

TOP-KT-012b - FHIR REST Client foutafhandeling

Beschrijving

In de interactie met de FHIR REST API kunnen zich, naast de fouten die zich voordoen aan de serverzijde, er zich ook fouten voordoen bij de verwerking van de resources door de cliënt van de FHIR REST API. Het is van belang deze fouten te melden, omdat anders de infrastructuur ervan uit kan gaan dat het clientsysteem de informatie correct verwerkt heeft, of correct kan verwerken. De foutafhandeling door de client bestaat uit het aanmaken van een AuditEvent, hiermee maakt de applicatie duidelijk aan de andere applicaties en de beheerder van het domein dat er iets niet goed is gegaan. Het type AuditEvent maakt duidelijk wat er exact aan de hand is. Een applicatie kan een FHIR resource om verschillende redenen niet verwerken. Dit onderscheid is belangrijk, omdat sommige fouten verwacht kunnen zijn en geen aandacht van een beheerder vereisen, terwijl andere type fouten onverwacht zijn en aandacht vereisen.

Overwegingen

Grijze gebieden in het FHIR profiel

Bij het vaststellen van het FHIR profiel is er een ultieme inspanning gedaan deze enkel en eenduidig te houden. Echter, bij een klein aantal velden is er een onmogelijkheid voorgekomen; in een bepaalde situatie bleek een veld totaal niet van toepassing, terwijl het in een andere situatie juist verplicht bleek. De leveranciers hebben hierover gestemd en besloten dat de koppeltaal infrastructuur er vanuit moet gaan dat het veld in deze situatie niet verplicht is, en applicaties, ook degene die het veld vereisen, hiermee om moeten gaan.

Daarnaast zijn er applicaties die een veld nodig hebben voor de juiste verwerking, terwijl dit veld in de meeste contexten onnodig of onbeschikbaar is. In deze situatie geldt tevens dat de applicatie die het veld nodig heeft moet kunnen aangeven de FHIR resource niet goed verwerkt kan worden.

In de Koppeltaal specificatie moet om bovengenoemde het scenarios als realistisch beschouwd worden. De implicatie hiervan is dat de specificatie vastgesteld wat het verwachte gedrag van de applicaties in dergelijk scenario.

Onverwerkbare informatie

Naast het grijze gebied in het FHIR profiel doet zich nog een ander probleem voor. Het FHIR profiel legt vast welke velden kunnen worden verwacht, en wat de inhoud van de velden is. Dit voorkomt niet dat in de verwerking van de velden zich er een probleem voordoet. Het kan zo zijn dat een veld illegale inhoud kent, bijvoorbeeld een illegaal e-mail adres, postcode of telefoonnummer. De validatieregels van de applicatie kunnen nu eenmaal strenger zijn dan Koppeltaal en het FHIR profiel deze stelt. Verder zijn er mogelijk restricties op het datamodel die van toepassing kunnen zijn. Kortom, er zijn legio aan redenen dat een applicatie een resource niet goed kan verwerken.

Verwacht vs. onverwacht, herstelbaar vs onherstelbaar (repareerbaar)

In de foutafhandeling is een onderscheid te maken tussen een aantal belangrijke facetten. Een van die facetten is of een fout in de kern verwacht of onverwacht is. Een verwachte fout doet zich voor als men

redelijkerwijs kan voorspellen dat een fout zich voordoet. Als bekend is dat een applicatie een veld verplicht heeft, maar het domein het niet verplicht stelt, kan men een fout van de applicatie die het verplicht stelt verwachten. Indien er zich een interne systeemfout voordoet bij de verwerking van een resource door de applicatie, kan men deze fout niet verwachten. Dit onderscheid is belangrijk, omdat onverwachte systeemfouten aandacht vereisen van de rest van de applicaties in het domein, en verwachte fouten niet.

Een ander facet van foutafhandeling is of de fout herstelbaar is of niet. Een verwerkingsfout die voortkomt uit tijdelijke problemen in de infrastructuur kunnen hersteld worden door de resource op een ander moment opnieuw proberen te verwerken. Een fout die voortkomt uit een validatiefout of ontbrekend veld kan niet hersteld worden door het opnieuw aanbieden van dezelfde versie van de resource.

Informeren

Het doel van de foutafhandeling aan de kant van de cliënt van de FHIR resource service is informeren. De cliënt moet in staat zijn de andere applicaties in het domein ervan op de hoogte te stellen dat de applicatie de resource niet goed verwerkt heeft. Wat de applicatie hiermee doet bepaald koppeltaal niet. Mogelijk kan een EPD inzichtelijk maken dat een resource door een van de applicaties in het domein niet goed verwerkt is.

TODO: overweging over bevestiging en verwerking en lazy vs direct. Afspraken tussen applicaties.

TODO: iets zeggen over scope: namelijk read. Alle ander meldingen worden door de FHIR server gemeld.

Toepassing, restricties en eisen

Aanleiding

De aanleiding van het aanmaken van een AuditEvent van dit type een fout die zich voordoet in een van de applicaties, niet fouten die zich voordoen bij de interacties met de FHIR resource service. Typisch, maar niet uitsluitend, komen de fouten voor in de volgende situaties:

- Het verwerken van resources die zijn opgehaald bij de FHIR resource service in het eigen systeem.
- De verwerking van FHIR resources betrokken in een update ontvangen uit een abonnement (subscriptie, Subscription). Dit is een specieker geval van het vorige item.
- Het doen van een launch tussen applicaties en de uitwisseling van gegevens en FHIR resources die hierbij betrokken zijn.

Typisch komen de situaties voort uit een gebeurtenis die buiten de applicatie ligt (subscriptie update, launch), maar zijn niet direct in de interactie met een service call en asynchroon. Het doel van het AuditEvent is de infrastructuur te laten weten dat iets niet juist is gegaan of niet juist verwerkt is.

Het AuditEvent

Voor terugkoppeling van een foutstatus moet de applicatie een AuditEvent object aanmaken. Voor de correcte verwerking maken we gebruik van het 'geen nieuws is goed nieuws' principe. Dit omdat de FHIR resource

service reeds AuditEvents vastlegt, ook van de interacties met de cliënt. Indien er een situatie zich voordoet waarbij de verwerking niet juist verloopt moet de applicatie een audit event aanmaken.

Type fouten

In koppelstaal gaan we ervan uit dat er drie type fouten zich kunnen voordoen aan de zijde van de client. In het onderdeel [scenarios](#) worden voorbeelden van deze type fouten gegeven. De drie type fouten zijn:

- Een tijdelijke fout, het systeem van de client kan de resource tijdelijk niet goed verwerken. De verwachting is dat dit in een later stadium wel kan. De client is verantwoordelijk voor het correct verwerken van de resource in een later stadium.
- Een gegevensverwerkingsfout, het systeem kan een resource niet verwerken omdat er iets in de gegevens ontbreekt of incorrect is waardoor de gegevens niet verwerkt kunnen worden. Deze valt in de categorie voorspelbaar, omdat vooraf tot bepaalde mate te voorspellen is dat de verwerking niet lukt omdat de gegevens ontbreken of in het juiste formaat zijn. Voorbeelden zijn een ontbrekend e-mail adres of een incorrect telefoonnummer. Deze fout kenmerkt zich door als oorzaak striktere of andere vereiste velden of validaties van waarden.
- Een terminale fout, het systeem van de cliënt kan de resource niet goed verwerken, en de verwachting is dat dit in de toekomst ook niet kan. Het opnieuw verwerken heeft geen zin. Dit type fout is onverwacht en heeft typisch te maken met bijvoorbeeld reeds bestaande gegevens die conflicterend zijn. Dit type is onverwacht, omdat deze in de kern niet te maken hebben met het inhoud van het bericht enkelvoudig, maar met de combinatie van die inhoud van het bericht en de status van het systeem van de cliënt.

De tabel beneden vat de structuur van het type fouten samen.

	Tijdelijke fout	Gegevensverwerkingsfout	Terminale fout
onverwacht/verwacht	onverwacht	verwacht	onverwacht
herstelbaar	ja	nee	nee

Wanneer een AuditEvent aan te maken

Het doel van het AuditEvent is de rest van het domein te laten weten dat de applicatie niet in staat is het FHIR object te verwerken op een dergelijke manier dat in het domein verwacht wordt. Dit houdt in dat een applicatie FHIR objecten kan negeren zolang duidelijk is dat de applicatie er niets mee doet. In de toekomst wordt overwogen op basis van FHIR profielen per domein of applicatie aan te geven welke entiteiten en velden binnen de entiteiten verplicht of verwacht zijn. Vooralsnog volstaat met een AuditEvent aan te geven dat de verwerking niet gelukt is. Het AuditEvent is een manier om als applicatie aan de andere applicaties in het domein aan te geven dat de verwerking niet goed gegaan is.

Scenario's

De applicatie mist een veld dat optioneel is. Dit is een verwachte fout die voorkomt uit de discrepantie van wat de applicatie verwacht en in het domein verplicht is. Dit is een gegevensverwerkingsfout.

De applicatie ontvangt een waarde in een veld dat niet voldoet aan de eigen validatieregels. In dit geval ontvangt de applicatie een waarde in een veld die geldig is voor het koppelstaaldomein, maar niet verwerkbaar

is in het eigen systeem. Denk aan een e-mail adres of een telefoonnummer dat aan meer eisen moet voldoen dan koppelstaal afdwingt. Dit is een gegevensverwerkingsfout.

De applicatie ontvangt een waarde die in conflict is met de staat van het systeem. Het verschil met het vorige scenario is dat de fout niet zozeer in de ontvangen gegevens zit, maar in de bestaande gegevens van het systeem. Een voorbeeld is een bestaande gebruiker met een e-mail adres of telefoonnummer die reeds bestaat en verder niet overeenkomt met de gegevens uit het FHIR resource. Dit type fout is een terminale fout.

De applicatie kan een subsysteem niet benaderen. De applicatie kan de entiteit niet verwerken omdat bijvoorbeeld de database niet benaderbaar is. In dit geval kan een tijdelijke fout verzonden worden.

De mapping van het AuditEvent

Het AuditEvent wordt volgens de tabel beneden gemapped. In de mapping doen we een aantal aannames die we uitteenzetten.

Veldnaam	Waarde
extension.request-id	De waarde van de betrokken X-Request-Id veld.
extension.correlation-id	De waarde van de betrokken X-Correlation-Id veld.
extension.trace-id	De waarde van de betrokken X-Trace-Id veld.
type	<pre>1 { 2 "system": "http://terminology.hl7.org/CodeSystem/audit-event-type" 3 "code": "rest" 4 }</pre>
subtype	Afhankelijk van het type actie die wordt uitgevoerd. <pre>1 { 2 "system": "http://hl7.org/fhir/restful-interaction" 3 "code": "read" 4 "display": "read" 5 }</pre>
action	R (Afhankelijk van het type actie die wordt uitgevoerd)
recorded	Timestamp van vastlegging, bijvoorbeeld: 2024-02-05T12:06:13+0000
outcome.code	"Verwacht" 4 "Onverwacht" 8
outcomeDesc	Een door mensen te begrijpen foutmelding.
agent.who	De client_id van de eigen applicatie <pre>1 { 2 "reference": "Device/<id client_id>" 3 }</pre>
agent.type	<pre>1 { 2 "coding" : { 3 "system": "http://dicom.nema.org/resources/ontology/DCM" 4 "code": "110150"</pre>

	<pre> 5 "display": "Application" 6 } 7 }</pre>
agent.requestor	true (Deze applicatie is de requester)
entity.what	Referentie naar de specifieke resource en versie, indien van toepassing.
	<pre> 1 { 2 "reference": "<ResourceType>/<id>/_history/<version>" 3 }</pre>
entity.query	De betrokken query, indien van toepassing. De query wordt base64 geëncodeerd opgeslagen.
source.site	Base URL van de applicatie waar het event wordt vastgelegd.
source.observer	De device reference van het device dat het event waarneemt, in dit geval de applicatie die het event aanmaakt.
	<pre> 1 { 2 "reference": "Device/<id client_id>" 3 }</pre>

De tracing headers

De tracing headers dienen worden overgenomen uit het originele read request wat betrokken is bij de fout. Verder heeft het POST request van het AuditEvent zelf ook nog de tracing headers. Deze moeten als volgt gevuld worden:

- X-Request-Id, een nieuwe waarde.
- X-Correlation-Id, de waarde van de originele X-Request-Id van de FHIR resource service.
- X-Trace-Id, de waarde van de originele X-Trace-Id van de FHIR resource service.

Het fouttype

De verschillende type fouten worden gemapped door middel van twee velden. De type en outcome. Hier wordt onderscheid gemaakt tussen drie typen:

- Een tijdelijke fout kan van type 4 (kleine fout) of 12 “fatale fout” indien het systeem tijdelijke beschikbaarheidsproblemen heeft.
- Een gegevensverwerkingsprobleem wordt altijd op outcome 4 gemapped.
- Een interne verwerkingsfout die in principe niet aan de input data valt te koppelen wordt altijd outcome 8.

In alle gevallen is het type "transmit"

	Tijdelijke fout	Gegevensverwerkingsfout	Interne fout
AuditEvent.type	transmit	transmit	transmit
AuditEvent.outcome	4 of 12	4	8

In het codesysteem <http://hl7.org/fhir/audit-event-outcome> worden de foutcodes als volgt vastgelegd.

Code	Display	Definition
------	---------	------------

0	Success	The operation completed successfully (whether with warnings or not).
4	Minor failure	The action was not successful due to some kind of minor failure (often equivalent to an HTTP 400 response).
8	Serious failure	The action was not successful due to some kind of unexpected error (often equivalent to an HTTP 500 response).
12	Major failure	An error of such magnitude occurred that the system is no longer available for use (i.e. the system died).

De action

De action wordt gemapped op E van Execute, omdat het een verwerking van gegevens in de applicatie.

Entity en query

De overige velden wijken niet af van wat verwacht wordt. De entity heeft betrekking op de entity die onverwerkbaar is, indien er in een resultaat (bundle) meerdere enties voorkomen die niet verwerkt kan worden, moet een AuditEvent worden aangemaakt met meerdere enties. Indien er sprake is van een query moet deze in het query veld gevuld worden.

outcomeDesc

De waarde van dit veld kan gevuld worden met een begrijpelijke beschrijving van de melding, met als doelgroep de beheerder van het domein waarin duidelijk leesbaar is wat het probleem heeft veroorzaakt. Hou er rekening mee dat de beschrijving absoluut geen herleidbare gegevens mag bevatten.

Eisen

- Een client van de FHIR resource service MOET een AuditEvent aanmaken indien deze niet in staat is de gegevens van de FHIR resource service als door het domein wordt verwacht te verwerken. Het AuditEvent is een manier om het domein te laten weten dat de applicatie het FHIR object niet in goede order verwerkt heeft.
- Het applicatie moet de relevante waarden van X-Request-Id, X-Correlation-Id, en X-Trace-Id uit het originele request in het AuditEvent meesturen.
- De POST request headers moet een nieuw X-Request-Id bevatten, de waarde van het X-Correlation-Id als de X-Trace-Id meegeven volgen de standaard regels rond het vullen van deze waarden.
- De mapping uit het onderdeel [De mapping van het AuditEvent](#) is van toepassing op de inhoud van het AuditEvent object.

Voorbeelden

Links naar gerelateerde onderwerpen

{}