõnumite struktuuri. Lisaks on stiili valik alati seotud kindla X-tee nimeruumiga ning tuleb jälgida, et need oleksid kooskõlas.

2.1.3 Vastused sisaldavad päringut

Vastused X-tee päringutele peavad alati sisaldama ka algset päringut. Vastuses toodud päringu koopia on üldjuhul muutmata kujul, kuid kui teenuse sisend sisaldab MIME-manust, siis väljundis olevas teenuse osas esitatakse manusena edastatud sisendparameetri väärtusena manuses olnud sisendparameetri SHA-512 räsi.

2.1.4 Ühes andmekogus võivad olla vaid ühe stiiliga X-tee teenused

Ühes WSDL dokumendis on võimalik korrektselt kirjeldada vaid ühe stiiliga teenuseid. Seega kui mõni asutus soovib pakkuda teenuseid nii vana kui ka uue stiiliga, siis tuleb registreerida

vähemalt kaks andmekogu.

2.1.5 X-tee päised

X-tee reeglite kohaselt peab alati kirjeldama ja edastama sõnumites kohustuslikud spetsiifilised X-tee päised. Seejuures tuleb jälgida, et need päised oleksid õige struktuuriga ja õiges nimeruumis. Täieliku informatsiooni X-tee päiste kohta leiab X-tee juhenditest.

2.1.6 Manused

X-tee juhendites on näited, kuidas peab igas X-tee versioonis manuseid saatma. Teiste meetodite kasutamine toob üldjuhul kaasa ühilduvusprobleemid ning seda tuleks vältida. Lisaks tuleks meeles pidada, et manuste kasutamist tuleb alati korrektselt kirjeldada ka WSDLis.

2.1.7 Üheselt mõistetav struktuur

X-tee päringute struktuur peaks olema lihtne ja üheselt mõistetav. Elemendi <any> või tüübi „anyType” kasutamisel ei ole võimalik üheselt aru saada, millist päringut teenus ootab ning millise vastuse võib sellele anda. Kui teenuse päringute või vastuste sisu ei ole võimalik rangelt kirjeldada, siis peaks teenus kasutama selliste andmete edastamiseks pigem manuseid. Rangelt määratud struktuur on vajalik näiteks MISPide jaoks, mis võimaldavad luua WSDL dokumentide põhjal veebiliideseid.

2.2 Valideerimine ja korrektne kood

X-tee teenuste kirjeldamiseks on kasutatud WSDLi standardit, mis omakorda peab olema korrektne XML dokument. Seetõttu peab X-tee teenuseid kirjeldav dokument läbima edukalt nii XMLi kui ka WSDLi valideerimise.

Vastavust WSDL standardile saab kontrollida näiteks tarkvara „Eclipse Java EE IDE for Web Developers” abil, mis sisaldab nii WSDL kui ka WS-I valideerimise tööriista:

<http://eclipse.org/home/index.php>

2.3 Operatsioonide versioonid

X-tee spetsifikatsioon ütleb, et teenuse muutmisel peab tegema uue nimetusega teenuse või muutma teenuse versiooni numbrit. Paraku ei ole WSDL-vahenditega võimalik kirjeldada korrektselt mitut samanimelist operatsiooni ühe WSDLi sees. Seega juhul, kui paralleelselt on kasutatud mitut samanimelist teenust, tuleks vanade teenuste kirjeldused eemaldada või panna kommentaari sisse. Kui on plaanis kasutada pikemat aega teenuse erinevaid versioone, siis üldjuhul on mõistlikum luua teenuse uuem versioon teise nimega.

2.4 WS-I

Veebiteenuste koosvõime parandamiseks on WS-I (Web Services Interoperability) organisatsioon loonud täiendavad reeglid WSDL standardile. Nende reeglite järgimine

lihtsustab liidestamist ning vähendab ühilduvusprobleeme.

Vastavust WS-I reeglitele saab kontrollida näiteks tarkvara „Eclipse Java EE IDE for Web Developers” abil, mis sisaldab nii WSDL kui ka WS-I valideerimise tööriista:

<http://eclipse.org/home/index.php>

2.5 Skeemide kasutamine

WSDL dokumendis on kõik tüübid ja elemendid kirjeldatud skeemides. Skeemid võivad paikneda WSDLi dokumendi sees, aga neid saab ka importida väljaspoolt.

Kui on vaja kasutada skeemide importimist, tuleks pöörata erilist tähelepanu sellele, kus paiknevad imporditud skeemid. On ilmne, et ei tasu viidata kohalikus arvutis või kohalikus võrgus olevale failile, sest välised arendajad ei pääse sellele ligi. Alati on hea, kui imporditud skeemid on lisatud RIHAsse. See vähendab riski, et imporditud skeemid ei ole ühel hetkel enam kättesaadavad.

Skeemide importimiseks ei piisa anuüksi nimeruumi nimetamisest WSDLi <definitions> plokis. Kui skeemi elemendid on kasutatud WSDLi sees, siis peab alati lisama <xsd:import> käsu WSDL dokumendis paikneva skeemi sisse. Näiteks xtee skeemi importimine on vajalik praktiliselt igas X-tee teenuste WSDLis, sest xtee skeemis on kirjeldatud X-tee dokumentatsiooni elemendid ja X-tee päised.

Lisaks tuleks skeemide importimisel pöörata alati tähelepanu sellele, et imporditud skeemides poleks sama nimeruumi elemendid ja tüübid topelt defineeritud. Alati tasub kaaluda, kas importimine on üldse vajalik või saab kõik tüübid ja elemendid panna WSDL faili sisse.

WSDLis paiknevate schemade puhul tuleks alati üritada vältida olukorda, kus sama nimeruumi elemendid on paigutatud mitme skeemi alla. Selline skeemide kasutamine on halvasti spetsifitseeritud WSDLi standardis ning paljudes rakendustes võivad tekkida probleemid.

Idealis peaks WSDL dokumendi sees olema ainult üks skeem, mille nimeruum on teenuse nimeruum. Kui aga tüüpide ja elementide arv on väga suur või on soov taaskasutada skeeme mitme erineva andmekogu sees, siis saab kasutada ka skeemide importimist.

2.6 Inimloetavus

Kõik WSDL dokumendid peavad olema inimloetavad. WSDL dokumendi kood peab seetõttu olema ilusasti struktureeritud ja dokumenteeritud. X-tee nõuded ei sea reegleid WSDLi loetavusele, kuid järgnevate soovitude järgimine on hea tava, mis lihtsustab ja kiirendab teenustega liidestamist.

2.6.1 Kommentaarid

WSDL dokumenti tuleks jätta vaid need kommentaarid, mis võivad olla kasulikud sihtrühmale: arendajatele ja analüütikutele, kes tegelevad teenustega liitumisega. Kindlasti tuleks lõppversioonist kustutada kõik üleliigsed kommentaarid ning välja kommenteeritud

koodijupid. Erandiks võiksid olla vaid teenuste vanade versioonide välja kommenteeritud kirjeldused, mida siia kasutatakse. Ka sel juhul tuleks siiski kaaluda nende kustutamist, sest üldjuhul on alati soovitatav kasutada uute liidestuste puhul teenuse viimast versiooni.

2.6.2 Treppimine

Koodi kirjutades tuleks kasutada treppimist ning vältida ülipikki ridu. Mõned RIHAs registreeritud andmekogud kasutavad niivõrd pikki ridu, et mõned redaktorid ei suuda sellist WSDL faili isegi avada.

2.6.3 Elementide nimetused

Elementide nimetused peaksid olema lihtsad ja arusaadavad. Nimetustes tuleks kasutada ASCII sümboleid ning vältida teiste sümbolite kasutamist. Kui teenus võib olla rahvusvahelise tähendusega, on soovitatav valida ingliskeelsed elementide nimetused.

2.6.4 Skeemide importimine

Välise skeemide kasutamine või mittekasutamine võib oluliselt mõjutada WSDLi loetavust, seetõttu tuleks seda alati väga hoolikalt kaaluda.

Välise skeemide kasutamise eelised:

- teeb WSDL dokumendi kompaktseks;
- võimaldab tüüpide ja elementide taaskasutamist, kui neid kasutavad mitmed andmekogud.

Välise skeemide kasutamise puudused:

- mõned redaktorid ei suuda imporditud skeeme mugavalt näidata;
- imporditud skeemid nõuavad eraldi majutamist veebiserverites ning kui veebiserver ei tööta, ei ole võimalik töötada ka WSDLiga;
- kui imporditud skeemid impordivad omakorda teisi skeeme, siis muutub skeemide hierarhia haldamine väga keeruliseks.

Seega võib välise skeemide importimist soovitada vaid juhul, kui teenuste arv on suur, teenuseid kirjeldav struktuur on väga mahukas või kui on soov teatud tüüpe ja elemente taaskasutada.

Imporditud skeemide kasutamisel on alati soovitatav lisada need skeemid RIHAsse, et maandada skeemide kättesaamatuks muutumise riski.

2.6.5 Nimeruumid

Nimeruumi *aliase* defineerimisel tuleks alati eelistada üldkasutatavaid lühendeid. Näiteks:

```
xmlns:xrd="http://x-road.ee/xsd/x-road.xsd",
```

```
xmlns:xtee="http://x-tee.riik.ee/xsd/xtee.xsd",
```

```
xmlns:xsd="http://www.w3.org/2001/XMLSchema",  
xmlns:xs="http://www.w3.org/2001/XMLSchema" jms.
```

Vältida tuleks ühe nimeruumi kasutamist mitme erineva *aliase* abil. Näiteks:

```
xmlns:xsd1="http://www.w3.org/2001/XMLSchema",  
xmlns:xsd2="http://www.w3.org/2001/XMLSchema",  
xmlns:xsd3="http://www.w3.org/2001/XMLSchema".
```

2.7 Kodeeringud

X-teel liikuvad sõnumid kasutavad UTF-8 kodeeringut. Kui teenus kasutab andmete edastamiseks manuseid, siis need võivad olla ka teises kodeeringus, kuid parema ühilduvuse saavutamiseks tuleks eelistada UTF-8 kodeeringut ka manuste puhul.

Teenuste WSDLi loomisel tuleks alati jälgida, et kõik elemendid, kirjeldused ja dokumentatsioonielemendid oleksid korrektselt UTF-8 kodeeringuga, muidu võivad tekkida probleemid dokumendi lugemisel.

2.8 WSDLi dokumenteerimine

X-tee spetsiifilised dokumentatsioonielemendid on mõeldud eelkõige lõppkasutajate jaoks, et oleks võimalik genereerida teenustele automaatselt veebiliideseid. Lisaks on dokumentatsioonielemendid väga kasulikud arendajatele, kes liidestavad infosüsteeme, ning ka äripoolse inimestele, kes näevad neid kirjeldusi RIHAs.

Täielik nimekiri X-tee spetsiifilistest elementidest ning nende kasutamise näited on X-tee dokumentatsioonis. Järgnevalt on loetletud ainult kõige enam kasutatud elemendid:

„/definitions/service/port/xtee:title” – andmekogu pealkiri,

„/definitions/portType/operation/documentation/xtee:title” – teenuse pealkiri,

„//annotation/appinfo/xtee:title” – parameetri pealkiri. See kirjeldus peaks olema iga sisend- ning väljundparameetri küljes.

Kui neid dokumentatsioonielemente on WSDLi sees korrektselt kasutatud, saab andmekogu jaoks luua lihtsasti MISPi kasutajaliidese. Seda veebiliidest on mugav kasutada teenuste testimiseks ja vajadusel ka selleks, et luua odav veebiliides lõppkasutaja jaoks.

X-tee spetsifikatsioon sisaldab ka arendajale mõeldud dokumentatsioonielemente, kuid arendajale mõeldud märkused saab panna mugavalt tavalise xml-kommentaari sisse. Nii saab anda WSDL dokumendi sees ülevaate teenuse tööst või selgitada paremini teenuse struktuuri.

Teenuste sees kasutatud elementide nimetused peaksid olema võimalikult lihtsad ja arusaadavad. Õeldakse, et hästi kirjutatud programmis dokumenteerib kood ennast ise. Samamoodi võiksid siin anda elementide ja teenuste nimetused hea ülevaate nende tähendusest.

3 Teenustega liidestamise dokumentatsioon

Arendajale mõeldud teenustega liidestamise dokumentatsioon on eraldiseisev dokument, mis selgitab piisava täpsusega teenuste sisemist toimimist, arhitektuuri ning kõikide sisend- ja väljundparameetrite tähendusi.

Minimaalselt on vaja lahti kirjutada kõikide sisendite ning väljundite tähendused. Sealhulgas peavad olema välja toodud kõik teenuste tagastatavad veateated ja väljades kasutatud klassifikaatorid.

Väga hea on see, kui RIHAs on saadaval ka arhitektuuri kirjeldus, protsesside kirjeldus ja andmekoguga seotud andmemudel. Need dokumendid annavad hea ülevaate andmekogu süsteemist ning lihtsustavad protsessidest arusaamist.

Lisaks on kasulik ka semantiline kirjeldus, mille saab vormistada kas põhjaliku OWL-failina või lihtsama dokumendina, kus loetletakse kasutusel olevad terminid ja kirjeldatakse nende semantilist tähendust (terminite sõnastikud).

4 Tüüpilised vead

4.1 WSDLi stiil

4.1.1 „Rpc/encoded” ja „document/literal wrapped” teenused ühes andmekogus

Mõnikord üritatakse panna ühe WSDL dokumendi sisse vana ja uue stiiliga X-tee teenuseid, kuid selle tulemusena tekib vigane teenuste kirjeldus. (Lisainfo: 2.1.4)

4.1.2 Vale stiili kasutamine

Väga tihti esineb viga, et kasutatakse stiili, mis ei vasta valitud X-tee versioonile. Näiteks vana X-tee versiooniga ei saa kasutada „dokument/literal” stiili ning stiilid „rpc/literal” ja „document/encoded” ei ole üldse X-tee teenustes kasutusel. (Lisainfo: 2, 2.1.2)

4.1.3 Atribuut „encodingStyle”

Atribuut „encodingStyle” peab olema <binding><operation> plokis määratud ainult „rpc/encoded” stiili kasutamisel. Tüüpiline viga on see, et „encoded” stiili puhul unustatakse see lisada, kuid lisatakse „literal” stiili puhul. Korrektsel kasutamisel näited:

„document/literal wrapped” stiil:

```
<soap:header message="tns:standardheader" part="service" use="literal"/>
```

„rpc/encoded” stiil:

```
<soap:header message="tns:standardpais" part="andmekogu" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
```


4.2 Vastused ei sisalda päringut

X-tee teenuse väljundit kandev sõnum peab sisaldama nii päringut kui ka keha komponente. Mõnikord on andmekogude kirjeldused vigased, sest väljundite kirjeldustes puuduvad päringute osad. Korrektsed kirjeldused, mis sisaldavad ka päringu osa:

„document/literal wrapped” stiil:

```
<element name="queryResponse" />
  <sequence>
    <element name="request" type="tns:queryInput" />
    <element name="response" type="tns:queryOutput" />
  </sequence>
</element>
...
<wsdl:message name="sendSmsResponse">
  <wsdl:part name="body" element="tns:sendSmsResponse"></wsdl:part>
</wsdl:message>
```

„rpc/encoded” stiil:

```
<message name="teenusResponse">
  <part name="paring" type="tns:teenus_paring"/>
  <part name="keha" type="tns:teenus_vastus"/>
</message>
```

(Lisainfo: 2.1.3)

4.3 X-tee päised

4.3.1 Atribuut „part”

Atribuuti „part” kasutatakse <binding><operation> plokis selleks, et viidata sõnumi osale, mida see <soap:body> või <soap:header> element kirjeldab. Kusjuures <soap:body> korral ei ole vaja seda atribuuti kasutada, sest X-tee teenuste puhul koosneb <soap:body> kõikidest sõnumi osadest. Element <soap:header> aga nõuab atribuudi „part” kasutamist, sest kõik X-tee päised peavad olema eraldi ridadel.

Tüüpilised vead atribuudi „part” kasutamisel:

- <soap:body> elemendi sees kasutatakse part=„keha” atribuuti, kuigi tegemist on teenuse vastusega, mis koosneb nii „keha” kui ka „paring” osast;
- <soap:header> elemendi sees kasutatakse part=„*” atribuuti. Nii tekib valideerimise viga, sest „*” ei ole korrektne sõnumi osa.

Korrektse kirjelduse näide („document/literal wrapped”):

```
<wsdl:input name="teenus">
  <soap:header message="tns:standardHeader" part="consumer"
use="literal" />
  <soap:header message="tns:standardHeader" part="producer"
use="literal" />
  <soap:header message="tns:standardHeader" part="userId" use="literal" />
  <soap:header message="tns:standardHeader" part="id" use="literal" />
  <soap:header message="tns:standardHeader" part="service" use="literal" />
```

```
<soap:body use="literal" />
</wsdl:input>
```

4.3.2 Päised puuduvad

Sageli ei ole X-tee kohustuslikke päiseid üldse kirjeldatud. See aga on vastuolus X-tee nõuetega, sest päised edastavad olulist informatsiooni teenuse kasutamise kohta. (Lisainfo: 2.1.5)

4.3.3 Vigane päiste struktuur

Mõnikord ei vasta WSDLis kirjeldatud päiste struktuur X-tee nõuetele. Päiste sõnum on sel juhul kirjeldatud nii („rpc/encoded”):

```
<message name="standardpais">
  <part name="pais" type="xtee:hdrstd" />
</message>
```

ja <binding><operation> sees on järgmine kirjeldus:

```
<soap:header message="tns:standardpais" use="encoded" part="pais"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd" />
```

Selline kirjeldus vastab tegelikult järgmisele SOAP päringule:

```
<SOAP-ENV:Header>
  <pais>
    <asutus>STRING</asutus>
    <andmekogu>STRING</andmekogu>
    ...
```

ning <pais> element on siin üleliigne.

Korrektne oleks kirjutada sellise X-tee päise sõnum („rpc/encoded”) nii:

```
<message name="standardpais">
  <part name="asutus" element="xtee:asutus" />
  <part name="andmekogu" element="xtee:andmekogu" />
  <part name="isikukood" element="xtee:isikukood" />
  <part name="nimi" element="xtee:nimi" />
  <part name="id" element="xtee:id" />
  <part name="toimik" element="xtee:toimik" />
</message>
```

ning <binding><operation> plokis kirjutada:

```
<soap:header message="tns:standardpais" part="asutus" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<soap:header message="tns:standardpais" part="andmekogu" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<soap:header message="tns:standardpais" part="isikukood" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<soap:header message="tns:standardpais" part="nimi" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<soap:header message="tns:standardpais" part="id" use="encoded"
```

```
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<soap:header message="tns:standardpais" part="toimik" use="encoded"
namespace="http://x-tee.riik.ee/xsd/xtee.xsd"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
```

4.4 Manused

Tüüpilised probleemid on seotud sellega, et üsna tihti kirjeldatakse manused ainult sõnumites (<message>) või ainult <binding> plokkis. Tegelikult peab manuste kirjeldus olema mõlemas kohas.

Mõnikord juhtub ka nii, et WSDL dokumendis puudub üldse manuste kirjeldus, kuigi teenus kasutab neid. See omakorda tähendab, et WSDLi kirjeldus ei vasta teenuse reaalsele toimimisele. (Lisainfo: 2.1.6)

4.5 Operatsioonide nimekiri

Kõige sagedasem probleem operatsioonidega on see, et <binding> ja <portType> plokkides ei ole kirjeldatud sama operatsioonide komplekti. Üldjuhul juhtub see järgmistel põhjustel:

- WSDLi looja unustas kirjeldada operatsioone <binding> või <portType> plokkis;
- WSDLi looja kustutas vanad operatsioonid ainult ühes kohas;
- WSDLi looja kirjeldas sama operatsiooni mitut versiooni.

Viimane põhjus on seotud tavaliselt sooviga jätta alles vanad X-tee teenuste kirjeldused WSDLi sees. Selle tulemusel muutub aga WSDLi dokument vigaseks, sest WSDLi töötlevad tööriistad ei oska enam tuvastada kumb operatsioonidest on tegelikult kasutusel. (Lisainfo: 2.3)

4.6 Skeemide kasutamine

4.6.1 Mitmekordselt defineeritud elemendid

Samas nimeruumis olevad elemendid ja tüübid võivad olla defineeritud ainult ühe korra. See reegel kehtib nii ühe skeemi kohta kui ka mitme sama nimeruumiga skeemi importimisel. (Lisainfo: 2.5)

4.6.2 Väliseid elemente on kasutatud ilma importimiseta

Kui WSDLis on kasutatud teistes failides paiknevaid elementide ja tüüpide kirjeldusi, siis tuleb need WSDLis oleva skeemi sisse importida. (Lisainfo: 2.5)

4.6.3 Kättesaamatud välised skeemid

Kui WSDL impordib faili, mis asub kohalikus masinas või kohalikus võrgus, siis suure tõenäosusega teiste arendajate arvutites seda faili ei ole. (Lisainfo: 2.5)

4.6.4 Mitu skeemi WSDLi dokumendi sees

Kui WSDLi sees paikneb mitu skeemi sama nimeruumi jaoks, siis see võib üldjuhul tekitada probleemi, sest WSDLi spetsifikatsioonis ei ole see koht piisava täpsusega kirjeldatud. Parema ühilduvuse tagamiseks tuleks vältida mitme skeemi panemist WSDLi faili sisse. (Lisainfo: 2.5)

4.7 Kodeeringud

Üsna tihti juhtub, et dokumentatsiooni elementides või kommentaarides on tekstid vales kodeeringus. WSDL dokumentide loomisel tuleks alati arvestada sellega, et dokument peab olema UTF-8 kodeeringus. Alati tuleb kasutada tööriistu, mis oskavad seda korrektselt töödelda. (Lisainfo: 2.7)