

Overige Registraties

Objecten en Objecttypen

Joeri Bekker

joeri@maykinmedia.nl

Maykin Media

webdevelopment

Definitie: Objecttype

“Een in JSON-schema vastgelegde definitie van een object tezamen met metadata. Elk objecttype vertegenwoordigt een verzameling objecten met vergelijkbare vorm en/of functie.”

Voorbeelden:

“Boom”, “Melding”, “Vordering”

Enigszins vergelijkbaar met:

Class (informatica), Informatietype (informatiemodel),
Entiteittype (UML), Zaaktype (RGBZ), etc.

Definitie: Object

“Een op zichzelf staand geheel van gegevens met een eigen identiteit”

Voorbeelden:

“Boom UTR-1958396”,

“Melding AMS-2020-000015”,

“Vordering 550e8400-e29b-41d4-a716-446655440000”

Enigszins vergelijkbaar met:

Instantie (informatica), Informatieobject (informatiemodel),
Entiteit (UML), Zaak (RGBZ), etc.

Doelen

- Uiteenlopende type objecten (**objecttypen**) standaardiseren
- Uiteenlopende **objecten** eenvoudig registreren en ontsluiten
- Objecten kunnen zowel **open-** als **niet-open** data zijn
- Zowel eigen objectentypen als landelijke objecttypen gebruiken
- Sneller **standaardiseren** en direct **in productie gebruiken**

Project scope

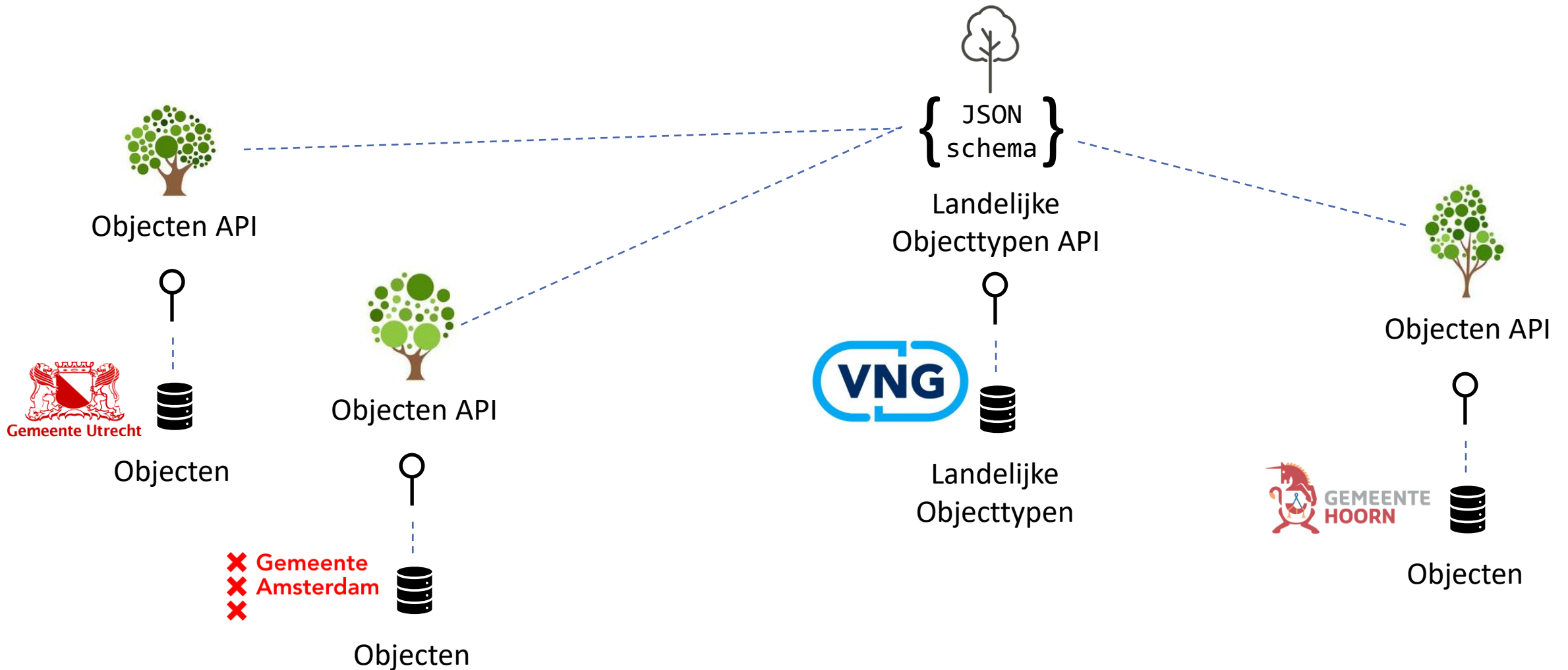
- Objecten en Objecttypen API-specificatie
- Objecten en Objecttypen API implementatie
- Minimale applicatie voor beheer van Objecttypen
- Documentatie van de API specificatie en software
- 3 voorbeeld objecttypen met bijbehorend JSON-schema
- Standaardisatie van de API bij VNG
- Standaardisatie proces inrichten voor objecttypen

En verder...

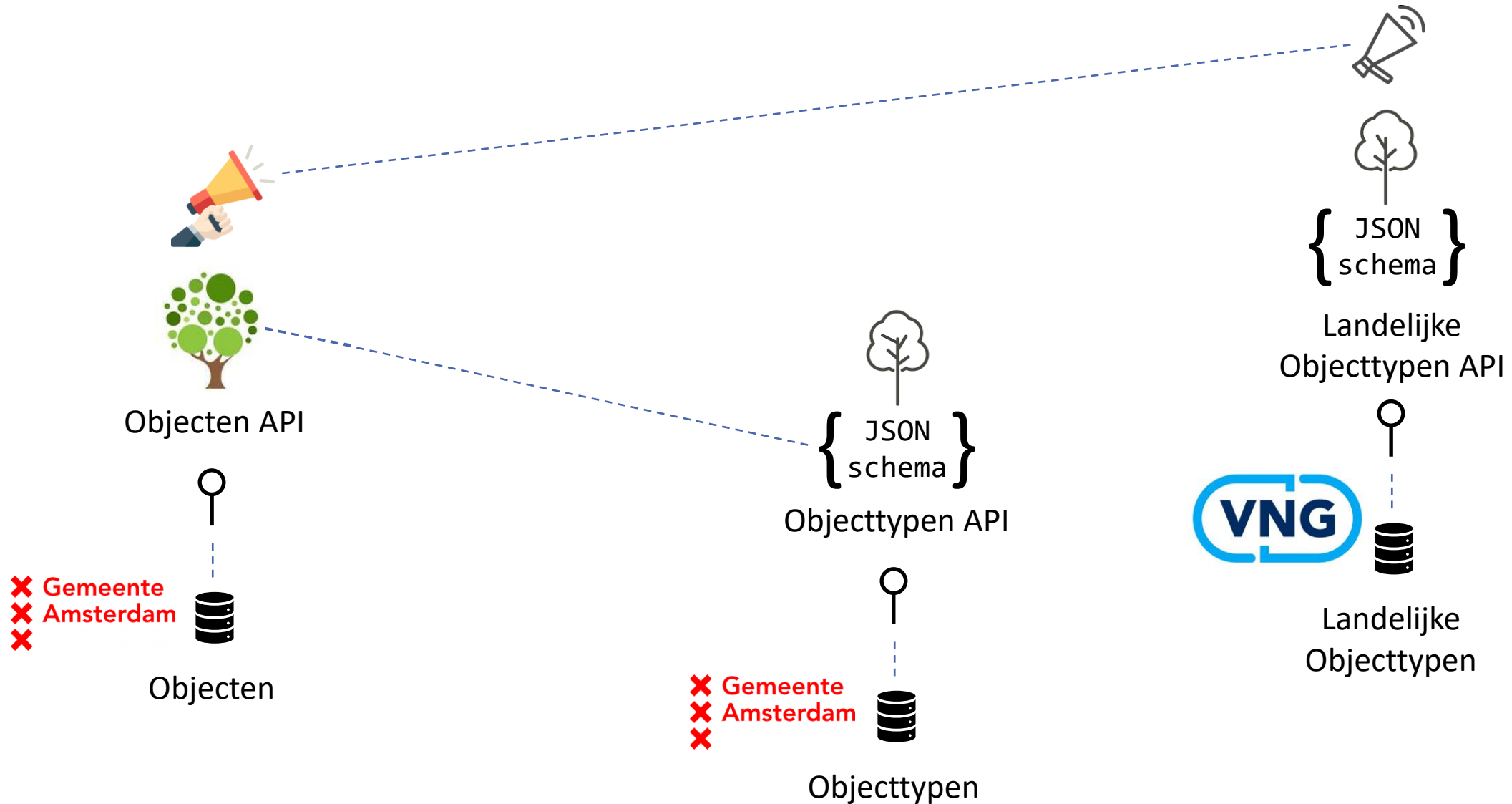
- Alles staat op Github
- Compliant met VNG standaarden (WIP)
- Compliant met Landelijke API strategie
- Alle code wordt getest middels Continuous Integration (CI)
- API specificaties worden gevalideerd voor code generatie en linting
- Er worden automatisch Docker containers gemaakt
- Postman scripts met de meeste API calls
- Er is een quickstart tutorial
- Demo applicatie t.b.v. demo's

Achtergrond

Hoe werkt het?

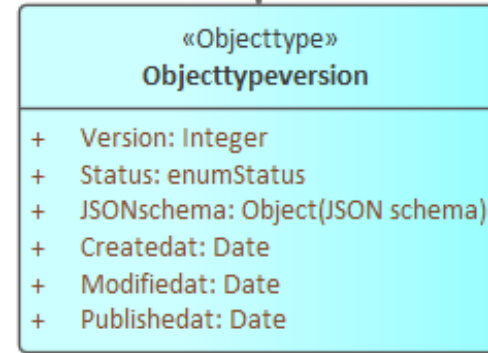
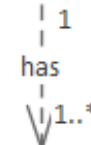
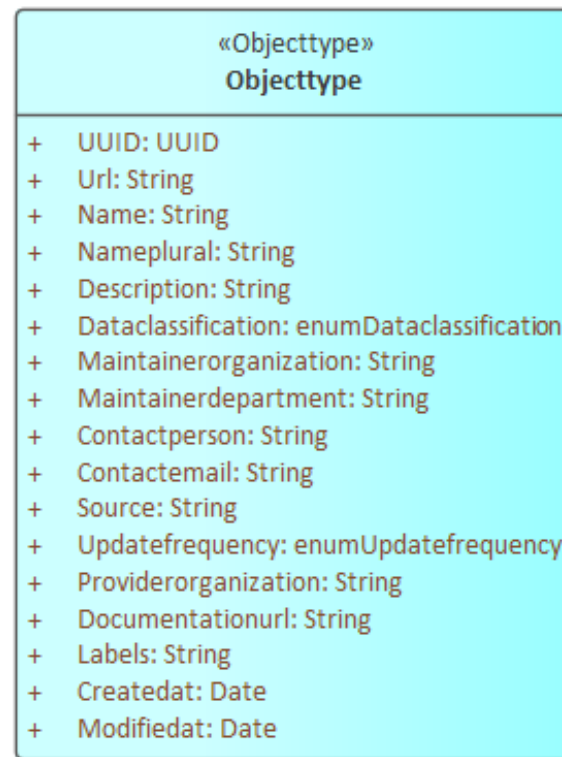
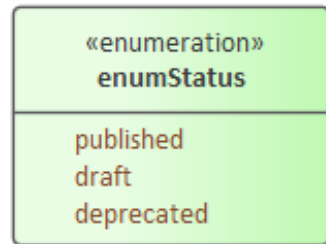
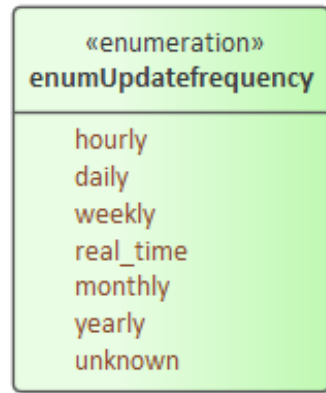
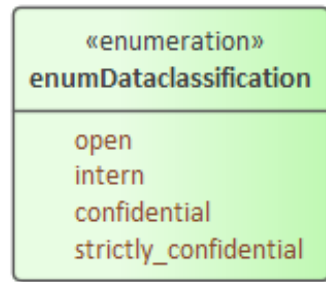


Hoe werkt het?

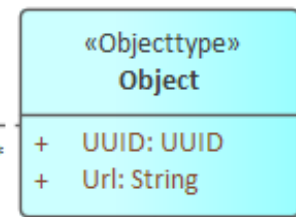
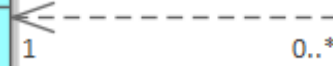


Informatie model

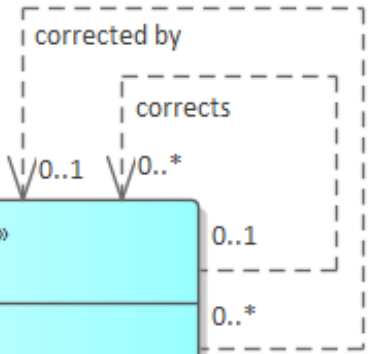
class IM Objecten en Objecttypen



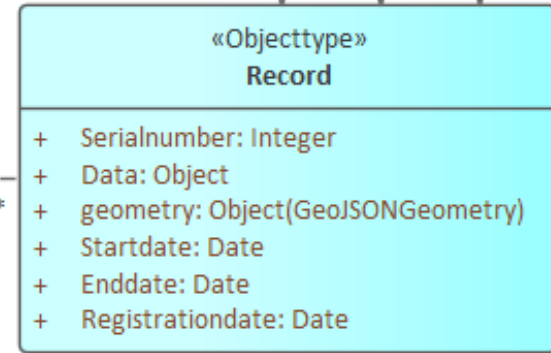
is defined by



has



is of



API specificaties

Objecttype aanmaken

1. Maak objecttype aan

```
POST /objecttype  
{ meta data }
```

2. Maak een initiële versie met het schema aan

```
POST /objecttype/a63f81/versions  
{ JSON schema }
```

3. Publiceer deze versie van het objecttype

```
PATCH /objecttype/a63f81/versions/1  
{ "status": "published" }
```

url	string <uri> (Url) [1 .. 1000] characters URL-referentie naar dit object. Dit is de unieke identificatie en locatie van dit object.
name required	string (Naam) [1 .. 100] characters Name of the object type
namePlural required	string (Nameplural) [1 .. 100] characters Plural name of the object type
description	string (Description) <= 1000 characters The description of the object type
dataClassification	string (Dataclassification) Enum: "open" "intern" "confidential" "strictly_confidential" Confidential level of the object type
maintainerOrganization	string (Maintainerorganization) <= 200 characters Organization which is responsible for the object type
maintainerDepartment	string (Maintainerdepartment) <= 200 characters Business department which is responsible for the object type
contactPerson	string (Contactperson) <= 200 characters Name of the person in the organization who can provide information about the object type
contactEmail	string (Contactemail) <= 200 characters Email of the person in the organization who can provide information about the object type
source	string (Source) <= 200 characters Name of the system from which the object type originates
updateFrequency	string (Updatefrequency) Enum: "real_time" "hourly" "daily" "weekly" "monthly" "yearly" "unknown" Indicates how often the object type is updated
providerOrganization	string (Providerorganization) <= 200 characters Organization which is responsible for publication of the object type
documentationUrl	string <uri> (Documentationurl) <= 200 characters Link to the documentation for the object type
labels	object (Labels) Key-value pairs of keywords related for the object type
createdAt	string <date> (Createdat) Date when the object type was created
modifiedAt	string <date> (Modifiedat) Last date when the object type was modified
versions	Array of strings <uri>

GET /api/v1/objecttypes/a63f81

```
{
  "url": "http://example/api/v1/objecttypes/a63f81",
  "name": "Boom",
  "description": "Register van bomen in de publieke ruimte.",
  "dataClassification": "open",
  "maintainerOrganization": "Gemeente Delft",
  "contactPerson": "Jan Eik",
  ...
  "versions": [
    "http://example/api/v1/objecttypes/a63f81/versions/1",
    "http://example/api/v1/objecttypes/a63f81/versions/2"
  ]
}
```

url	string <uri> (Url)
version	integer (Versie) Integer version of the OBJECTTYPE
objectType	string <uri> (Objecttype)
status	string (Status) Enum: "published" "draft" "deprecated" Status of the object type version
jsonSchema	object (JSON schema) JSON schema for Object validation
createdAt	string <date> (Createdat) Date when the version was created
modifiedAt	string <date> (Modifiedat) Last date when the version was modified
publishedAt	string <date> (Published_at) Date when the version was published

```

{
  "url": "http://example/api/v1/objecttypes/a63f81/versions/1",
  "version": 1,
  "objectType": "http://example/api/v1/objecttypes/a63f81",
  "status": "published",
  "jsonSchema": {
    "type": "object",
    "title": "Tree",
    "$schema": "http://json-schema.org/draft-07/schema#",
    "required": [
      "diameter"
    ],
    "properties": {
      "diameter": {
        ...
        "type": "integer",
        "description": "Diameter gemeten op 0,3m hoogte (cm).",
      },
      "plantDatum": {
        "type": "string",
        "format": "date",
        "description": "Datum waarop de boom is gepland."
      }
    }
  },
  "createdAt": "2020-11-14",
  "modifiedAt": "2020-11-16",
  "publishedAt": "2020-11-16"
}

```

url	string <uri> (Url)
version	integer (Versie) Integer version of the OBJECTTYPE
objectType	string <uri> (Objecttype)
status	string (Status) Enum: <code>"published"</code> <code>"draft"</code> <code>"deprecated"</code> Status of the object type version
jsonSchema	object (JSON schema) JSON schema for Object validation
createdAt	string <date> (Createdat) Date when the version was created
modifiedAt	string <date> (Modifiedat) Last date when the version was modified
publishedAt	string <date> (Published_at) Date when the version was published

```

{
  "url": "http://example/api/v1/objecttypes/a63f81/versions/1",
  "version": 1,
  "objectType": "http://example/api/v1/objecttypes/a63f81",
  "status": "published",
  "jsonSchema": {
    "type": "object",
    "title": "Tree",
    "$schema": "http://json-schema.org/draft-07/schema#",
    "required": [
      "diameter"
    ],
    "properties": {
      "diameter": {
        "type": "integer",
        "description": "Diameter gemeten op 0,3m hoogte (cm).",
      },
      "plantDatum": {
        "type": "string",
        "format": "date",
        "description": "Datum waarop de boom is gepland."
      }
    }
  },
  "createdAt": "2020-11-14",
  "modifiedAt": "2020-11-16",
  "publishedAt": "2020-11-16"
}

```

Objecttypen API features

- Objecttypen
 - Lezen en aanmaken
 - Bewerken en verwijderen zolang er geen gepubliceerde versies zijn.
- Versies van objecttypen
 - Lezen
 - Aanmaken, met schema validatie
 - Bewerken en verwijderen zolang deze status “draft” hebben.
- Versies publiceren en depreciëren
- Autorisatie middels API token (alle rechten)

Object aanmaken

1. Maak object aan

```
POST /object  
{ record data }
```

url	string <uri> (Url) [1 .. 1000] characters URL-referentie naar dit object. Dit is de unieke identificatie en locatie van dit object.
type required	string <uri> (Type) [1 .. 200] characters Url reference to OBJECTTYPE in Objecttypes API
record required	object (Record)

uuid	string <uuid> (Uuid) Unique identifier (UUID4)
typeVersion required	integer (Typeversion) [0 .. 32767] Version of the OBJECTTYPE for data in the object record
data	object (Data) Object data, based on OBJECTTYPE
geometry >	object (GeoJSONGeometry)
startDate required	string <date> (Startdate) Legal start date of the object record
endDate	string <date> (Enddate) Legal end date of the object record
registrationDate	string <date> (Registrationdate) The date when the record was registered in the system
correctionFor	string <uuid> (Correctionfor)
correctedBy	string (Correctedby)

GET /api/v1/objects/47454f

```
{
  "url": "http://example/api/v1/objects/47454f",
  "type": "http://example/api/v1/objecttypes/a63f81",
  "record": {
    "index": 6,
    "typeVersion": 1,
    "data": {
      "diameter": 26,
      "plantDate": "2000-11-01"
    },
    "geometry": {
      "type": "Point",
      "coordinates": [
        4.896787,
        52.37359492959213
      ]
    },
    "startDate": "2016-10-24",
    "endDate": null,
    "registrationDate": "2016-10-29",
    "correctionFor": 5,
    "correctedBy": null
  }
}
```

url	string <uri> (Url) [1 .. 1000] characters URL-referentie naar dit object. Dit is de unieke identificatie en locatie van dit object.
type required	string <uri> (Type) [1 .. 200] characters Url reference to OBJECTTYPE in Objecttypes API
record required	object (Record)

uuid	string <uuid> (Uuid) Unique identifier (UUID4)
typeVersion required	integer (Typeversion) [0 .. 32767] Version of the OBJECTTYPE for data in the object record
data	object (Data) Object data, based on OBJECTTYPE
geometry >	object (GeoJSONGeometry)
startDate required	string <date> (Startdate) Legal start date of the object record
endDate	string <date> (Enddate) Legal end date of the object record
registrationDate	string <date> (Registrationdate) The date when the record was registered in the system
correctionFor	string <uuid> (Correctionfor)
correctedBy	string (Correctedby)

GET /api/v1/objects/47454f

```
{
  "url": "http://example/api/v1/objects/47454f",
  "type": "http://example/api/v1/objecttypes/a63f81",
  "record": {
    "index": 6,
    "typeVersion": 1,
    "data": {
      "diameter": 26,
      "plantDate": "2000-11-01"
    },
    "geometry": {
      "type": "Point",
      "coordinates": [
        4.896787,
        52.37359492959213
      ]
    },
    "startDate": "2016-10-24",
    "endDate": null,
    "registrationDate": "2016-10-29",
    "correctionFor": 5,
    "correctedBy": null
  }
}
```

url	string <uri> (Url) [1 .. 1000] characters URL-referentie naar dit object. Dit is de unieke identificatie en locatie van dit object.
type required	string <uri> (Type) [1 .. 200] characters Url reference to OBJECTTYPE in Objecttypes API
record required	object (Record)

uuid	string <uuid> (Uuid) Unique identifier (UUID4)
typeVersion required	integer (Typeversion) [0 .. 32767] Version of the OBJECTTYPE for data in the object record
data	object (Data) Object data, based on OBJECTTYPE
geometry >	object (GeoJSONGeometry)
startDate required	string <date> (Startdate) Legal start date of the object record
endDate	string <date> (Enddate) Legal end date of the object record
registrationDate	string <date> (Registrationdate) The date when the record was registered in the system
correctionFor	string <uuid> (Correctionfor)
correctedBy	string (Correctedby)

GET /api/v1/objects/47454f

```
{
  "url": "http://example/api/v1/objects/47454f",
  "type": "http://example/api/v1/objecttypes/a63f81",
  "record": {
    "index": 6,
    "typeVersion": 1,
    "data": {
      "diameter": 26,
      "plantDate": "2000-11-01"
    },
    "geometry": {
      "type": "Point",
      "coordinates": [
        4.896787,
        52.37359492959213
      ]
    },
    "startDate": "2016-10-24",
    "endDate": null,
    "registrationDate": "2016-10-29",
    "correctionFor": 5,
    "correctedBy": null
  }
}
```

Objecten API features

- Objecten
 - Lezen, aanmaken, bewerken en verwijderen
 - Aankmaken en bewerken valideren de data a.d.h.v. gekoppelde objecttype
- Historie
 - Word automatisch bijgehouden
 - Alle records inzien van een enkel object
 - `GET /api/v1/objects/47454f/history`
- Zoeken en filteren
 - Op type en elk willekeurig data-attribuut
 - Op geo-coördinaten of polygonen
- Autorisatie middels API token
 - lees- en/of schrijfrechten op specifieke objecttypen.

Demo

objecten
bomen meldingen



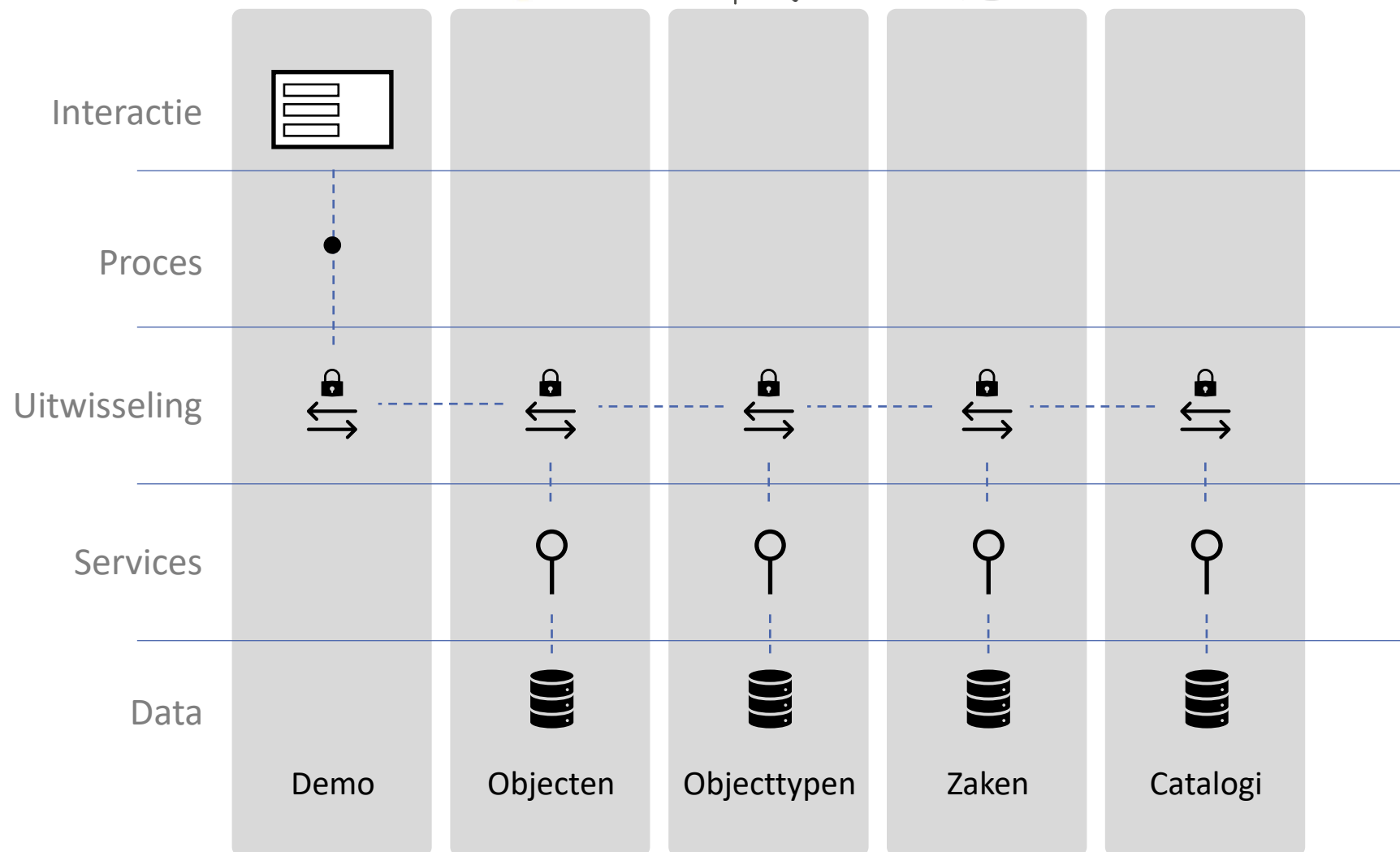
objecttypen
boom melding



zaak
mor



zaaktype
melding openbare ruimte (mor)





<https://github.com/maykinmedia/objects-api>



<https://hub.docker.com/r/maykinmedia/objects-api>



<https://objects-and-objecttypes-api.readthedocs.io/>

RTD



<https://commonground.nl/groups/view/54477963/objecten-en-objecttypen-api>

Einde