

Foutafhandeling

1 Inleiding

Foutafhandeling in de gegevensuitwisseling via berichtenverkeer.

2 Analyse

Korte beschrijving van de huidige situatie.

Zowel StUF, NEN-3610 als SuwiML hebben haar eigen richtlijnen en filosofieën met betrekking tot de foutafhandeling. De analyse is gebaseerd op de huidige standaard van Digikoppeling. Digikoppeling schrijft binnen deze standaard SOAP1.1 voor als basis.

Wat bieden de vier standaarden voor dit onderwerp?

2.1 Algemeen

Alle standaarden geven aan dat op transport en protocol niveau geen standaardisatie uitgevoerd kan worden. Op dit niveau zijn al voorzieningen getroffen op internationaal niveau. De standaarden hebben daarom ook niets beschreven m.b.t. de afhandeling van fouten op dit vlak.

2.2 Digikoppeling

Digikoppeling geeft richtlijnen voor het melden van technische fouten. Technische fouten worden in de SOAP:fault beschreven (SOAP versie 1.1). Er wordt gevraagd om deze meldingen op te nemen in de service. De meldingen die Digikoppeling voorstelt, worden in de SOAP:fault faultcode en faultstring opgenomen. Functionele foutmeldingen kunnen opgenomen worden SOAP:fault detail. Hieronder de structuur van de SOAP:fault.

```
<xs:complexType name="Fault" final="extension">
  <xs:annotation>
    <xs:documentation>Fault reporting structure</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="faultcode" type="xs:QName"/>
    <xs:element name="faultstring" type="xs:string"/>
    <xs:element name="faultactor" type="xs:anyURI" minOccurs="0"/>
    <xs:element name="detail" type="tns:detail" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="detail">
  <xs:sequence>
    <xs:any namespace="##any" minOccurs="0" maxOccurs="unbounded" processContents="lax"/>
  </xs:sequence>
  <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
```

Afbeelding 1 SOAP fault-structuur uit Digikoppeling 2.0

Functionele foutmeldingen worden door Digikoppeling niet bepaald of moeten gestandaardiseerd worden. Hiervoor zijn de leverende partijen zelf voor verantwoordelijk. En geeft Digikoppeling aan dat deze meldingen gedeeld moet worden via Digikoppeling Service Register.

Binnen de koppelvlakstandaard WUS van **Digikoppeling** zijn er 4 verschillende fouttypen te onderscheiden: protocol-, transport-, functionele- en technische fouten. Protocol- en transportfouten zijn in het algemeen in het protocol zelf gedefinieerd. De foutmeldingen zijn ingebakken in de standaard software en daarom worden ze niet beschreven. Functionele fouten zijn voor verschillende organisaties verschillend en het communiceren van de foutmelding zal niet altijd eenduidig zijn. In de SOAP-Fault van de SOAP-specificatie zijn de technische foutmeldingen beschreven.

2.3 StUF

StUF ondersteund SOAP:faults . Naast de technische meldingen worden de functionele foutmeldingen in de detail van de SOAP:fault opgenomen. Hieronder de passage uit de StUF-documentatie met betrekking tot foutafhandeling.

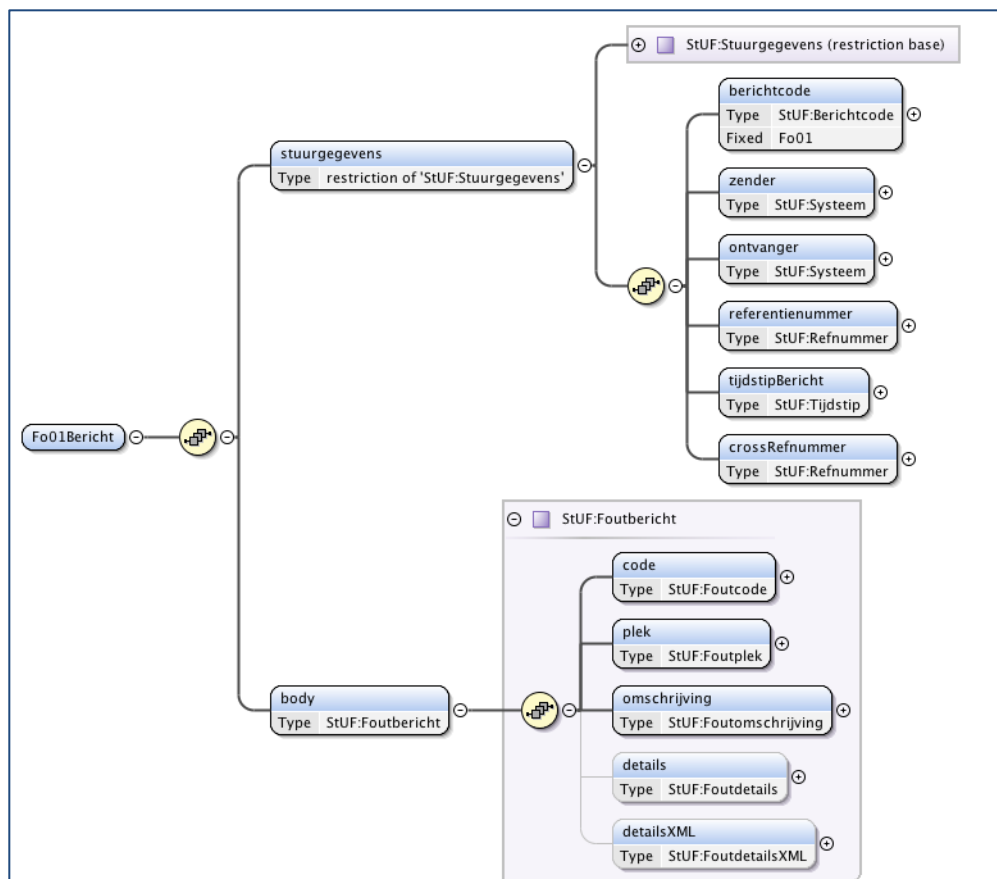
Ten behoeve van foutafhandeling kent SOAP het <SOAP:fault> element.

De elementen van het <SOAP:fault> element worden gevuld op basis van de elementen binnen het StUF-foutbericht:

- 1. Het element <SOAP:faultcode> wordt gevuld met de namespace kwalifier voor de namespace "http://schemas.xmlsoap.org/soap/envelope/" gevolgd door een ':' en de waarde van <StUF:plek> binnen het foutbericht, bijvoorbeeld "SOAP-ENV:Client".*
- 2. Het element <SOAP:faultstring> wordt gevuld met de waarde van <StUF:omschrijving> binnen het foutbericht.*
- 3. Het element <SOAP:details> wordt gevuld met het StUF-foutbericht.*
- 4. Het element <SOAP:faultactor> wordt niet opgenomen in het <SOAP:fault> element.*

Afbeelding 2 Uit protocobinding StUF

Er worden elementen uit een foutbericht gedupliceerd naar de vergelijkbare elementen binnen de SOAP:fault. En wordt het element SOAP:faultactor niet gebruikt. Dit is echter wel verplicht als de zender van de SOAP:fault niet de uiteindelijke bestemming is van het request. Daarnaast bevat een SOAP:fault detail informatie, die in de header van het bericht opgenomen zijn (denk hierbij aan de elementen van WS-adressing).



Afbeelding 3 Structuur Fo01 bericht.

StUF beschrijft in de standaard ook een lijst met mogelijke foutmeldingen. Deze lijst is door Digikoppeling als basis gekozen om de standaard Digikoppeling lijst samen te stellen.

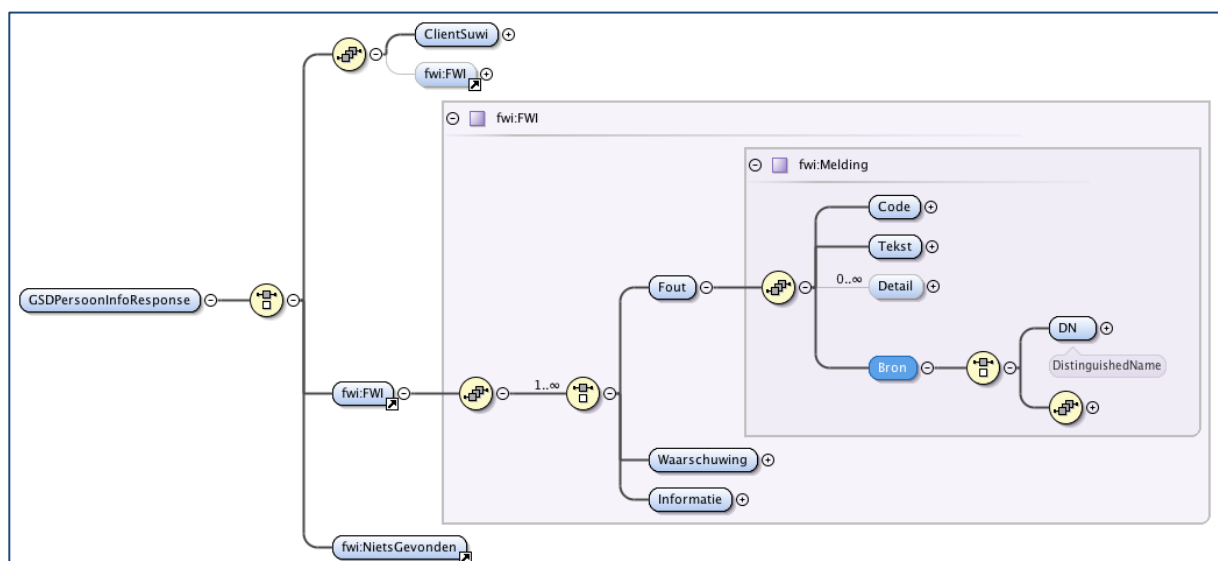
StUF maakt voor verschillende soorten berichtuitwisselingen (synchroon, asynchroon en technische synchrone respons bij verzending van een asynchroon bericht) onderscheid in welke detailinformatie bij foutsituaties gecommuniceerd moet worden. Niet elke foutmelding bevat daarmee dezelfde informatie in het onderdeel SOAP:fault detail.

2.4 NEN-3610

NEN-3610 houdt zich voor foutmeldingen aan de internationale standaarden. Geonovum maakt gebruik van een validatieservice (validator) en conformiteitstoetsing om te controleren of de standaarden technisch correct zijn toegepast. De foutmeldingen zijn afgeleid van de internationale standaarden. In de interactiepatronen wordt er verwezen naar de foutmeldingen. Omdat NEN-3610 ook gebruik maakt van StUF, zal de methode van het melden van foutmeldingen gelijk zijn.

2.5 SuwiML

SuwiML geeft in de standaard aan, dat voor het melden van fouten ook de SOAP:fault gebruikt moet worden. Net zoals bij StUF wordt in de SOAP:fault ook een detail meegeleverd, op basis van een in het koppelvlak gedefinieerd Foutmeldingen blok (FWI). FWI staat voor fout, waarschuwing en informatie. Als er een fout optreedt wordt het onderdeel FWI als detail in de SOAP:fault opgenomen. Er wordt echter geen informatie vanuit de het FWI-deel overgenomen in SOAP:fault elementen overgenomen. Hier worden meldingen vanuit de broker opgenomen. Net zoals bij StUF wordt er door SuwiML geen gebruik gemaakt van SOAP:faultactor.



Afbeelding 4 Structuur FWI in bericht.

In de afbeelding is te zien, dat SuwiML ook voorziet in een terugkoppeling van fouten, waarschuwingen en informatie, als er wel een response geleverd wordt - het request is goed bevonden. Het gaat dan om een situatie, dat er wel informatie geleverd wordt. En dat er bij de verwerking/levering van gegevens/informatie iets nog niet goed is gegaan. Of er kan extra informatie meegegeven worden. Bij SuwiML kunnen ook meer dan één melding voorkomen van verschillende bronnen. Maar dit mogen geen meldingen zijn, die gecommuniceerd zouden moeten worden in een SOAP:fault. Deze voorziening binnen de berichtdefinitie is gekozen, om de afnemers zelf te kunnen laten beoordelen, om een bericht in zijn geheel of deels te laten verwerken of alsnog af te keuren.

Bijvoorbeeld: Als UWV tijdens het samenstellen van de response delen van het bericht niet kan vullen. De arbeidsgegevens en uitkeringsgegevens kunnen worden geleverd, maar alleen het deelsysteem voor Arbeidsgeschiktheid heeft niet of niet op tijd kunnen reageren.

SuwiML schrijft in de standaard geen foutmeldingen voor aan gebruikers van de standaard. Maar levert de meldingen door zoals door de bronnen aangeleverd worden.

Of een bericht nu synchroon of asynchroon uitgewisseld wordt, SuwiML maakt geen onderscheid in de informatie die gecommuniceerd wordt bij foutsituaties.

2.6 De belangrijkste overeenkomsten zijn:

1. Alle betrokken standaarden zeggen dat foutsituaties op protocol- en transportniveau niet gestandaardiseerd hoeven te worden.
2. De DK standaard “gebruik de SOAP:fault voor het melden van fouten” worden door alle standaarden gebruikt.
3. Details van fouten worden door StUF, NEN-3610 en SuwiML in SOAP:fault detail opgenomen. Fouten worden gecommuniceerd in de vorm technische, functionele foutmeldingen.
4. De technische foutmeldingen worden niet vertaald naar voor gebruikers betekenisvolle meldingen.

2.7 De verschillen zijn:

1. Iedereen heeft zijn eigen lijst met foutmeldingen met eigen coderingen. En SuwiML heeft geen lijst van meldingen. Maar neemt de meldingen van de bronnen 1 op 1 over.
2. DK foutmeldingen zijn niet opgenomen binnen de standaarden. Digikoppeling heeft de voor de opgenomen foutmeldingen als basis de lijst van StUF gebruikt.
3. Melden van fouten in de body van het response *versus* Een apart foutbericht (net name bij asynchrone uitwisselingen met StUF).
4. Op verschillende niveaus foutmeldingen in de body *versus* Op één niveau in de body van een foutbericht
5. Verschillende type foutberichten voor verschillende soorten uitwisselingen (synchroon/asynchroon) *versus* Eén type afhandeling ongeacht soort uitwisseling.
6. Stuurgegevens worden niet in de melding opgenomen (alleen in de header) *versus* Stuurgegevens worden in de body van de melding opgenomen.

2.8 Mogelijkheid tot harmonisatie

Het meest opportune onderdeel tot harmonisatie is de wijze waarop de SOAP:faults gevuld worden bij de verwerking van requests. Zoals Digikoppeling al heeft erkend zullen er op functioneel gebied verschillen in meldingen zijn. Wij zullen die verschillen moeten accepteren. Maar wat de overige standaarden op zijn minst kunnen doen, zijn de meldingen op het gebied van header foutmeldingen standaardiseren. Met als vertrekpunt de lijst van Digikoppeling. Het gaat dan voornamelijk over de inhoud van het SOAP fault element Faultstring.

Ook kunnen bijvoorbeeld afspraken gemaakt worden over het format van dit element. Zodat de eigen codes die aan meldingen toegekend worden, in het eigen domein beheerd kunnen blijven.

De faultstring zou dan de volgende opbouw kunnen krijgen:

[<beheerder van de lijst met meldingen>.<code>]: <omschrijving>

In bijlage 3 is een tabel opgenomen waarin te foutstructuren van StUF en SuwiML geharmoniseerd zijn voor wat betreft de SOAP fault-elementen. Overigens zijn er een

paar elementen die extra aandacht en uitzoekwerk vereisen. Deze bijlage is feitelijk de impactanalyse die nodig is om wijzigingsvoorstellen bij beide standaarden in te dienen.

Voorbeelden:

<faultstring>[StUF.StUF055]: Berichtbody is niet conform schema in sectormodel</faultstring>
of
<faultstring>[Digikoppeling.0008]: Het bericht is niet volgens document-literal
style</faultstring>

Het element faultactor binnen de SOAP:fault wordt nu niet gebruikt. Terwijl dit toch een functie heeft binnen uitwisselingen. Het heeft wel de voorkeur om dit een duidelijke plek te geven binnen harmonisatie. En deze dan op te nemen in Digikoppeling.

Conclusie:

Een algehele en uniforme lijst met alle mogelijk fouten, zal er waarschijnlijk nooit komen. Maar er kunnen relatief gemakkelijke onderlinge afspraken gemaakt worden, om een verbetering door te voeren. Daarmee zal de interoperabiliteit op dit gebied vergroten tussen de standaarden. De focus ligt op gebruik van de SOAP-standaard en hergebruik. Maar er moet ruimte blijven v.w.b. domein specifieke foutmeldingen.

3 Voorstel

Korte beschrijving van het voorstel.

Voorstellen:

1. Gebruik de technische foutmeldingen volgens het protocol
2. Elimineer doublures in beschikbare lijsten.
3. Zorg voor een juiste categorisering van meldingen, zodat fouten op een juiste manier verwerkt kunnen worden. In bijlage 2 is de lijst van Digikoppeling als voorbeeld voorzien van categorieën.
4. Harmoniseren van de wijze waarop foutmeldingen binnen de SOAP:fault opgenomen worden. Houdt hierbij rekening met de hoe meldingen bij gebruikers terecht komen.
5. Houd al rekening met de overgang naar SOAP 1.2 (Of maak een vertaler voor partijen die deze versie nu al gebruiken.)

Hoe ziet de mogelijke standaard voor dit onderwerp er uit?

- Technische foutmeldingen zo veel mogelijk aansluiten op SOAP-Faults en de Nederlandse vertaling zoals in Digikoppeling is beschreven.
- Functionele foutmeldingen moeten eenduidig worden geïnterpreteerd.
- De wijze van foutafhandeling is uniform.

4 Aanpak

- Maak gezamenlijk een document, waarin de specifieke afspraken over het melden van fouten in SOAP:faults beschreven wordt.
- Adopteer en implementeer de lijst met DK meldingen die na harmonisatie ontstaat bij alle standaarden.

Bijlage 1: Een willekeurig en fictief voorbeeld ter illustratie

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Header/>
<SOAP-ENV:Body>
<SOAP-ENV:Fault>
<faultcode>SOAP-ENV:Server</faultcode>
<faultstring>[SU0098] Het bronsysteem BROKER heeft niet tijdig gereageerd. Mogelijk is het systeem tijdelijk niet beschikbaar, of is
het te druk. Probeer het later nog eens</faultstring>
<faultactor>http://bkwi.nl/SuwiML/SuwiBroker</faultactor>
<detail>
<?xml version="1.0" encoding="UTF-8"?>
<fwi:FWI xmlns:smlct="http://bkwi.nl/SuwiML/Codelijsten/Typed"
xmlns:fwi="http://bkwi.nl/SuwiML/FWI/v0201"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://bkwi.nl/SuwiML/FWI/v0201
file:/Users/tzwaan/Documents/Berichten/WerkpleinDienstverleningEWwb-v0201-b02.zip%20Folder/FWI/v0201/FWI.xsd">
  <Fout>
    <Plek>Server</Plek>
    <Code>SU0098</Code>
    <Tekst> Het bronsysteem BROKER heeft niet tijdig gereageerd. Mogelijk is het systeem tijdelijk niet beschikbaar, of is het te
druk. Probeer het later nog eens </Tekst>
    <Bron>
      <DN>cn=GWS4All,o=Gemeentelijke Socialedienst Leiden, c=nl</DN>
    </Bron>
    <Detail>Time out generated, service did not respond within 30.000 ms</Detail>
  </Fout>
</fwi:FWI></detail></SOAP-ENV:Fault>
</SOAP-ENV:Body></SOAP-ENV:Envelope>
```

Bijlage 2: Categorisering technische foutmeldingen

Verschillende fout-categorieën zijn:

1. syntax fouten, zowel op structuur als op inhoudelijke waarden cq. domein check
2. inhoudelijke fouten, waaronder ook protocol fouten, leidend tot onbestaanbare en/of inconsistente situaties
3. fouten doordat een service niet (onvoldoende QoS) beschikbaar is, waaronder ook time-out en autorisatie problemen/fouten.

Per categorie kan op hoofdlijnen een procedure voor de foutafhandeling gedefinieerd worden.

1. bij syntax fouten dient zo mogelijk aangegeven te worden welk element fout is (zoals in foutmeldingen 0005 t/m 0008 aangegeven staat)
2. Zo mogelijk aangeven waarom . Bij protocol fouten aangeven wat de status is en wat wel verwacht wordt, bij inhoudelijke fouten aangeven dat het bericht vanwege inconsistentie niet verwerkt kan worden. (dit is eigenlijk geen transport/koppelvlak probleem , maar veeleer een business probleem met een bijbehorende afhandelingsprocedure, vgl. de terugmelding in het stelsel).

Lijst (voorbeeld)van technische foutmeldingen met classificatie naar fout categorien

nr	omschrijving	categorie	toelichting
0001	invalide envelope	1	Voldoet niet aan verwachte syntax. Structuur van de envelope matcht niet met wat er verwacht wordt
0002	requester systeem niet geauthoriseerd	3	Service niet beschikbaar (QoS). Door gebrek aan bevoegdheden.
0003	request bericht invalide soap action	2	De inhoud leidt niet tot een voltooide actie, is niet gedefinieerd of onbegrijpelijk. Protocol fout
0004	element niet conform xsd	1	Voldoet niet aan verwachte syntax
0005	WS addressing header "to" ontbreekt	1	Voldoet niet aan verwachte syntax
0006	WS addressing header "action" ontbreekt	1	Voldoet niet aan verwachte syntax
0007	WS addressing header "messageID" ontbreekt	1	Voldoet niet aan verwachte syntax
0008	WS addressing header "relates to" ontbreekt	1	Voldoet niet aan verwachte syntax
0009	Niet volgens utf	1	Voldoet niet aan verwachte characterset
0010	Headers anders dan WS-headers	1	Voldoet niet aan verwachte syntax
0011	Header andere waarde dan voorgeschreven	1	Voldoet niet aan verwachte spec/waarde
0050	proces voor afhandelen bericht geeft fout	2	De inhoud leidt niet tot een voltooide actie, is niet gedefinieerd of onbegrijpelijk.
0051	antwoordende systeem geeft time-out fout	3	Service niet beschikbaar (QoS). Door gebrek aan resources/ verwerkings-capaciteit.
0100	opslaan bericht niet mogelijk ¹	1	Voldoet niet aan verwachte syntax Bericht is onleesbaar
0100	opslaan bericht niet mogelijk	2	De inhoud leidt niet tot een voltooide actie, is niet gedefinieerd of onbegrijpelijk. Bericht is inconsistent

¹ Voor deze foutcode is het niet duidelijk waarom het bericht niet opgeslagen kan worden, vandaar dat het voor de 3 categorien uitgesplitst is.

			met overige inhoud
0100	opslaan bericht niet mogelijk	3	Service niet beschikbaar (QoS). Database/storage is niet beschikbaar, (mogelijk door gebruik/lock van ander systeem)

Bijlage 3 Mapping en harmonisatie StUF en SuwiML op SOAP fault elementen

StUF →	StUF “nieuw”	SOAP fault element	SuwiML “nieuw”	←SuwiML nu
plek - String (enumeration) - Verplicht (1..1)	Plek (nog verder analyseren)	<faultcode>	Plek	Vanuit service
code - string lengte 7 - verplicht (1..1)	code - string lengte ∞ - verplicht (1..1)	<faultstring> (conform voorstel)	Code - string lengte ∞ - verplicht (1..1)	Code - string lengte ∞ - verplicht (1..1)
omschrijving - string lengte 200 - verplicht (1..1)	omschrijving - string lengte ∞ - verplicht (1..1)		Omschrijving - string lengte ∞ - verplicht (1..1)	Tekst - string lengte ∞ - verplicht (1..1)
Niet gebruikt.	Vullen met informatie uit de stuurgegevens van het Fo-bericht.	<faultfactor>	Uit de SuwiML header of Bron FWI.	Niet gebruikt.
details - string lengte 1000 - optioneel (0..1)	details - ANY lengte ∞ - optioneel (0..∞)	<detail>	Details - ANY lengte ∞ - optioneel (0..∞)	Details - string lengte ∞ - optioneel (0..∞)
detailsXML - ANY lengte ∞ - optioneel (0..∞)	vervalt			