



GraphQL tjenester

Modernisering af Datafordeleren

Temamøde den 23. oktober 2024





OBS: Mødet bliver optaget





Velkomst

Stine Kern Licht

- Projektleder for KDS
- stkel@kds.dk



Klimadatastyrelsen

August Clement Leve

- Domæne lead for Netcompany
- aucl@netcompany.com

Netcompany



Agenda

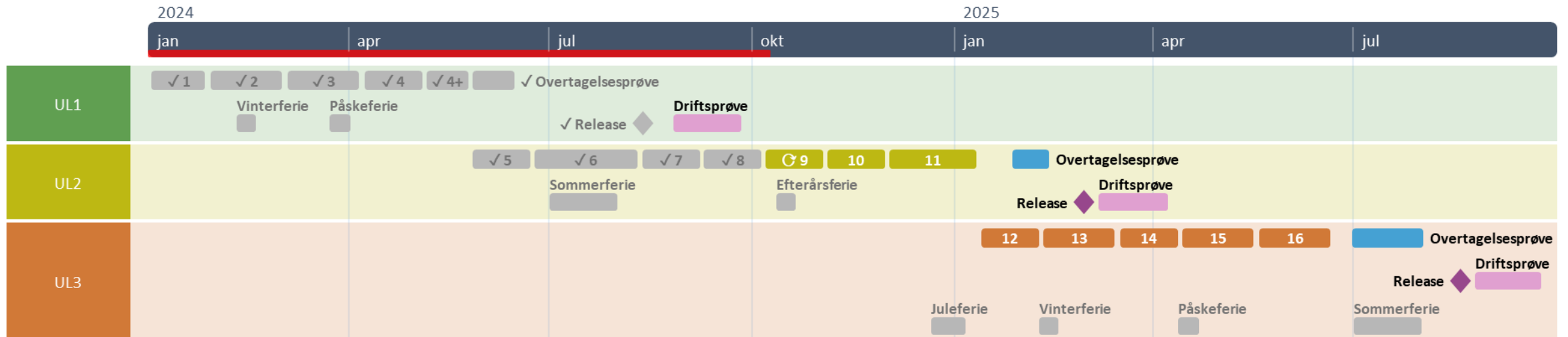
- Velkomst
- Kort om udviklingsleverance 2
- Forretningsmæssige behov
- Præsentation af GraphQL tjenester
 - Bitemporalitet
 - Filtrering og paginering
- Overgang til den nye funktionalitet
- Videre forløb
- Spørgsmål

Praktisk information:

- Stil gerne spørgsmål undervejs (virtuelle deltagere: skriv spørgsmål i chatten)
- Slides udsendes efterfølgende
- Video gøres tilgængelig i en begrænset periode

Kort om udviklingsleverance 2

Tidsplan for udviklingsleverancer



Udviklingsleverance 1

- 4 sprints
- Start: Januar 2024
- Release: Juni 2024
- Funktionelle fokusdomæner:
 - Fildownloads
 - Geodatatjenester

Udviklingsleverance 2

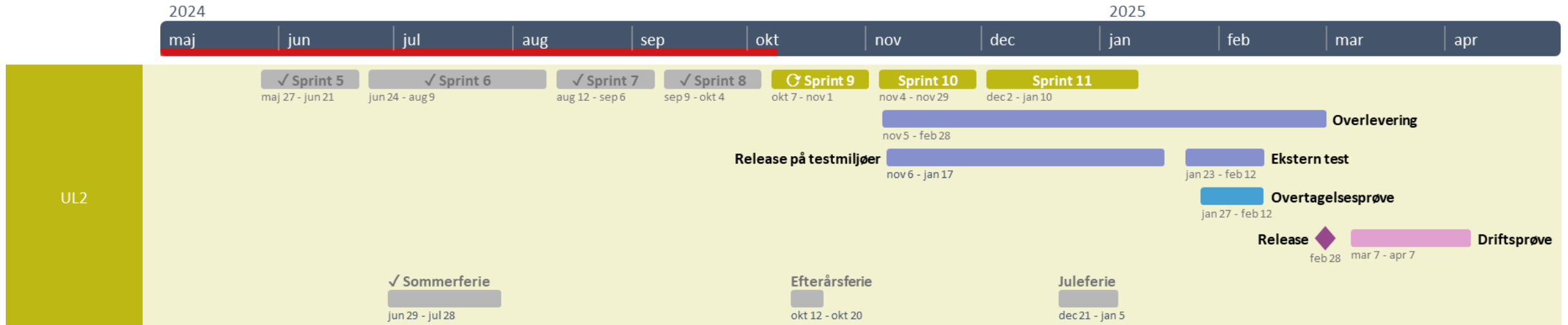
- 7 sprints
- Start: Maj 2024
- Release: Februar 2025
- Funktionelle fokusdomæner:
 - GraphQL-tjenester
 - Hændelser
 - Selvbetjening

Udviklingsleverance 3

- 5 sprints
- Start: Januar 2025
- Release: August 2025
- Funktionelle fokusdomæner:
 - GraphQL-tjenester
 - Geodatatjenester

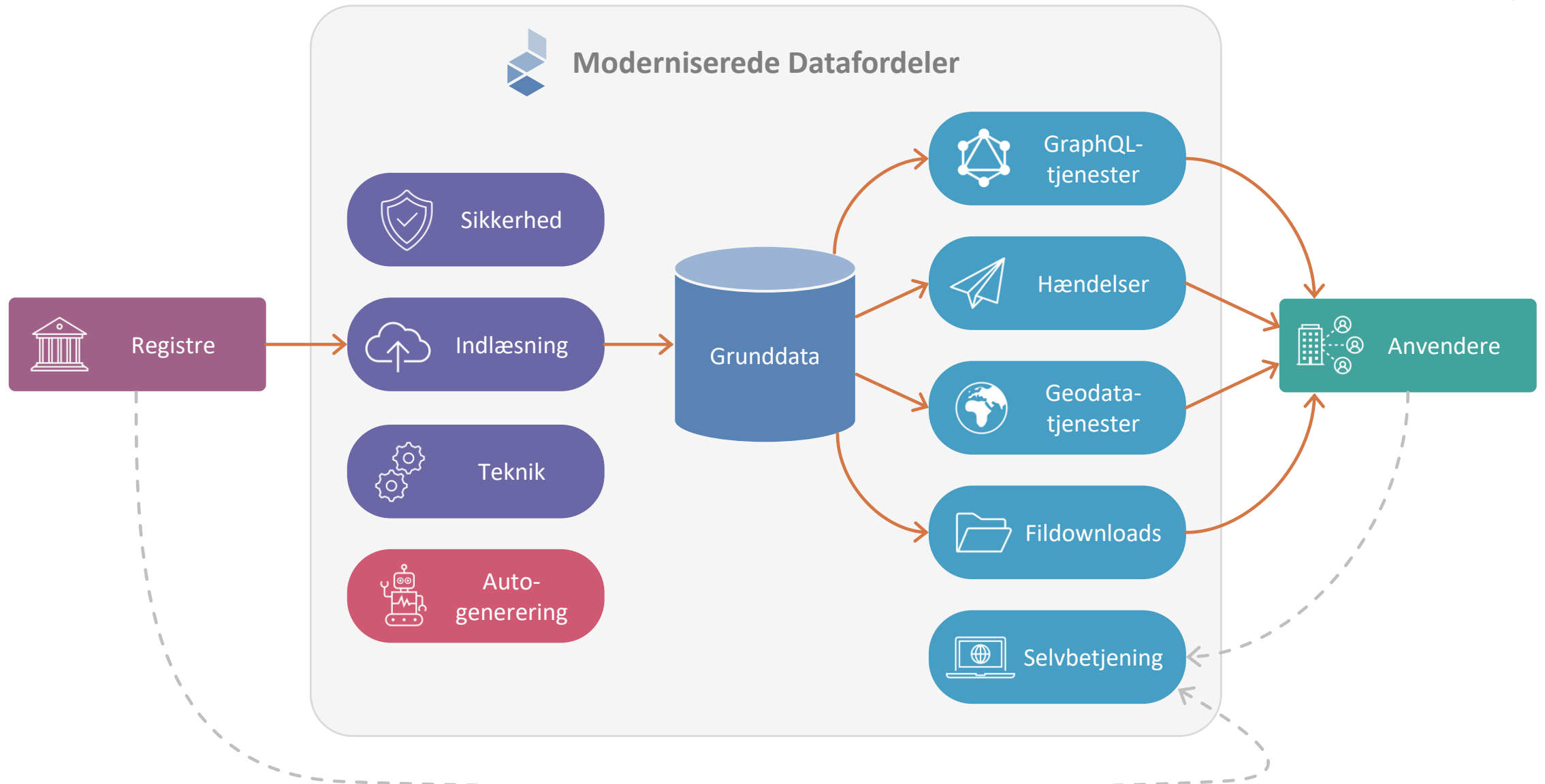


Tidsplan for udviklingsleverance 2

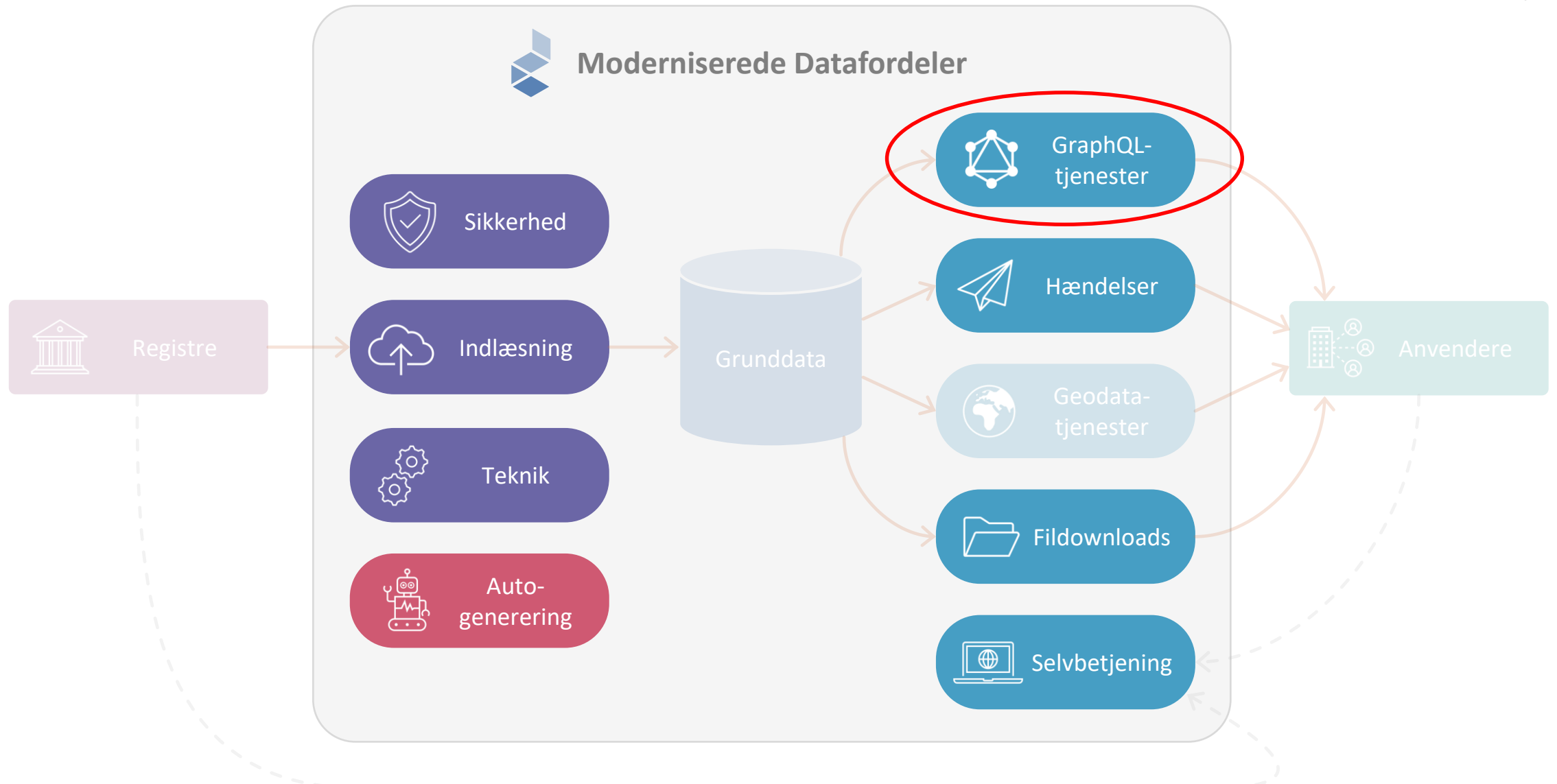


- Sprint 5 startede op den 27. maj.
- Overlevering til forvaltning i januar og februar med udrulning til testmiljøer – med ekstern test og overtagelsesprøve.
- **Release til produktion tentativt planlagt til den 28. februar** – med efterfølgende driftsprøve.

Funktionelle og tekniske domæner



Funktionelle og tekniske domæner for UL2



Forretningsmæssige behov



DATAFORDELER



Forretningsmæssige behov

- ✗ Specialudviklet programmel
- ✗ Tungt og dyrt vedligehold af manuelt specificerede og kodede tjenester.
- ✗ Fast definerede tjenester uden fleksibilitet for anvenderne.
- ✗ Lang time-to-market.

- ✓ Standardprogrammel
- ✓ Lavere vedligeholdelsesomkostninger og automatisk genererede tjenester.
- ✓ Ensartede tjenester og fleksible søgemuligheder for anvenderne.
- ✓ Kortere time-to-market.



Værdier ved nye GraphQL-tjenester

Nuværende Datafordeler

- ÷ Nuværende REST-tjenester med forskellig logik og struktur fra register til register
- ÷ Komplekst for registre at specificere REST-tjenester via DLS – og høj time-to-market for implementering heraf
- ÷ Meget manuelt arbejde ifm. implementering af nye og ændrede REST-tjenester
- ÷ Mange data, der ellers er på Datafordeleren, er ikke udstillet via REST-tjenester
- ÷ Udstillingens output skemaer følger ikke altid grunddatamodellen

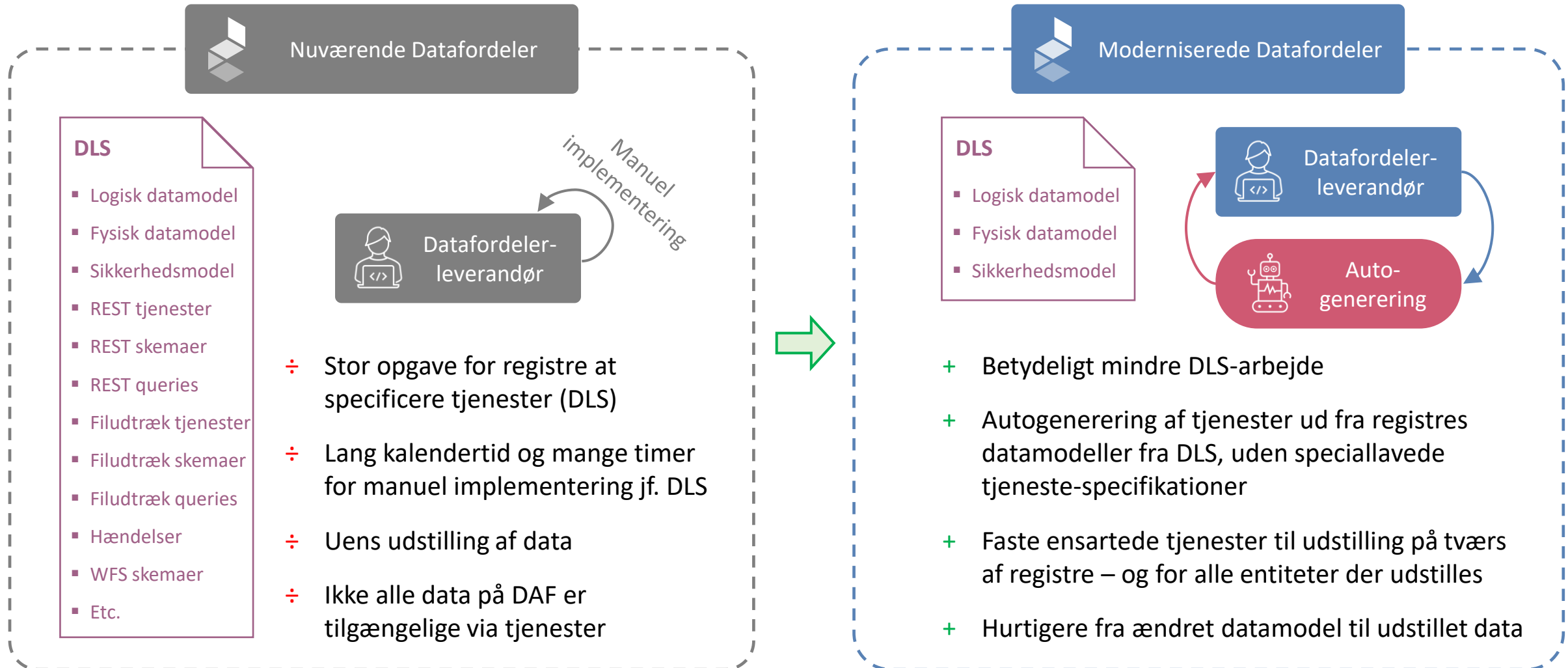


Moderniserede Datafordeler

- + GraphQL-udstilling for alle entiteter, som anvendere har adgang til
- + Samme logik og struktur for opslag af data, uanset register
- + Ingen register-specifikation (DLS) af udstillings-skemaer, tjenesteparametre eller tjenestelogik
- + Autogenerering af implementering af nye og ændrede GraphQL-tjenester, med minimalt manuelt arbejde
- + Anvendere kan nøjes med at hente de data der er brug for
- ! Anvendere skal (som hidtil) forholde til sammenhæng mellem data ud fra grunddatamodellen

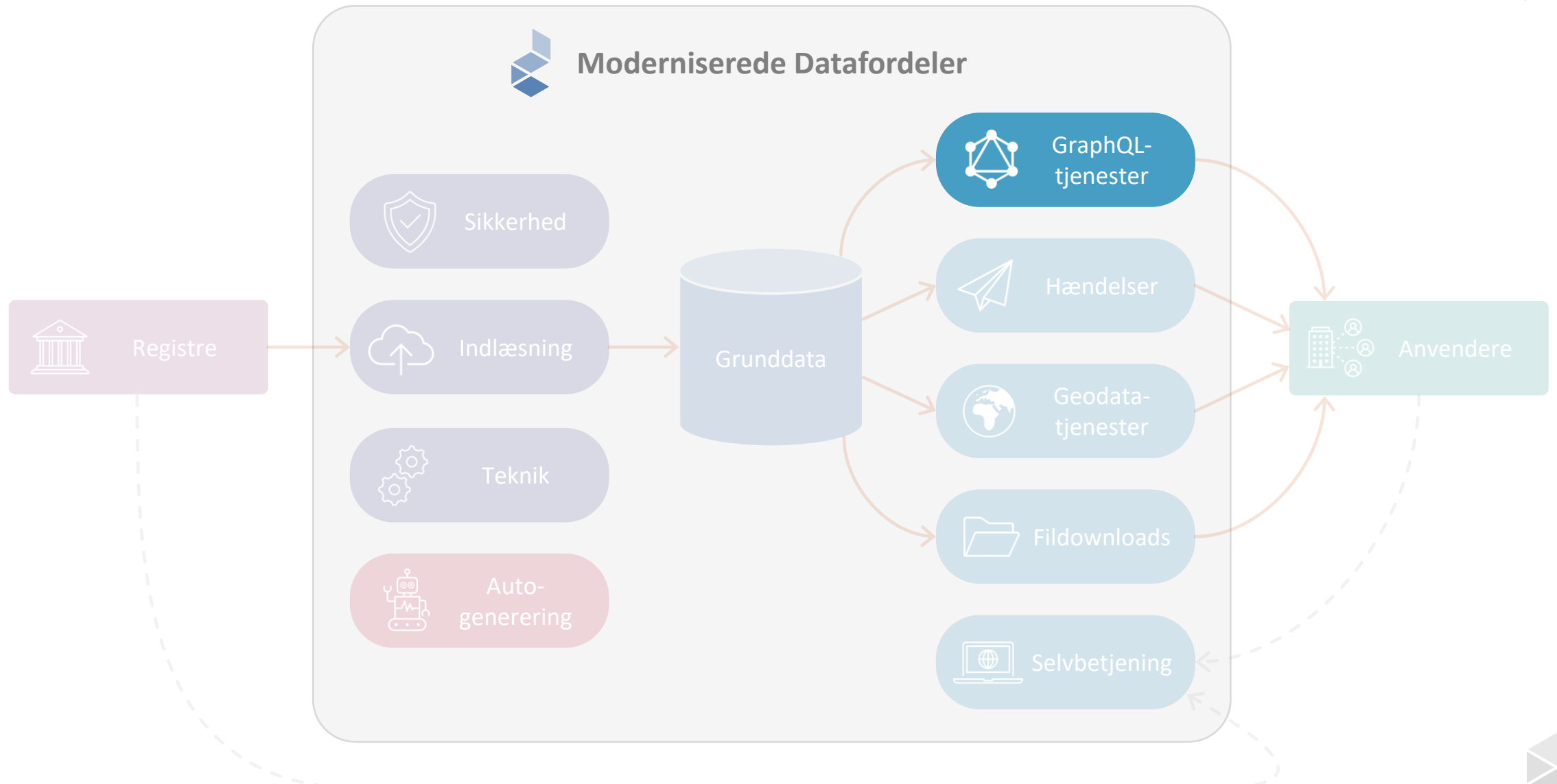


Værdi ved autogenerering af nye tjenester



Præsentation af GraphQL tjenester

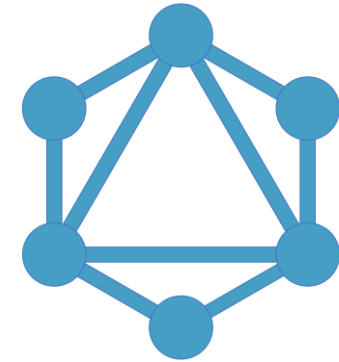
Funktionelle og tekniske domæner





GraphQL-tjenester

- Nuværende Datafordeler: **REST-tjenester**, der returnerer statiske data og er defineret af registrene i DLS'en i hånden.
- Moderniseret Datafordeler – UL2: **Entitetsbaserede GraphQL-tjenester**, som returnerer den data anvenderen selv angiver.






Fordele og ulemper ved GraphQL-tjenester kontra OData

OData

- + OData understøtter aritmetiske operatører og aggregering out-of-the-box.
- + OData genererer automatisk metadata til anvendelse af andre programmer.
- ÷ OData har tekniske begrænsninger på datamodeller som ikke altid kan overholdes ved registrene
- ÷ Yderligere specifikationer er nødvendige for filtrering og forespørgsler.
- ÷ Tilpasning til specifikke behov er vanskelig.
- ÷ Dokumentationen til rammeværktøjet er utilstrækkelig, og der mangler støtte til filtrering af indlejrede objekter.

GraphQL

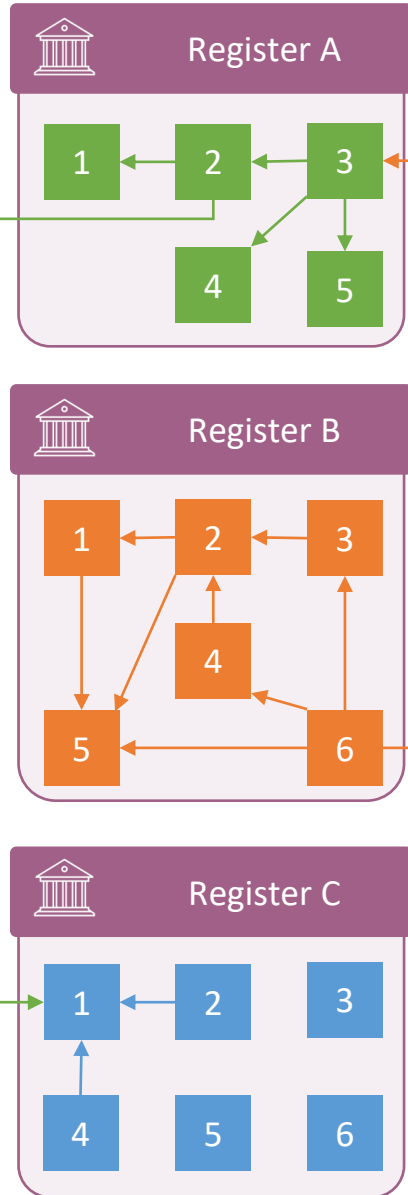
- + GraphQL genererer mere detaljeret dokumentation og er mere fleksibelt. Dette gør det nemmere for anvendere at komme i gang med at bruge tjenesterne.
 - + GraphQL giver granulær kontrol over filtrering og inkluderer metadata om sortering og filtrering.
 - + GraphQL rammeværket gør det nemmere at implementere speciallavede ting ift. Odata.
 - ÷ GraphQL mangler indbygget support til aritmetiske operatører og aggregeringer.
- 



GraphQL-tjenester

Entitetsbaserede GraphQL-tjenester

Register datamodeller



Entitetsbaserede GraphQL-tjenester



Funktionalitet i GraphQL-tjenesterne

- GraphQL
- Tjeneste pr. entitet
- Bitemporalitet + point-in-time
- Filtrering på attributter
- GraphQL operatører (uden joins)
- Geospatial operatører

Eksempel:

```

/BBR/v1?query=
{
  BBR_Bygning(
    where: {
      status: { eq: "6" }
      byg404koordinat: {
        within: {
          wkt: "POLYGON
            ((530500 6147880,
              530510 6147880,
              530510 6147800,
              530500 6147800,
              530500 6147880))"
          crs: 25832
        }
      }
    }
  )
}

```



GraphQL-tjenester

Entitetsbaserede GraphQL-tjenester

GraphQL query



<https://graphql.datafordeler.dk/DAR/v1?query=>



Query

Register

```

{
  DAR_Husnummer ← Entitet
  {
    id_lokalId
    virkningFra
    virkningTil
    husnummertekst
    postnummer
    kommuneinddeling
  }
}
  
```

Felter



Resultat

```

{
  "data": {
    "DAR_Husnummer": [
      {
        "id_lokalId": "2b190e33-7f85-3634-e044-0003ba298018",
        "virkningFra": "2018-05-03T16:58:34.000000Z",
        "virkningTil": null,
        "husnummertekst": "25",
        "postnummer": "d92cd14b-c832-45d7-a465-c93dfe1a3a85",
        "kommuneinddeling": "389142"
      }
    ],
    :
  }
}
  
```

Detaljer kan ændre sig

NOTE: Viste query er simplificeret

Visualisering af forskel mellem GraphQL- og REST-tjenester



Nuværende Datafordeler

```
https://services.datafordeler.dk/BBR/BBRPublic/1/REST/bygning?  
nord=6147800 & syd=6147880 & oest=530500 & vest=530510 & status=6
```

```
[  
  {  
    "datafordelerOpdateringstid": "2021-04-02T04:25:59.562061+02:00",  
    "byg007Bygningsnummer": 21,  
    "byg021BygningensAnvendelse": "421",  
    "byg0260Pførelsesår": 1960,  
    "byg032YdervæggensMateriale": "1",  
    "byg033Tagdækningsmateriale": "3",  
    "byg037KildeTilBygningensMaterialer": "1",  
    "byg038SamletBygningsareal": 65,  
    "byg040BygningensSamledeErhvervsAreal": 65,  
    "byg041BebyggetAreal": 65,  
    "byg053BygningsarealerKilde": "1",  
    "byg054AntalEtager": 1,  
    "byg404Koordinat": "POINT(530502.29 6147876.88)",  
    "husnummer": "0a3f5090-1981-32b8-e044-0003ba298018",  
    "id_lokalId": "0ea64718-1182-4a1a-be1a-4fbe293ca5ce",  
    "kommunekode": "0621",  
    ... }  
  ]
```

Moderniserede Datafordeler

```
https://graphql.datafordeler.dk/BBR/v1?query={BBR_Bygning(where: {  
  status: { eq: "6" }  
  byg404koordinat: { StWithin: "POLYGON((530500 6147880, 530510  
    6147880, 530510 6147800, 530500 6147800, 530500 6147880))"}  
}) {  
  byg007Bygningsnummer, byg0260Pførelsesår, byg404Koordinat,  
  husnummer, id_lokalId, kommunekode  
}}
```

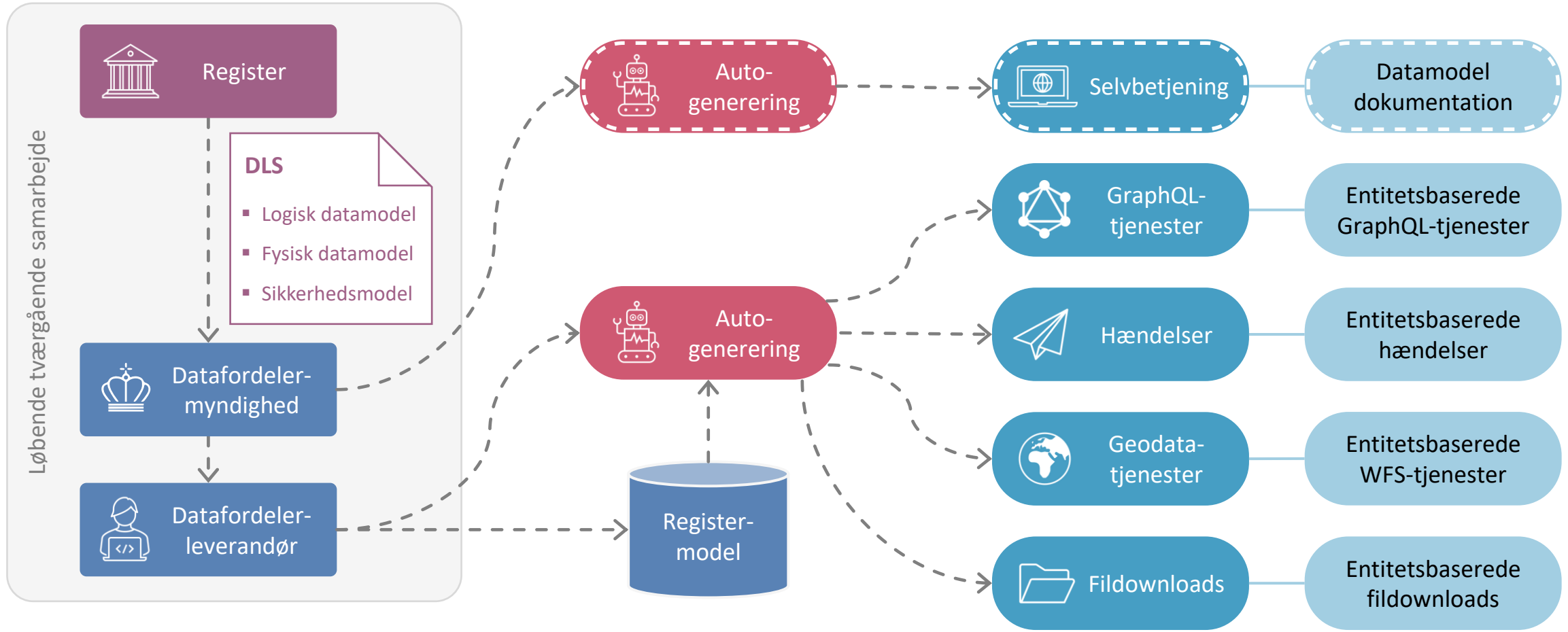
```
{ "data": {  
  "BBR_Bygning": {  
    {  
      "byg007Bygningsnummer": 21,  
      "byg0260Pførelsesår": 1960,  
      "byg404Koordinat": "POINT(530502.29 6147876.88)",  
      "husnummer": "0a3f5090-1981-32b8-e044-0003ba298018",  
      "id_lokalId": "0ea64718-1182-4a1a-be1a-4fbe293ca5ce",  
      "kommunekode": "0621"  
    }  
  }  
}}
```

Detaljer kan ændre sig

NOTE: Viste query er simplificeret

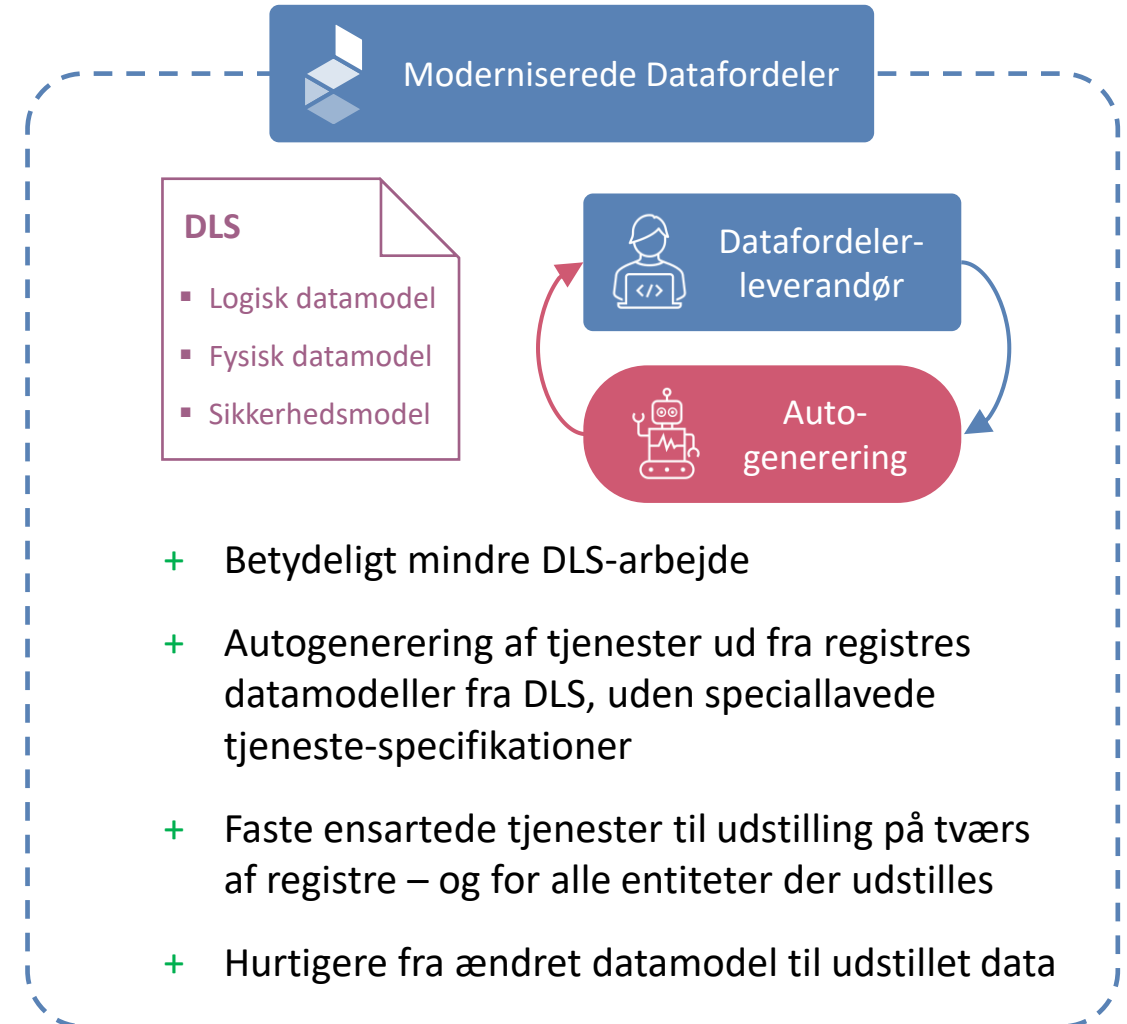
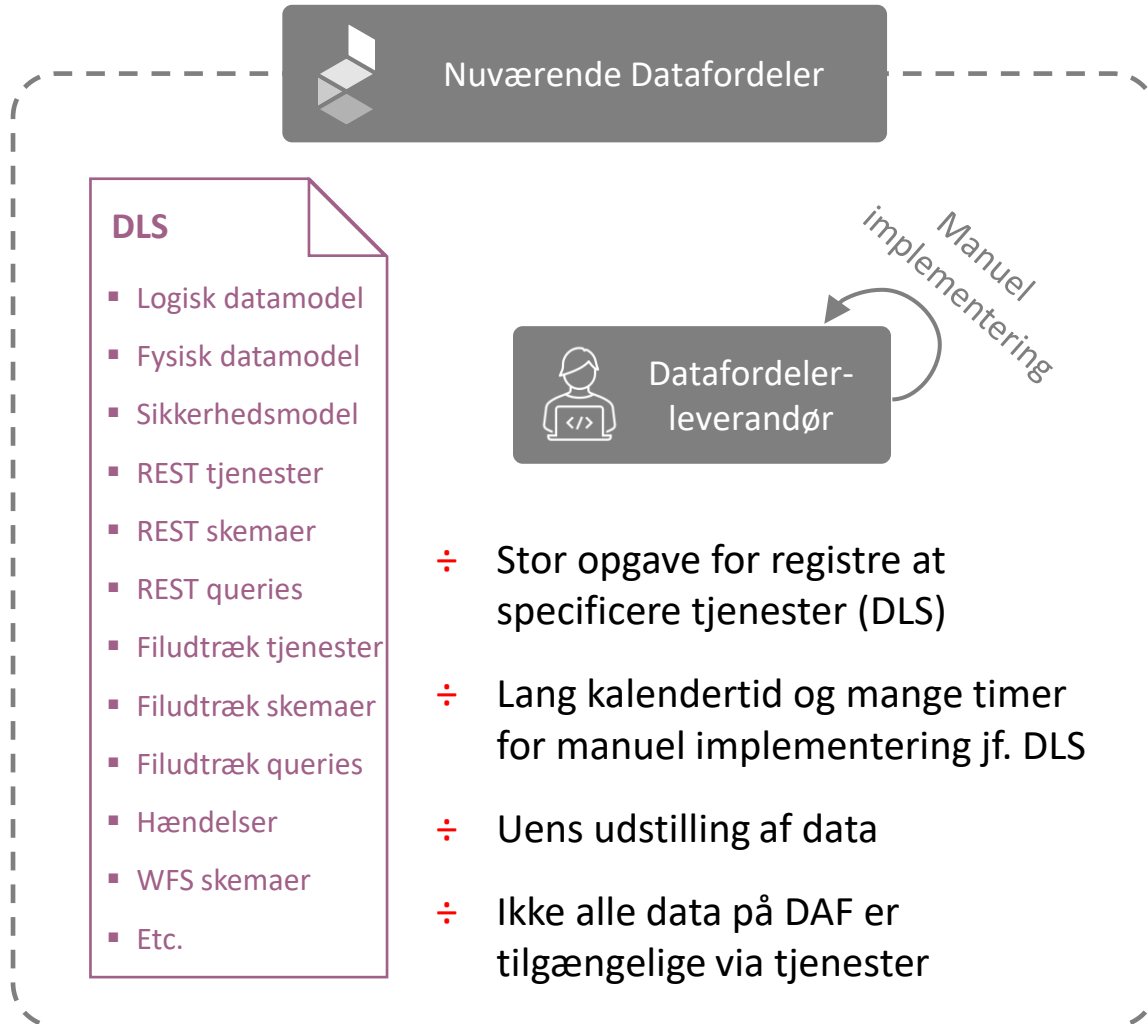


Udstilling af register metadata

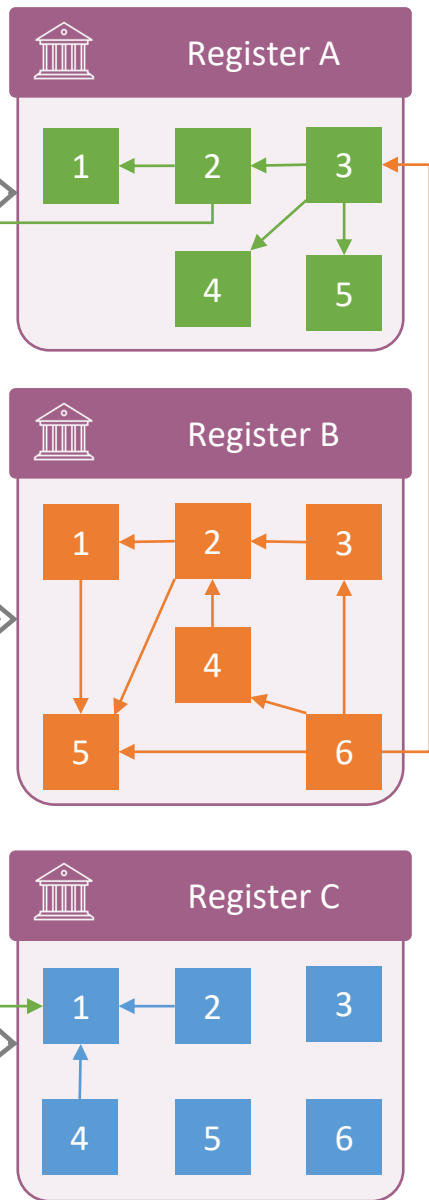




Gevinster ved autogenerering



Register datamodeller



Entitetsbaserede GraphQL-tjenester



Register metadata til GraphQL-tjenester

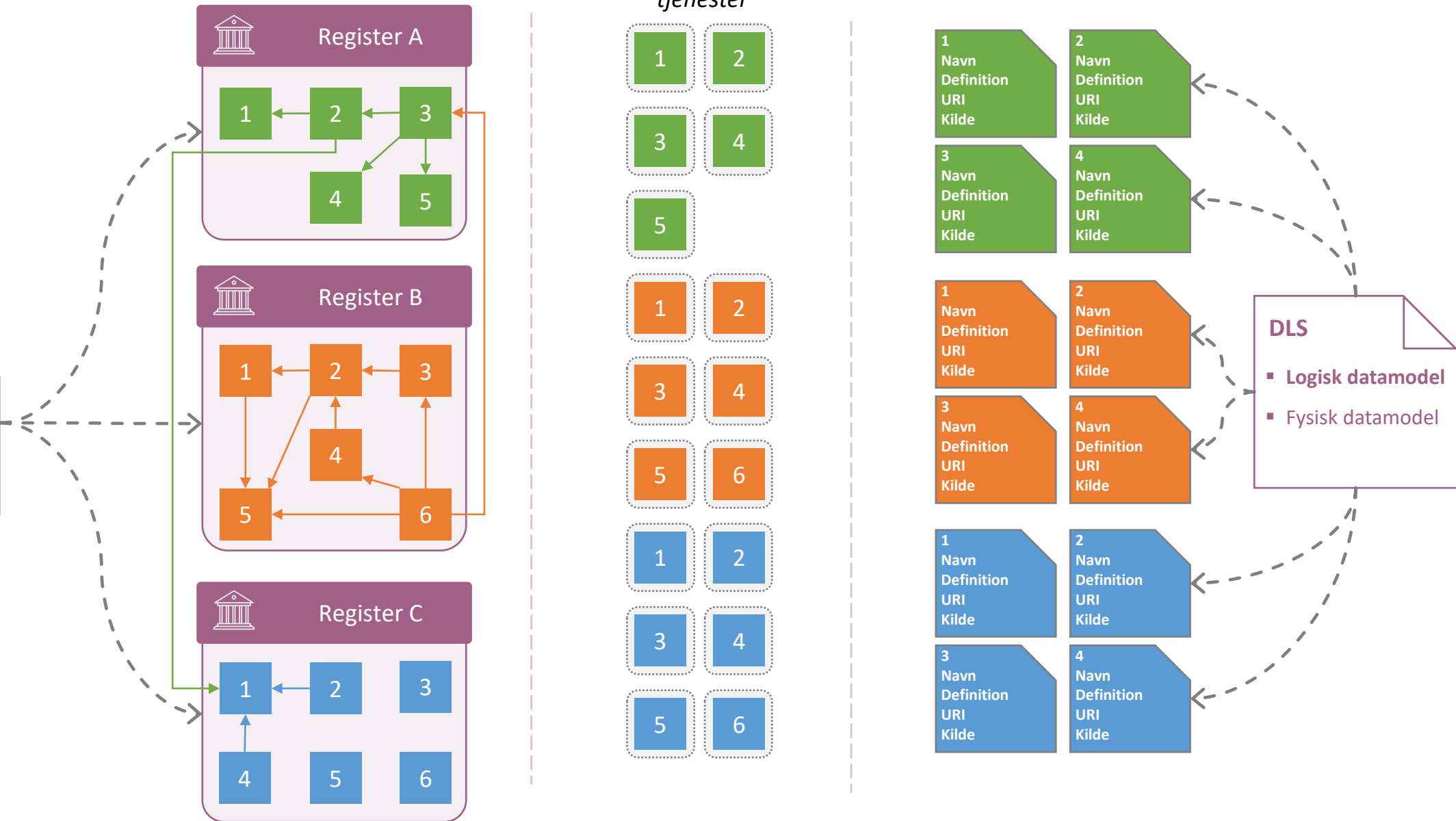


DLS

- Logisk datamodel
- Fysisk datamodel

DLS

- Logisk datamodel
- Fysisk datamodel





GraphQL-tjenester

Entitetsbaserede
GraphQL-tjenester

Udstilling af registermetadata



<https://graphql.datafordeler.dk/BBR/v1/schema?>

1
Navn
Definition
URI
Kilde

Navn:	byg007Bygningsnummer
URI:	https://data.gov.dk/model/profile/bbr/byg007Bygningsnummer
Foretrukken term:	Bygningsnummer
Definition:	angiver bygningens nummer indenfor ejendommen
Kommentar:	Løbenummer under ejendommen. Værdierne 1-9999 er tilladte
Juridisk kilde:	https://www.retsinformation.dk/eli/ta/2019/797
Kilde:	https://instruks.bbr.dk/instruks/0/30
Multiplicitet:	1
Type:	Integer
Navn:	byg021BygningensAnvendelse
URI:	https://data.gov.dk/model/profile/bbr/byg021BygningensAnvendelse
Foretrukken term:	Bygningens anvendelse
Definition:	angiver bygningens hovedanvendelse
Kommentar:	Bygningens hovedanvendelse. Har en bygning flere anvendelser registreres den anvendelse som hovedparten af bygningens areal har.

Detaljer kan
ændre sig



GraphQL-tjenester

Entitetsbaserede
GraphQL-tjenester

Overblik over registre



Registerdata

- Tabulære data
 - BBR
 - DAR
 - DAGI
 - GeoDK Vektor
 - DHM (Højdekurver & Oprindelse)
 - EBR
 - Fikspunkter
 - MAT
 - Danske Stednavne
 - Kortbladsinddelinger
 - CVR ←
 - VUR ←

Entiteten CVRPerson
bliver beskyttet og
resten af CVR's data
bliver udstillet
offentligt.



Beskyttet registerdata

- CPR (UL3)
- CVR (kun CVRPerson)
- SVR
- EJF
- VUR

1.Ejendomsvurdering.bebyggelseProcent,
2.Ejendomsvurdering.ejerlejlighedKode,
3.Ejendomsvurdering.grundværdiOmråde,
4.Ejendomsvurdering.meddelelsestypeKode,
5.Ejendomsvurdering.pg4KodeVurdering,
6.Ejendomsvurdering.pg4KodeVurderingOphøjet,
7.Ejendomsvurdering.klageType,
8.Ejendomsvurdering.omvurderingGrund1Kode,
9.Ejendomsvurdering.omvurderingGrund2Kode,
10.Loftansættelse.reguleringssatsB



GraphQL-tjenester

Adgangsbegrænsning



- VUR
 - 10 beskyttede felter udeladt fra snitfladen
 - Data udstilles uden adgangsbegrænsning
- CVR
 - Kun temporale opslag (registreringstid sættes til nu)
 - CVRPerson er adgangsbegrænset
- SVR & EJF
 - Ved manglende rettigheder følger tjenesterne GraphQL-over-HTTP specifikationen, og returnerer passende HTTP statuskode og fejlbesked.

Detaljer kan ændre sig

GraphQL-tjenester

Adgangsbegrænsning



Manglende adgang

```
https://graphql.datafordeler.dk/CVR/v1?query={CVR_CVRPerson
{
  id, cprperson, status, registreringfra, registreringtil,
  virkningfra, virkningtil
}
}
```

```
{
  "errors": [
    {
      "message": "The current user is not
authorized to access this resource.",
      "extensions": {
        "traceId": "00-
348583c7f621fbe0cf7289ef7ee4fa31-971cb56c47d658d0-
01",
        "code": "DAF-AUTH-0001"
      }
    }
  ],
  "data": null
}
```



Bruger med adgang

```
https://graphql.datafordeler.dk/CVR/v1?query={CVR_CVRPerson
{
  id, cprperson, status, registreringfra, registreringtil,
  virkningfra, virkningtil
}
}
```

```
{ "data": {
  "CVR_CVRPerson": {
    {
      "id": "0ea64718-1182-4a1a-be1a-
4fbe293ca5ce",
      "cprperson": 000000000,
      "status": "3",
      "registreringfra": "2018-05-
03T16:58:34.000000Z",
      "registreringtil": null,
      "virkningfra": "2018-05-
03T16:58:34.000000Z",
      "virkningtil": null
    }
  }
}
```

NOTE: Viste query er simplificeret

Detaljer kan ændre sig

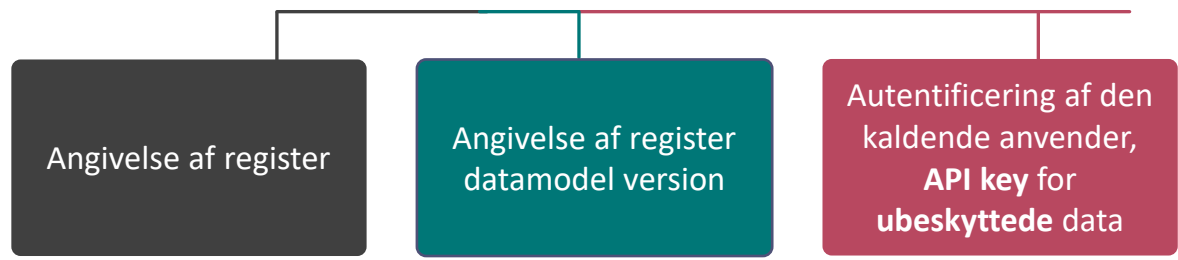



GraphQL-tjenester

Sådan tilgår du ubeskyttede data


<https://graphql.datafordeler.dk/DAR/v1?APIKey=eyJzdWliOilxM...>

Resultat: Registerdata
tilsvarende forespurgt data i
GraphQL-query i JSON-format



 **Query**

```
{DAR_Adresse(
  registreringstid: "2020-06-10T13:31:54.478174Z"
  virkningstid: "2020-06-10T13:31:54.478174Z"
  where: { id_lokalId: { eq: "5661ca25-f0f0-25c4-e044-0003ba298018" }}}
{
  id_lokalId
  registreringFra
  registreringTil
  virkningTil
  adressebetegnelse
  husnummer }}
```

 **Resultat**

```
[
  {
    "id_lokalId": "5661ca25-f0f0-25c4-e044-0003ba298018",
    "registreringFra": "2020-06-10T13:31:54.478174Z",
    "registreringTil": null,
    "virkningFra": "2020-06-10T13:31:54.478174Z",
    "virkningTil": null,
    "adressebetegnelse": "Sankt Kjelds Gade 1, 2100 København Ø",
    "husnummer": "0a3f507a-eeb8-32b8-e044-0003ba298018"
  }
]
```

NOTE: Viste query er simplificeret

Detaljer kan ændre sig



GraphQL-tjenester

Sådan tilgår du beskyttede data

GET <https://graphql.datafordeler.dk/EJF/v1?>

Params • Authorization Headers (7) Body Pre-request Script Tests Settings

Headers 6 hidden

	Key	Value
<input checked="" type="checkbox"/>	Authorization	bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

Resultat: Registerdata tilsvarende det forespurgte i GraphQL-query i JSON format

Query

```
{EJF_Ejerskab(
  registreringstid: "2020-06-10T13:31:54.478174Z"
  virkningstid: "2020-06-10T13:31:54.478174Z"
  where: { id_lokalId: { eq: "5661ca25-f0f0-25c4-e044-0003ba298018" } }
) {
  virkningFra
  virkningTil
  registreringFra
  registreringTil
  id_lokalId
  ejerforholdskode
  primaerKontakt }}
```

Resultat

```
[
  {
    "id_lokalId": "5661ca25-f0f0-25c4-e044-0003ba298018",
    "registreringFra": "2020-06-10T13:31:54.478174Z",
    "registreringTil": null,
    "virkningFra": "2020-06-10T13:31:54.478174Z",
    "virkningTil": null,
    "ejerforholdskode": "123130586",
    "primaerKontakt": "August Clement Leve"
  }
]
```

NOTE: Viste query er simplificeret



GraphQL-tjenester

Bitemporalitet

- Anvendere kan optionelt angive et registreringstidspunkt og/eller virkningstidspunkt i deres forespørgsel.
- Hvis tidspunkt er angivet, filtrerer backenden med registerets bitemporale regler.



GraphQL-skema

```
"Query type for DAR - Danmarks Adresseregister"
type Query{
  DAR_Husnummer(
    registreringstid: DafDateTime
    virkningstid: DafDateTime
    first: Int "Returns the first _n_ elements from the list."
    after: String
    "Returns the elements in the list that come after the specified cursor"
    where: DAR_HusnummerFilterInput): DAR_HusnummerConnection
}
```



Query

```
query DAREksempel {
  DAR_Husnummer(
    registreringstid: "2024-09-06T10:00:00.123456Z"
    virkningstid: "2024-09-06T10:00:00.123456Z"
  ) {
    id_lokalId
    registreringFra
    registreringTil
    virkningFra
    virkningTil
  }
}
```

Detaljer kan ændre sig

GraphQL-tjenester

Filtrering og paginering



- Anvendere forventes at kunne filtrere på det der understøttes I dag
- Anvendere kan også paginere på data



Query

```
query {
  DAR_Husnummer(
    first: 3,
    after: "AAAAAAAAAAAA=",
    registreringstid: "2024-01-01T01:01:01.123456Z",
    virkningstid: "2023-01-02T03:04:05.678901+01:00"
    where: {
      id_lokalId: {
        eq: "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa"
      }
    }
  ){
    pageInfo {
      endCursor
      hasNextPage
      hasPreviousPage
      startCursor
    }
    edges {
      cursor
      node {
        id_lokalId
      }
    }
  }
}
```



Resultat

```
{
  "data": {
    "DAR_Husnummer": {
      "pageInfo": {
        "endCursor": "KQAAAAAAAAAA=",
        "hasNextPage": true,
        "hasPreviousPage": false,
        "startCursor": "AAAAAAAAAAAA="
      },
      "edges": [
        {
          "cursor": "AAAAAAAAAAAA=",
          "node": {
            "id_lokalId": "aaaaaaaa-aaaa-aaaa-aaaa-aaaaaaaaaaaa",
          }
        }
      ]
    }
  }
}
```



GraphQL-tjenester

Geometri filtrering

- Anvendere kan foretage geometri filtrering på registres geometri felter.
- Understøttede filtre:
 - ST_Contains
 - ST_Covers
 - ST_Crosses
 - ST_Intersects
 - ST_Within
 - ST_Overlaps
 - ST_Touches
 - ST_CoveredBy
- Filtre kan kombineres med *and* og *or*



GraphQL-skema

```
input SpatialFilterInput {
  and: [SpatialFilterInput!]
  or: [SpatialFilterInput!]
  contains: SpatialContainsOperationFilterInput
  covers: SpatialCoversOperationFilterInput
  crosses: SpatialCrossesOperationFilterInput
  intersects: SpatialIntersectsOperationFilterInput
  within: SpatialWithinOperationFilterInput
  overlaps: SpatialOverlapsOperationFilterInput
  touches: SpatialTouchesOperationFilterInput
  coveredBy: SpatialCoveredByOperationFilterInput
}
```

"Filter type for the spatial method ST_Within. Is applied as ST_Within(A, B) where A is the register data field and B is the input value."

```
input SpatialWithinOperationFilterInput {
  "The WKT representation of the geometry"
  wkt: String!
  "The coordinate reference system of the geometry."
  crs: Int!
}
```


Detaljer kan ændre sig

NOTE: Viste query er simplificeret



GraphQL-tjenester

Geometri filtrering - Within



Query

Hent:

DAR NavngivenVej

Registreringstid:

I dag

Virkningstid:

I dag

Filter:

ST_Within geofiltrering med polygon

```
POLYGON ((550332 6314743,  
          550328 6315051,  
          550666 6315055,  
          550670 6314747,  
          550332 6314743))
```

Felter:

```
id_lokalid,  
registreringFra  
registreringTil  
virkningFra  
virkningTil  
vejnavn  
vejnavnebeliggenhed_vejnavnelinje
```



Query

```
query DARGeoEksempel {  
  DAR_NavngivenVej(  
    registreringstid: "2024-10-23T10:00:00.123456Z"  
    virkningstid: "2024-10-23T10:00:00.123456Z"  
    where: {  
      vejnavnebeliggenhed_vejnavnelinje: {  
        within: {  
          crs: 25832  
          wkt: "POLYGON ((550332 6314743,  
                        550328 6315051,550666 6315055, 550670 6314747, 550332  
                        6314743))"  
        } } })  
      {  
        id_lokalId  
        registreringFra  
        registreringTil  
        virkningFra  
        virkningTil  
        vejnavn  
        vejnavnebeliggenhed_vejnavnelinje {  
          wkt  
        }  
      }  
    }  
  }  
}
```

Detaljer kan ændre sig

GraphQL-tjenester

Geometri filtrering - Within



Resultat

```
{  
  "data": {  
    "DAR_NavngivenVej": [  
      {  
        "id_lokalId": "c2ad1452-88cb-4b49-a1e5-da1b68752a12",  
        "registreringFra": "2023-10-12T07:59:25.554381Z",  
        "registreringTil": null,  
        "virkningFra": "2023-10-12T07:59:25.554381Z",  
        "virkningTil": null,  
        "vejnavn": "Danefæet",  
        "vejnavnebeliggenhed_vejnavnelinje": {  
          "wkt": "MULTILINESTRING Z((550543.78 6314975.51 0, [...]))"  
        },  
      },  
      {  
        "id_lokalId": "9dbd098c-4bd9-4291-b53d-9696833935f6",  
        "registreringFra": "2023-10-12T07:59:21.444546Z",  
        "registreringTil": null,  
        "virkningFra": "2023-10-12T07:59:21.444546Z",  
        "virkningTil": null,  
        "vejnavn": "Mønten",  
        "vejnavnebeliggenhed_vejnavnelinje": {  
          "wkt": "MULTILINESTRING Z((550426.47 6314936.97 0, [...]))"  
        },  
      },  
    ],  
  },  
}
```

Vejen 'Hellekisten' er ikke inkluderet i resultatet, fordi ST_Within geofilteret kræver at hele vej geometrien skal være inden for polygonen.



NOTE: Viste resultat er simplificeret



Specialbyggede GraphQL-tjenester

Til særlige behov

- I nogle tilfælde, fx håndtering af CVR, EJF og CPR, er der behov for specialbyggede tjenester.
- Også specialbyggede tjenester kan dog gøre brug af automatiseringsmotoren.
 - Kræver ikke, at der bygges en helt ny vertikal "ved siden af" de autogenererede tjenester.
 - Bygger derimod ovenpå den eksisterende autogenererede "register-data" infrastruktur. Der kan dermed bl.a. bruges:
 - samme underliggende autogenererede kode til at hente data fra databasen.
 - samme autogenererede bitemporale filtreringslogik.
 - Tjenesterne kan gøre brug af samme GraphQL filtreringsmuligheder som de autogenererede tjenester.



Visualisering af forskel mellem GraphQL- og REST-tjenester



Nuværende Datafordeler

```
https://services.datafordeler.dk/BBR/BBRPublic/1/REST/bygning?  
nord=6147800 & syd=6147880 & oest=530500 & vest=530510 & status=6
```

```
[  
  {  
    "datafordelerOpdateringstid": "2021-04-02T04:25:59.562061+02:00",  
    "byg007Bygningsnummer": 21,  
    "byg021BygningensAnvendelse": "421",  
    "byg0260Pførelsesår": 1960,  
    "byg032YdervæggensMateriale": "1",  
    "byg033Tagdækningsmateriale": "3",  
    "byg037KildeTilBygningensMaterialer": "1",  
    "byg038SamletBygningsareal": 65,  
    "byg040BygningensSamledeErhvervsAreal": 65,  
    "byg041BebyggetAreal": 65,  
    "byg053BygningsarealerKilde": "1",  
    "byg054AntalEtager": 1,  
    "byg404Koordinat": "POINT(530502.29 6147876.88)",  
    "husnummer": "0a3f5090-1981-32b8-e044-0003ba298018",  
    "id_lokalId": "0ea64718-1182-4a1a-be1a-4fbe293ca5ce",  
    "kommunekode": "0621",  
    ... }  
  ]
```

Moderniserede Datafordeler

```
https://graphql.datafordeler.dk/BBR/v1?query={BBR_Bygning(where: {  
  status: { eq: "6" }  
  byg404koordinat: { StWithin: "POLYGON((530500 6147880, 530510  
    6147880, 530510 6147800, 530500 6147800, 530500 6147880))"}  
}) {  
  byg007Bygningsnummer, byg0260Pførelsesår, byg404Koordinat,  
  husnummer, id_lokalId, kommunekode  
}}
```

```
{  
  "data": {  
    "BBR_Bygning": {  
      {  
        "byg007Bygningsnummer": 21,  
        "byg0260Pførelsesår": 1960,  
        "byg404Koordinat": "POINT(530502.29 6147876.88)",  
        "husnummer": "0a3f5090-1981-32b8-e044-0003ba298018",  
        "id_lokalId": "0ea64718-1182-4a1a-be1a-4fbe293ca5ce",  
        "kommunekode": "0621"  
      }  
    }  
  }  
}
```

Detaljer kan ændre sig

NOTE: Viste query er simplificeret

Overgang til den nye funktionalitet



DATAFORDELER



Overgang til ny funktionalitet

- Fuld paralleldrift – ingen ændring af den eksisterende funktionalitet ved idriftsættelsen.
- Skifte fra nuværende REST-tjenester til entitetsbaserede GraphQL tjenester.
- Når nye GraphQL tjenester skal bruges, anvendes ny API-gateway, ny portal og ny sikkerhed.
- Ved beskyttede data skal der ansøges om adgang på ny via ny portal og nye processer.
- **Påvirkning af anvendere**
 - Anvendere skal skifte til de nye tjenester – i eget tempo, under paralleldrift.
 - Nye tjenester tilgås via ny API-gateway, ny portal og ny sikkerhed.
 - **Transitionsguide** om, hvordan anvendere konkret laver skiftet – iht. hentning og brug af data. Klar til ekstern test.
- **Påvirkning af registre**
 - Ingen påvirkning af registersystemer.
 - Skal ikke længere lave tungt DLS-arbejde for at få udstillet data som filer.
 - Har taget stilling til, hvilke entiteter der ønskes udstillet og disses bitemporale model samt beskyttede data.



Proces og principper for overgangen

- En moderniseret Datafordeler betyder forandringer.
- Vi har fokus på den kørende forretning og stabil drift.
- Sådan gør vi:
 - Ny funktionalitet sættes i drift parallelt med eksisterende funktionalitet.
 - Piloter skifter til ny funktionalitet, og der gennemføres driftsprøve.
 - Når stabil drift er verificeret, planlægger vi udfasning af eksisterende funktionalitet i dialog med registre og anvendere.
 - Der varsles med minimum 6 måneder ved større ændringer.
 - Vi leverer dokumentation, vejledning og transitionsguides.



Videre forløb



DATAFORDELER



Videre forløb

- Udviklingen af udviklingsleverance 2 fortsætter frem til januar.
- Overlevering og afprøvning forventes i januar og februar.
- Forventet idriftsættelse ultimo februar.

I hører fra os!

- Generelt informationsmøde om UL2 forud for ekstern test og idriftsættelse.
- Informationsmøde om plan og scope for UL3.
- Løbende nyhedsbreve.
- Information om udviklingsfasen på Datafordeler.dk.





Nyhedsbrev og mere information

- Læs mere om projektet:
<https://datafordeler.dk/modernisering>
- Tilmeld jer til vores [nyhedsbrev](#) og få besked, når der er nyt.
 - Via ovenstående link findes tilmelding:



