

Memo

Aan
DD-API-werkgroep

Datum	Kenmerk	Aantal pagina's
14 september 2018	11202186-000-DSC-0020-C	9
Van	Doorkiesnummer	E-mail
Stef Hummel	+31(0)6 1019 8112	Stef.Hummel@deltares.nl

Onderwerp
Wijzigingen DD-API 2.0 t.o.v. 1.0

Inleiding

Het afgelopen jaar is er gewerkt aan het uitbreiden van de specificaties van de DD-API. Dit heeft geleid tot een voorstel voor versie 2.0.

De uitbreidingen zijn gericht op:

- Het aankunnen van verwachtingen, inclusief ensemble runs
- Het in meer detail kunnen beschrijven van metingen, door:
 - o interval-informatie toe te voegen
 - o gerelateerde (doorgaans afgeleide) grootheden van een meting gegroepeerd aan te kunnen bieden (d.m.v. zgn. 'aspecten-sets')

Tijdens het uitwerken zijn ook een aantal aanpassingen voorgesteld die niet met de inhoud te maken hebben, maar met het conform bestaande standaarden opbouwen van onderdelen van de response.

Een groot deel van de uitbreidingen is optioneel. Systemen die geen intervallen en aspecten-sets kennen hoeven m.b.t. deze functionaliteit geen aanpassingen in de bestaande DD-API-1.0-implementatie. (Op een kleine uitbreiding na: een endpoint `/aspectSets` dat een lege array teruggeeft om aan te geven dat er geen aspectSets zijn.)

De algemene aanpassingen en de uitbreidingen t.b.v. van verwachtingen leiden vanzelfsprekend wel tot wijzigingen in de bestaande implementatie. Het voorliggende memo beschrijft om welke wijzigingen het gaat.

Wijzingen in de response

Algemeen

Om aan te sluiten op de richtlijnen van de DSO (Digitaal Stelsel Omgevingswet) wordt op een tweetal plekken de response aangepast:

- De structuur van het *Location* resource object wordt omgezet naar GeoJSON (zie de betreffende paragraaf verderop).
- De attributen voor *paging* worden binnen een apart object geplaatst. (zie de laatste paragraaf van het memo).

Daarnaast is besloten om flexibeler om te gaan met de identificatie van *Location*, *ObservationType*, en *TimeSeries*:

Alle resource objecten met een identifier:		
Was	Wordt	Motivatie
uuid	id	De verplichting een <i>uuid</i> te moeten leveren, daar waar men vaak al een andere identifier beschikbaar heeft, leidt tot extra werk. De inhoud van het <i>id</i> veld mag een <i>uuid</i> zijn, maar ook een systeem-eigen (al of voor een mens betekenisvolle) identifier.

Nieuw Resource object Source

Het resource object *Source* beschrijft herkomst en de soort data van een tijdserie, en vervangt de resource objects *DataSource* en *TimeSeriesType*, die dus komen te vervallen.

De reponse van een tijdserie bevat nu dus een verwijzing naar een *Source*, met daarin de volgende attributen:

```
"source": {
  "process": "forecast",
  "name": "HIRLAM_7.2",
  "description": "Weather forecast model",
  "realizationCount": 24,
  "institution": {
    "name": "KNMI",
    "description": "Dutch Weath..."
  }
},
```

Als *realizationCount* groter dan 1 is, gaat het om een ensemble run.

Resource object Location

De structuur van het *Location* resource object wordt omgezet naar GeoJSON, conform richtlijnen van de DSO (Digitaal Stelsel Omgevingswet). De attributen worden verhuisd naar een "properties" deel (zie volgende pagina).

Omdat *id*, *name* en *code* dan niet meer op het hoofdniveau zitten krijgen die een prefix:

Location, wijzigingen in attributen		
Was	Wordt	Motivatie
(uu)id name code	locationId locationName locationCode	Zitten niet meer op het hoogste niveau in het object, en krijgen daarom een verduidelijkende prefix.

<i>Location, wijzigingen in attributen</i>		
Was	Wordt	Motivatie
–	referenceLevel	Als er een derde (verticale) coördinaat wordt meegegeven geeft het referenceLevel aan t.o.v. waarvan die hoogte (die altijd uitgedrukt moet worden in meters) geldt. Optioneel, default NAP.

Voorbeeld van de response van een location:

```

"location": {
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [
      4.80,
      52.2
    ]
  },
  "properties": {
    "node": {
      "id": "abcdefgh-ijkl-mnop-qrst-uvwxyz123456",
      "baseUrl": "https://some-waterboard-lizard.nl/dd/v2",
      "name": "Lizard",
      "description": "....."
    },
    "url": "https://some-url.nl/dd/v2/locations/SAB",
    "locationId": "SAB",
    "locationCode": "STUW_A_BOVEN",
    "locationName": "Lith.Stuw-A-boven",
    "crs": {
      "type": "name",
      "properties": {
        "name": "EPSG:28992"
      }
    }
  }
}

```

Daar waar DD-API 1.0 voor `location.geometry.type` alleen `Point` kende, kent DD-API 2.0 ook het type `MultiPoint` (b.v. voor een sensor die een grootte op een aantal diepten meet). Daarnaast kan goedbeschouwd ook het type `Polygon` gebruikt wordt, als het gaat om een waarde op vlak (b.v. in een polder).

Resource object ObservationType

<i>ObservationType, wijzigingen in attributen</i>		
Was	Wordt	Motivatie
extra	–	Was bedoeld voor ecologische data. Wordt nu gedekt door de DD-ECO-API

<i>ObservationType</i> , wijzigingen in attributen		
Was	Wordt	Motivatie
-	url	Toegevoegd om consistent te zijn met Location en Timeseries
-	node	Toegevoegd om consistent te zijn met Location en Timeseries
-	aspectSet	Optioneel. Biedt de mogelijkheid om meerdere 'aspecten' van een grootheid te bundelen (b.v. gemiddelde, minimum en maximum in een 10-minuten interval).
-	description	Optioneel. Toegevoegd om het observatietype uitgebreider te kunnen beschrijven.

Resource object TimeSeries

<i>TimeSeries</i> , wijzigingen in attributen		
Was	Wordt	Motivatie
start end	startTime endTime	Consequent met <i>Observations & Measurements</i> , dat onderdeel is van Aquo's <i>IM-Metingen</i> . Beide tijdstippen zijn 'inclusief': er zijn waarden beschikbaar vanaf (dus op) <code>startTime</code> , tot en met <code>endTime</code> .
Interval	intervalLength	Duidelijker. Het gaat hier om de duur tussen twee tijdstippen.
-	analysisTime	Optioneel. Nodig voor het kunnen onderscheiden van elkaar opvolgende verwachtingen. Bevat het tijdstip waarop de tijdserie (b.v. de verwachting) gegenereerd is.
-	realization	Optioneel. Nodig voor het kunnen onderscheiden van de individuele ensemble members in een ensemble run.
-	source	(Uitgebreidere) vervanging van <code>timeseriesType</code> en <code>datasource</code>
datasource	-	Valt nu onder <code>source</code>
timeseriesType	-	Valt nu onder <code>source</code>

Naast deze hernoemde en toegevoegde attributen is er een attribuut inhoudelijk gewijzigd. Het attribuut `events` kan nu het volgende bevatten:

- De `timeStamp/value data`, zoals gedefinieerd in DD-API 1.0
- Idem, maar dan naast `timeStamp` de optionele interval-informatie:
 - o `startTime` (inclusief, dus geldig vanaf)

- o `endTime` (geldig tot aan – dit is dus anders dan bovengenoemde `endTime` van de tijdserie zelf)
 - o `resultTime` (tijdstip waarop de waarde voor dit interval ter beschikking is gekomen)
- De waarden van de grootheden in een `AspectSet` (indien het `observationType` een `aspectSet` specificatie bevat; wordt momenteel alleen bij RWS gebruikt).
- De waarden op meerdere punten (indien `location.geometry` van het type `MultiPoint` is; wordt momenteel alleen bij RWS gebruikt).

Nadere uitwerking van het rapporteren van fouten

In geval van een foutmelding (d.w.z. van een HTTP-statuscode die een fout aangeeft) bevat de response een aantal attributen (allemaal optioneel) die de fout beschrijven. De naamgeving van deze attributen is conform IETF's RFC8707. Voorbeeld van de volledige response in geval van een fout:

```
{
  "type": "https://content.mkk-dl.rws.nl/c/esb/ValidatieFout",
  "title": "Hier staat wat er mis is gegaan",
  "status": 404,
  "detail": "Verplichte query parameter ontbreekt.",
  "instance": "urn:esblog:{timestamp}"
}
```

Nieuw resource object Provider

RWS wil bij het gebruik binnen van de DD-API binnen de MKK-DL aan kunnen geven welk systeem de data heeft geleverd. Daartoe kan optioneel aan het begin van de response een blok `provider` meegegeven worden:

```
{
  "provider": {
    "name": "MKK-DL",
    "supportUrl": "http://www.rws.nl/support/mkk-dl"
  }
}
```

Wijzigingen in de end points

End point /sources vervangt end point /datasources

Dit is een rechtstreeks gevolg van het vervangen van het resource object *DataSource* door *Source*.

End point */locations*

<i>/locations, wijzigingen in filter-parameters</i>		
Was	Wordt	Motivatie
name	locationName	Consequent met de naam van het attribuut in het resource object.
code	locationCode	Consequent met de naam van het attribuut in het resource object.
Node	nodeId	Expliciet maken dat het om de id van de node gaat (en niet om b.v. zijn naam).

Nieuw end point */locations/geojson*

Biedt de mogelijkheid om naar DD-format (format=json) ook een pure GeoJSON feature collection te retourneren.

<i>/locations/geojson, filter-parameters (dezelfde als bij <i>/locations</i>)</i>		
Was	Wordt	Motivatie
name	locationName	Consequent met de naam van het attribuut in het resource object.
code	locationCode	Consequent met de naam van het attribuut in het resource object.
node	nodeId	Expliciet maken dat het om de id van de node gaat (en niet om b.v. zijn naam).

End point */timeSeries*

<i>/timeSeries, wijzigingen in filter-parameters</i>		
Was	Wordt	Motivatie
start	startTime	Consequent met de naam van het attribuut in het resource object.
end	endTime	Consequent met de naam van het attribuut in het resource object.
observationType	observationTypeId	Expliciet maken dat het om de id van de het observationType gaat (en niet om b.v. het object of een van de andere attributen daaruit).
-	realization	Nodig voor het kunnen specificeren van het gewenste ensemble member van een ensemble run.
-	aspectSet	Alleen voor systemen die aspect sets implementeren. Nodig voor het kunnen specificeren van b.v. de gewenste groep golfparameters.

<i>/timeSeries, wijzigingen in filter-parameters</i>		
Was	Wordt	Motivatie
–	sourceName	Biedt de mogelijkheid om te filteren op de bron van de data. Implementatie optioneel. Momenteel alleen nodig voor systemen die aan RWS' MKK-DL moeten toeleveren.

Nieuw end point /aspectSets

Dit end point retourneert de aspectSets die een systeem kent. Systemen die geen aspectSets kennen retourneren een lege lijst.

Paging

De API-strategie van de DSO bevat onderstaande richtlijn voor paging.

```
{
  "_links": {
    "self": {
      "href": "https://.../api/v1/aanvragen?pageSize=4&page=3"
    },
    "first": {
      "href": "https://.../api/v1/aanvragen"
    },
    "prev": {
      "href": "https://.../api/v1/aanvragen?pageSize=4&pagina=2"
    },
    "next": {
      "href": "https://.../api/v1/aanvragen?pageSize=4&pagina=4"
    },
    "last": {
      "href": "https://.../api/v1/aanvragen?pageSize=4&pagina=5"
    }
  }
}
```

We nemen dit over, waarbij we dan:

- alleen prev en next eisen, en first, last en self optioneel maken.
- onze count en maxPageSize (en ook een minPageSize) toevoegen
- voor de duidelijkheid count hernoemen naar totalObjectCount.
- Het "href" tussenniveau weglaten, omdat we altijd met de url's werken.

De bestaande DD-API-1.0 paging-attributen verhuizen dus naar een blok "paging". (Dit prefereren we boven _links, omdat het beter aangeeft waar het om gaat)

Voorbeeld:

```
{
  "paging": {
    "totalObjectCount": 17,
    "prev": "https://.../dd/v2/locations?pageSize=4&page=2",
    "next": "https://.../dd/v2/locations?pageSize=4&page=4",
    "maxPageSize": 100,
    "minPageSize": 2
  }
}
```

Generieke specificatie van filter parameters

We voorzien dat er in de toekomst uitbreidingen gewenst zijn met betrekking tot de mogelijke query-parameters die een end point biedt.

Om niet bij elke gewenste nieuwe parameter de specificaties aan te hoeven passen kennen we een algemeen mechanisme om een query samen te stellen. De manier om een query te formuleren is overgenomen uit de DD-ECO-API, en kent de onderstaande syntax (multi-line weergegeven, maar vanzelfsprekend bedoeld als één regel:

```
<resourceType>filter=
  <variable>:<comparer>:<comparevalue>
  [;<variable>:<comparer>:<comparevalue>]
```

Voor de comparer kent de DD-ECO-API een groot aantal mogelijkheden (zie de tabel op de volgende pagina). In de DD-API-2.0 definiëren we in eerste instantie alleen de 'eq' operator (equal).

Om de filter-specificaties herbruikbaar te maken binnen de end points worden drie 'resourceType' filters gedefinieerd:

- locationFilter=<query-formulering>
- observationTypeFilter=<query-formulering>
- timeseriesFilter=<query-formulering>

Het implementeren van de filters is volledig optioneel. Een systeem dat de DD-API implementeert moet in zijn documentatie aangeven welke resource-type-filters hij kent op welke end points, en op welke variabelen (d.w.z. attributen van de resource objecten) gefilterd kan worden.

Kandidaten voor attributen waarop gefilterd kan worden zijn b.v.:

-	quantity	Biedt de mogelijkheid om rechtstreeks te filteren op de grootheid (b.v. concentratie), zonder via het observationType te hoeven gaan.
-	parameterCode	Biedt de mogelijkheid om rechtstreeks te filteren op parameterCode (b.v. concentratie), zonder via het observationType te hoeven gaan.

Voor de volledigheid de comparers in de DD-ECO-API:

Comparer	Description	Data types
eq	equal	string, date, number
ne	unequal	string, date, number
lt	less than	date, number
le	less than or equal to	date, number
ge	greater than or equal to	date, number
gt	greater than	date, number
in	item is one of the values in the list	string, number
not	item is NOT one of the values in the list	string, number
like	String contains the value.	string
startswith	String starts with the value.	string
endswith	String ends with the value.	string
inbbox	Item is within the bounding box, defined by 4 numbers	bbox
inpolygon	Item is within the polygon, defined by list of numbers, where each x and y are a numbers in the array	bbox
notinbbox	Item is NOT within the bounding box, defined by 4 numbers	polygon
notinpolygon	Item is NOT within the polygon, defined by list of numbers, where each x and y are a numbers in the array	polygon
wkt	Item is within the Well-known-text definition	Wkt
all	all items in the list must be present in the queried item	string, number