

NVDB REST API v2

Først

- v1 eller *beta-APIet* ble bygd for å understøtte [vegkart](#)
- Søkeobjektet er ikke lagd for API-brukere
- Defaultresponsene returnerer for mye data
- v2 er ment som et kjerne-API:
 - XML eller JSON
 - Ingen GeoJSON/GML-transformasjon slik v1 hadde
 - Ingen CVS/XLS-eksport slik v1 hadde
 - Konsistent “pagination” på endepunktene
- Transformasjoner blir gjort i *Eksporttjenesten*(™)
- Vi *kaller* v2 et REST-API men det oppfyller ikke alle [HATEOAS](#)-prinsippene

Enkeltopjektoppslag

— — —

- Filter-parametere innskrenker søket. F.eks. **?fylke=16**
- [?inkluder=\[metadata|egenskaper|lokasjon|\(...\)\]](#) begrenser hvilke *informasjonsdeler* av objektene man ønsker
- [?dybde=\[1|2|3|\(...\)|full\]](#) gir deg objekter med døtre
- [?geometritoleranse=\[10|20|30\]](#) lar deg forenkle geometriene
- [?srid=\[4326|32633\]](#) transformerer geometri
- Snarvei: [/v2/vegobjekt?id=85339421](#) videresender *ikke* parametere

Vegobjektsøk

— — —

- Inget eksplisitt søkeendepunkt
- Samme parametere som for enkelttoppslag med unntak av *dybde*
- Eksempel: www.vegvesen.no/nvdb/api/v2/vegobjekter/581

Vegobjektsøk, egenskapsfilter

— — —

- Egenskapstypenavn støttes *ikke*
- **?egenskap=**<egenskapstype><operator><verdi>
- Verdioperatorer: = | != | < | > | <= | >=
- Kombinasjonsoperatorer: AND | OR
- Nullstøtte for å spesifiser “mangler verdi”
- Wildcard-støtte: tekst og datoer
- Eksempel: [\(...\)/v2/vegobjekter/45?egenskap="1820=20 OR 1820=50"](http://(...)/v2/vegobjekter/45?egenskap=1820=20 OR 1820=50)

Vegobjektsøk, avanserte spørringer

— — —

- Overlapp: Alle **trafikkulykker** på strekninger med **fartsgrense** lik 80 km/t:
[\(...\)/571?overlapp="105\(egenskap\(2021\)=2738\)"](#)
- Relasjon: **Tunneller** med **tunnelløp** med lengde over 2 km.
[\(...\)/581?egenskap="relasjon\(67,egenskap\(1317\)>2000\)"](#)
- Egenskapsfilteret ved overlapp- og relasjonsspørring følger **samme egenskapsfiltersyntaks**
- Relasjonsspørringer støtter også *nøstede* spørringer

Vegobjektsøk, segmentering

— — —

- Vegobjektene er segmentert på følgende områder
 - Fylke
 - Kommune
 - Region
 - Vegavdeling
 - Vegreferanse
- Aktiveres med `?segmentering=[true|false]`
- Eksempel:
[\(...\)/v2/vegobjekter/67?segmentering=true&kommune=1663]((...)/v2/vegobjekter/67?segmentering=true&kommune=1663)
- Fremtid: segmentering på *kontraktsområder* og *riksvegruter*

Vegobjektsøk, statistikk

— — —

- Samme argumenter som for søket
- Eksempel:

www.vegvesen.no/nvdb/api/v2/vegobjekter/581/statistikk

Vegobjektsøk, endringer

— — —

- **?type**=[endret,slettet]
- *Endret* gir en liste av nye/endrede objekter
- *Slettet* gir en liste av *NVDB-slettinger*
- **?etter** lar deg velge tidsperiode for søket
- Fremtid: **?type=satt_historisk** for å finne objekter som er satt historisk uten at ny versjon er opprettet

Vegnettsøk

— — —

- Kun XML/JSON som resten av APIet
- Projisert vegreferanse på veglenkenivå
- **?srid** for å gjøre geometritransformering
- Mer avansert filtrering:
 - Områdefiltre (fylke, kommune, etc.)
 - vegreferanse
- Fremtid:
 - GML/GeoJSON i *Eksporttjenesten*
 - *Kontraktsområde-* og *riksvegrutefiltre*
 - Statistikk slik som på objektsøket
- Eksempel:

[\(...\)/v2/vegnett/lenker?antall=5&vegreferanse=?v](http://(...)/v2/vegnett/lenker?antall=5&vegreferanse=?v)

Stedfestingssøk, posisjon

— — —

- [/v2/posisjon](#) med følgende parametere
 - **?nord/ost** eller **?lat/lon**
 - **?maks_avstand**
 - **?maks_antall** (for å begrense antall treff)
 - **?konnekteringslenker**=true|false
 - **?detaljerte_lenker**=true|false
 - **?vegreferanse** (begrenset til hp)
 - **?srid** (for geometritransformering)

Stedfestingssøk, veglenke/vegref

— — —

- **/v2/veg**
 - [?vegreferanse=1600Ev6hp12m1000](#)
 - [?veglenke=0.21279373@72811](#)
 - **?srid** (for geometritransforming)

Annet

— — —

- Datakatalogen er selvsagt fremdeles med:
[\(...\)/v2/vegobjekttyper](#)
 - Flere nye informasjonsfelter er tatt med
 - ?**inkluder**=[egenskapstyper|relasjonstyper|styringsparametere]
 - Fremtid: objektkategori
- Nytt: **Enhetsliste** [\(...\)/v2/vegobjekttyper/enheter](#)
- Nytt: **Datatyper** [\(...\)/v2/vegobjekttyper/datatyper](#)

Eksporttjenesten

— — —

- Ment å inneholde datatransformasjoner som tidligere var del av APIet
- *Ikke* responsversjonert; vil alltid gi nyeste versjon
- Første utgave har kun CSV-eksport for vegobjekter.
- Parametere? Følger kjerne-APIet (v2)
- Eksempel:
www.vegvesen.no/vegkart/eksport/vegobjekter/67?kommune=1601&inkluder=egenskaper
- Fremtid:
 - GML/GeoJSON for vegnett
 - GML/GeoJSON for vegobjekter

API-evolusjon

— — —

- 2 dimensjoner: *API-versjon* og *respons-revisjon*
- Egen mime type:
`application/vnd.vegvesen.nvdb-v2-rev0+json`
- Endringer:
 - Additive endringer i endepunktene (dvs. nye endepunkter og nye parametere) eller nye felter i respons krever ingen endring av klienter.
 - Destruktive endringer vil gi ny versjon og/eller revisjon
- Eksempel: POINT **Z**(297359.859375 6729450.60351563 35.34)

Fremtid

— — —

- v2 er designet slik at historikkfunksjonalitet faller naturlig inn uten å måtte lage v3 (håper vi). Utviklingen at dette ligger dessverre noe lenger frem i tid, ironisk nok.
- Vegkart går fremdeles mot v1. “Oversetting” til v2 har kommet langt.

Open Source API-klient for Java8

— — —

- Lisens: BSD 2-clause
- Fiks ferdig domenemodell og parsere for alle endepunktene
- Publisert til jCenter
- URL: <https://github.com/nvdb-vegdata/nvdb-api-client>
- GAV:

```
<dependency>  
  <groupId>no.vegvesen.nvdb</groupId>  
  <artifactId>nvdb-api-client</artifactId>  
  <version>1.0.0</version>  
</dependency>
```

API-klient, eksempler

— — —

```
ClientFactory factory = new ClientFactory("https://www.vegvesen.no/nvdb/api/v2", "nvdbapi-client", "ACME");
RoadObjectClient client = factory.createRoadObjectClient();

// Enkeltopjektoppslag
RoadObject ro = client.getRoadObject(534, 1);

// Søk
RoadObjectRequest req = RoadObjectRequest.newBuilder()
    .withIncludes(RoadObjectClient.Include.ATTRIBUTES, RoadObjectClient.Include.LOCATION)
    .withCounty(16)
    .withPage(Page.count(3500))
    .build();

RoadObjectClient.RoadObjectsResult iterator = client.getRoadObjects(581, req);
while (iterator.hasNext()) {
    List<RoadObject> page = iterator.next();
    // Do stuff
}
```

Nyttige lenker

— — —

- API v2: <https://www.vegvesen.no/nvdb/apidokumentasjon>
- API Client:
<https://github.com/nvdb-vegdata/nvdb-api-client>
- Vegkart: <http://www.vegkart.no>

Spørsmål?