

TOP-KT-014 - Versiebeheer en -beleid

✓ Versiegeschiedenis...

Versie	Datum	Status	Wijzigingen
1.0.0	27 Feb 2023	definitief	Geen wijzigingen sinds laatste concept
0.1.0	9 Feb 2023	concept	

Beschrijving

Wanneer resources worden uitgewisseld, is de toepasselijke FHIR-versie van toepassing op de gehele interactie, niet alleen op de resources. De semantiek van de MIME-types, de RESTful URL's, de zoekparameters en de algehele interactie zijn gebonden aan een bepaalde FHIR-versie.

Overwegingen

Waarom versies

Het is zeer onwaarschijnlijk dat een Koppeltaal interface ongewijzigd blijft. Als de Koppeltaal eisen in der loop ter tijd veranderen, kunnen er nieuwe verzamelingen van FHIR resources, profielen en interacties worden toegevoegd, kunnen de relaties tussen de resources veranderen en kan de structuur van de gegevens in de resources worden gewijzigd.

Hoewel het bijwerken van de Koppeltaal interfaces voor het afhandelen van nieuwe of verschillende behoeften een relatief eenvoudig proces lijkt, moet men echter rekening houden met de effecten die dergelijke wijzigingen kunnen hebben op de aanbiedende en afnemende partijen die de verschillende Koppeltaal interfaces gebruiken. Het probleem is dat, hoewel de leverancier die een Koppeltaal toepassing levert en controle heeft over die interface, geen volledige controle heeft over de voorzieningen en toepassingen, die mogelijk worden geleverd door andere organisaties. Belangrijk is het zorgen dat bestaande toepassingen ongewijzigd blijven werken, terwijl nieuwe toepassingen kunnen profiteren van nieuwe functies en FHIR resources.

Versiebeheer moet ervoor zorgen dat een Koppeltaal interface of resource aan kan geven welke functies en FHIR resources beschikbaar worden gemaakt en een cliënt toepassing interacties kan uitvoeren die doorgestuurd kan worden naar een specifieke versie van een functie of FHIR resource.

FHIR versies

Op dit moment wordt alleen FHIR R4 (4.0) ondersteund. Mogelijk wordt in de toekomst doorontwikkeld naar FHIR R4B (4.3). Op dat moment zijn er twee FHIR versies in beeld en dient een client te weten welke versie de FHIR resource service ondersteund. Om dit te ondersteunen wordt er voor gekozen de om de FHIR versie van de FHIR service duidelijk te communiceren in zowel de URL als in de MIME-type. Door de versie in de URL vast te leggen wordt ook afgedwongen dat verschillende versies op verschillende URLs beschikbaar zijn, door

de versie als MIME-type te gebruiken in de Accept en Content-Type header kunnen fouten voorkomen worden.

API versies

De verschillende APIs, waaronder de FHIR API, kennen ook versies. Om deze kenbaar te maken, en om test en acceptatie processen te ondersteunen, mogen deze APIs een versie in hun URL gebruiken om zo kenbaar te maken met welke versie van de API er op dat moment gebruikt wordt. Indien er gebruik gemaakt wordt van versionering in de API, moet er gebruik gemaakt van een vast versioneringssysteem waarmee onderscheid wordt gemaakt tussen brekende en niet-brekende verschillen tussen de versies.

Toepassing en restricties

FHIR versies in de FHIR service

De FHIR service MOET de FHIR versie bekend maken, dit kan door de volgende methoden:

1. Capability Statement van de FHIR resource service. Deze methode is **verplicht**. In het capability statement staat de versie in het `fhirVersion` veld.

```
1 {  
2   "resourceType": "CapabilityStatement",  
3   "fhirVersion": "4.0.1",  
4   ...  
5 }
```

2. De FHIR versie onderdeel te maken van de MIME-type als parameter in de `Content-Type` header. Deze methode is **optioneel**.

```
1 Accept: application/fhir+json; fhirVersion=4.0; charset=utf-8
```

3. De FHIR versie in de URL van de FHIR service als pad toe te voegen in de URL. Deze methode is **optioneel**. Een voorbeeld voor versie `4.0.1` ziet er als volgt uit:

```
1 https://fhir.example.com/fhir/<domain>/4.0.0/metadata
```

FHIR versies vanuit de API client

De FHIR client mag gebruik maken van de `fhirVersion` parameter in de `Content-Type` of `Accept` header, maar is dit niet verplicht. De FHIR resource service moet deze valideren met de eigen FHIR versie, met uitzondering van een conversie scenario. Van dit laatste scenario wordt in koppeltaal geen gebruik gemaakt.

API versionering.

Ook de verdere APIs in het domein, zoals de autorisatie service, mogen in hun URL de versie van hun API verwerken. Indien dit gedaan wordt, moet dit volgens het volgende nummersysteem gedaan worden:

```
1 <major>.<minor>.<build>
```

- Major versie nummer, een major versie omvat een breaking change aan een voorziening of toepassing waarbij de afnemer of aanbieder de software functioneel en technisch moet aanpassen om de nieuwe versie te ondersteunen.
- Minor versie nummer, een minor versie betreft aanpassingen die de bestaande functionaliteit niet aanraakt/beïnvloedt

- Build versie nummer, dit veld is verplicht en mag gebruikt worden om revisienummers uit versiebeheer of buildomgevingen te correleren. Indien niet van toepassing moet de waarde 0 gebruikt worden.

Eisen

[VER - Eisen \(en aanbevelingen\) voor versiebeheer](#)

Links naar gerelateerde onderwerpen

FHIR versioning: <https://hl7.org/fhir/versioning.html>

Capability Statement: [CapabilityStatement \(eHealth Mogelijkheden\)](#) en <https://www.hl7.org/fhir/r4/capabilitystatement.html>

{}