Stamdataservice på NSP

Guide til anvendere

Indholdsfortegnelse

1 Formål 3

2 Introduktion til Stamdataservicen 4

2.1 Adgang til Stamdataservicens funktionalitet 4

3 Kopiregisterservicen (KRS) 5

3.1 Struktur af kald til KRS 5

3.2 Fejlsituationer 5

3.3 Endpoint URL 6

3.4 Eksempel på kald til KRS 6

3.5 Detaljeret svar fra KRS 7

3.6 Parsing af output 7

3.7 Paginering 8

4 Registre 9

4.1 Opsummering af registre udstillet gennem stamdataservicen 9

4.2 Særlige forhold for de udstillede registre 10

5 Enkeltopslag i registre 12

5.1 Forespørgsler og dataformat 12

5.2 Enkeltopslag i autorisationsregistret 12

5.3 CPR registret: Det Gode CPR Opslag 14

5.4 CPR registret: getPersonDetails 15

5.5 CPR-Abonnementsservice 18

6 Ændringslog 22

Bilag A: Eksempel på kald til abonnementsservicen 23

# Formål

Nærværende dokument indgår i den samlede dokumentationspakke for stamdataregistret på NSP. Dokumentationspakken giver tilsammen det fulde overblik over dokumentationen.

”Guide til anvendere” er målrettet udviklere og arkitekter hos leverandører, som ønsker at anvende funktionaliteten udstillet af stamdataservicen. Guiden indeholder snitfladebeskrivelser,

Guiden kan indeholde kodeeksempler, interaktionsbeskrivelser, servicebeskrivelser, snitfladebeskrivelser mv.

# Introduktion til Stamdataservicen

Stamdataservicen er en national service udstillet på NSP, som giver adgang til relevante nationale stamdataregistre, opsamlet af NSP fra forskellige myndigheder og dataansvarlige enheder.

Stamdataservicen udstiller flere snitflader, der tilsammen giver mulighed for både at holde komplette lokale kopier af stamdataregistrene ajour og for at lave enkeltopslag i udvalgte registre. For en overordnet beskrivelse af den udstillede funktionalitet henvises der til Arkitektur og Design, der på lige fod med nærværende dokument indgår i den samlede dokumentationspakke for stamdataservicen.

## Adgang til Stamdataservicens funktionalitet

Anvendelse af stamdataservicen forudsætter en aftale med NSP-operatøren. Yderligere oplysninger om betingelserne for anvendelse af stamdataservicen, herunder håndtering af dataansvar og tekniske forudsætninger, kan fås ved henvendelse til NSP-operatørens servicedesk (<http://nsp.nsi.dk>).

Adgangen til funktionaliteten er styret i flere ”lag”, idet de enkelte stamdataregistre indeholder forskelligartet information, og kan være styret af forskellige politikker. I praksis styres adgangen gennem whitelists på CVR nummer niveau[[1]](#footnote-1) for de enkelte services.

Alle snitflader udstillet af stamdataservicen er baseret på Den Gode Web-Service 1.0.1 (DGWS 1.0.1) som er en national standard for identitetsbaserede webservices i sundhedssektoren. I forhold til stamdataservicen forudsættes der autentifikation på DGWS niveau 3, dvs. brug af STS-signerede system-IDkort.

NSI leverer gratis biblioteksunderstøttelse af store dele af webservice kommunikationen, herunder generering og parsning af IDkort og headere. Der henvises til DGWS dokumentationen (<http://www.medcom.dk/wm110731>) for generel information om DGWS og SEAL dokumentationen (<http://www.sosi.dk/twiki/bin/view/ProjectManagement/SOSIProducts>) for generel information om biblioteksunderstøttelsen.

# Kopiregisterservicen (KRS)

Kopiregisterservicen (KRS) giver systemer mulighed for at etablere og ajourføre en lokal kopi af et register, som f.eks. CPR-registret eller autorisationsregistret.

Som beskrevet i afsnit 2.1 er det nødvendigt at etablere en aftale hos NSP-operatøren for at få adgang til de respektive registre. Præcist hvilke registre og data-typer man får adgang til er afhængigt af aftalen. Der henvises til dokumentet *Design og Arkitektur* for en oversigt over hvilke registre, KRS pt. kan levere data fra.

## Struktur af kald til KRS

Kopiregisterservicen tager følgende input:

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Beskrivelse** | **Eksempel** |
| Register | Det register, der ønskes udtræk fra. | ”CPR” |
| Datatype | Hvert register er opdelt i en række datatyper, og et kald til KRS returnerer en enkelt af disse. | ”person” |
| Versionsnummer | Af hensyn til bagud­kompabilitet versioneres udtræks­funktionaliteten for hvert enkelt register | ”V1” |
| Offset | En parameter, der angiver hvorfra i servicens udtræk af ændrede data svaret skal påbegyndes. | ”1024” |
| MaxRowCount | Det maksimalt antal returnerede rækker. Anvendes til at styre størrelsen af svarene fra KRS. | ”512” |

Kaldet til Kopiregisterservicen er formelt specificeret i WSDL’en *stamdata\_krs.wsdl*, der kan rekvireres ved henvendelse til NSP-operatøren.

En komplet gennemgang af tilgængelige registre, datatyper samt versioner er at finde i dokumentet ”Registerspecifikation for Anvendere”.

## Fejlsituationer

Hvis der er en fejl i forespørgslen eller der opstår en fejl på serveren vil servicen returnere en fejlmelding af typen *ReplicationFault* med en beskrivelse af fejlen (jævnfør WSDL som specificeret i ovenstående afsnit). Hvis forespørgslen går godt returnerer servicen en besked af typen *ReplicationResponse* der indeholder et ”any”-element. Dette any-element er af typen atom baseret på ATOM 1.0 [[RFC4287](http://tools.ietf.org/html/rfc4287)]. ATOM er en syndikeringsprotokol og designet til at holde styr på ændringer i en ressource. I dette tilfælde er ressourcerne datatyperne i et register. Hvordan dette atom-element skal tolkes er forklaret nærmere i afsnit 3.3.

## Endpoint URL

Kopi register servicen er som udgangspunk konfigureret på

<hostnavn>:8080/stamdata-batch-copy-ws/service/StamdataReplication

og wsdl kan hentes ved at sætte ?wsdl efter altså:

<hostnavn>:8080/stamdata-batch-copy-ws/service/StamdataReplication?wsdl

## Eksempel på kald til KRS

Som klient sender man et ReplicationRequest til servicen.

**Forespørgsel:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Header>

…

</S:Header>

<S:Body>

<ns1:ReplicationRequest xmlns:ns1="http://nsi.dk/2011/10/21/StamdataKrs/">

<register>cpr</register>

<datatype>person</datatype>

<version>1</version>

<offset>0</offset>

</ns1:ReplicationRequest>

</S:Body>

</S:Envelope>

Headeren skal indeholde en DGWS 1.0.1 header.

Hvis alt går som forventet og forespørgslen bliver godkendt modtages et svar:

**Svar:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Header>

…

</S:Header>

<S:Body>

<ns1:ReplicationResponse xmlns:ns1="http://nsi.dk/2011/10/21/StamdataKrs/">

<atom:feed xmlns:atom="http://www.w3.org/2005/Atom" xmlns="http://nsi.dk/-/stamdata/3.0/cpr">

<atom:id>tag:nsi.dkersonLookupWithSubscriptionIntegrationTest.java,2011:cpr/person/v1</atom:id>

<atom:updated>2011-10-25T07:02:08.045Z</atom:updated>

<atom:title>Stamdata Registry Feed</atom:title>

<atom:author>

<atom:name>National Sundheds IT</atom:name>

</atom:author>

…

</atom:feed>

</ns1:ReplicationResponse>

</S:Body>

</S:Envelope>

Opstår der derimod en fejl, bliver en DGWS 1.0.1 Fault sendt tilbage, f.eks.:

**Fejl:**

<soapenv:Envelope ...>

<soapenv:Header>...</soapenv:Header>

<soapenv:Body>

<soapenv:Fault>

<faultcode>Server</faultcode>

<detail>

<medcom:FaultCode>expired\_idcard</medcom:FaultCode>

</detail>

<faultstring>The ID card has expired.</faultstring>

</soapenv:Fault>

</soapenv:Body>

</soapenv:Envelope>

## Detaljeret svar fra KRS

Her følger et mere detaljeret output fra KRS. Tolkning af dette er beskrevet i afsnittet efter.

<atom:feed xmlns:atom="http://www.w3.org/2005/Atom" xmlns="http://nsi.dk/-/stamdata/3.0/cpr">

<atom:id>tag:nsi.dk,2011:cpr/person/v1</atom:id>

<atom:updated>2011-11-07T09:56:12.278Z</atom:updated>

<atom:title>Stamdata Registry Feed</atom:title>

<atom:author>

<atom:name>National Sundheds IT</atom:name>

</atom:author>

<atom:entry>

<atom:id>tag:nsi.dk,2011:cpr/person/v1/13206597710000000085</atom:id>

<atom:title/>

<atom:updated>2011-11-07T09:56:11.000Z</atom:updated>

<atom:content type="application/xml">

<person>

<cpr>0102451234</cpr>

<koen>M</koen>

<fornavn>Hans</fornavn>

<mellemnavn/>

<efternavn>Hansen</efternavn>

<coNavn></coNavn>

<lokalitet/>

<vejnavn>Ligusterv&#xE6;nget</vejnavn>

<bygningsnummer>42</bygningsnummer>

<husnummer>123</husnummer>

<etage>12</etage>

<sideDoerNummer>th.</sideDoerNummer>

<bynavn>Enby</bynavn>

<postnummer>1448</postnummer>

<postdistrikt>N&#xF8;ddebo</postdistrikt>

<status>01</status>

<gaeldendeCPR>3105459876</gaeldendeCPR>

<foedselsdato>1947-12-24T00:00:00+01:00</foedselsdato>

<stilling/>

<vejKode>740</vejKode>

<kommuneKode>314</kommuneKode>

<validFrom>2011-11-07T10:56:21+01:00</validFrom>

<validTo>2012-11-07T10:56:11+01:00</validTo>

</person>

</atom:content>

</atom:entry>

</atom:feed>

## Parsing af output

Det mest interessante ved et response er ’entry’-elementerne. Hvert ’entry’-element indeholder et snapshot af en record fra registret. Med snapshot menes, at det var sådan data så ud på et bestemt tidspunkt. Stamdata bruger fuld historik når man laver et udtag. Det vil sige at man f.eks. kan få information om hvordan en person record så ud for 1 år siden – med adresse, navn etc. Der er ingen garanti for hvor langt tilbage i tiden stamdata har information.

Hvert entry element har et ’content’-element som indeholder alt domæne-data. I eksemplet ovenfor er det datatypen ’person’ som er indeholdt.

Det er to slags nøgler som identificerer en record: Unikke nøgler (indeks) og versionsnumre.

### Unikke Nøgler

Hver datatype har et nøgleelement som unikt bestemmer en record. Eksempelvis identificeres en person unikt ved sit CPR-nummer i CPR-registret. Derfor vil man, når man persisterer records fra cpr/person/v1 bruge cpr-elementet som unik nøgle. Unikke nøgler for de enkelte datatyper er beskrevet i listen over registre i dokumentet "Registerspecifikation for Anvendere”.

### Revisionsnummer

Revisionsnumre bestemmer en record unikt indenfor en given datatype. Man kan se det som en primær nøgle i en database. Et revisionsnummer er også en slags historisk ID. Det vil sige at det bestemmer en record unikt i historien, i modsætning til unikke nøgler.

Revisionsnummeret kan findes ved at kigge på entry-elementernes id-element. F.eks. i

tag:nsi.dk,2011:sor/apotek/v1/168763721800723

er 1687637218007236 revisionsnummeret.

Med andre ord, bestemmer en unik nøgle en bestemt entitet, revisionsnummeret bestemmer et snapshot af den entitet. Begge disse numre er vigtige på flere måder. Den unikke nøgle kan optræde i flere forskellige entry-elementer. Det er på grund af at en entitet ændre sig med tiden, men revisionsnumre vil altid være forskellige.

## Paginering

Antallet af records i et register kan være meget stort, i visse tilfælde flere millioner records. Derfor bliver man nødt til at opdele et udtræk i flere kald. Response fra det tidligere eksempel, er et eksempel på en page. Når man er klar til at hente næste page sender man et request med nyt offset i ReplicationResponse beskeden. Offset-parameteren i næste request sættes til det sidste versionsnummeret man har modtaget.

Det er muligt at angive hvor mange records man højest ønsker i et svar fra servicen ved at sætte parameteren maxRecords i ReplicationRequest forespørgelsen. Der er på serveren en øvre grænse på denne parameter der overskriver for høje værdier i en forespørgelse. Hvis parameteren ikke specificeres i kaldet indsættes denne grænse som antal records.

# Registre

Dette afsnit danner et overblik over de forskellige registre der findes i Stamdata.

Hvert register kan have flere datatyper (også kaldet tabeller eller views). Der findes XSD-skemaer for alle datatyper i et register. Skemaerne kan rekvireres ved henvendelse til NSP-operatøren.

For hvert register vil der være en beskrivelse af registret og information om hvilke data der eventuelt er undladte fra kilderegistret.

## Opsummering af registre udstillet gennem stamdataservicen

I nedenstående tabel ses en opsummering for hvert enkelt register. I de følgende afsnit ses supplerende informationer for registrene.

| **Register** | **Manglende delregistre** | **Fremtidig historik** | **Opdaterings­frekvens** |
| --- | --- | --- | --- |
| **CPR**  *Registret er baseret på det CPR udtræk, der er navngivet “CPR Direkte Personnummer OFF4 V3.* | Ja, se CPR | Nej | Dagligt. Der er enkelte dage hvor der ikke modtages opdateringer (typisk søn- og helligdage og i ferien). |
| **Autorisation**  *Sundhedsstyrelsens offentlige autorisations­register.* | Nej | Nej | Modtages dagligt. |
| **SKS**  *Register over SKS for sygehus-afdelinger.* | Nej | Nej | Dagligt. |
| **SOR**  *Sundhedsvæsenets Organisationsregister.* | Ja, se SOR (Sundhedsvæsenets OrganisationsRegister) | Ja | Dagligt. |
| **Taksten**  *Lægemiddelstyrelsens officielle takst over lægemidler.* | Nej | Ja | For hovedtaksten, hver 14. dag.  For opdaterings­takster, dagligt, dog ikke weekender og ferier. |
| **Yderregistret**  *Ydere i sundhedsvæsenet, f.eks. klinikker, privatpraktiserende læger og speciallæger.* | Nej | Nej | Dagligt. |
| **”Sikrede”**  *Information om sygesikring, bl.a. borgeres relation til ”egen læge” samt sygesikringsgruppe.* | Ja, se ”Sikrede” | Nej | Dagligt. |
| **”Bemyndigelse”**  Information om sundhedsfagliges uddelegering af rettigheder | Nej | Nej | Ved ændringer/tilføjelser – da bemyndigelser skal processeres hurtigst muligt. |
| **”NPI-SOR relationer”**  relationstabeller der gør søgning i SOR data nemt for NPI projektets samtykkeservice | Nej | Nej | Dagligt, Hænger sammen med det eksisterende SOR-register. Data må ikke afvige fra dette. |

## Særlige forhold for de udstillede registre

### CPR

For mange recordtyper er det angivet hvornår en evt. ændring er gældende fra. Det har dog vist sig at disse datoer ikke altid er korrekte, og kan ikke anvendes som validFrom. I stedet er der valgt at ændringer altid starter på udtræksdatoen.

Ikke alle ændringer kan udtrykkes i udtrækket. F.eks. er der ingen mulighed for at slette en forældremyndighed.

Det er ikke hele CPR der er tilgængelige igennem Stamdataservicen. Det er kun records af følgende typer der er tilgængelige:

* P12170-001 Personoplysninger
* P12170-002 Aktuelle adresseoplysninger
* P12170-003 Klarskriftsadresse
* P12170-004 Beskyttelse
* P12170-008 Aktuelle navneoplysninger
* P12170-014 Børn
* P12170-016 Forældremyndighed

P12170-017 Umyndiggørelse + værge

### Autorisation

Der leveres et fuldt udtræk ved hver opdatering.

### SKS

Der leveres et fuldt udtræk ved hver opdatering.

### SOR (Sundhedsvæsenets OrganisationsRegister)

Udtrækket hentes fra sundhedsstyrelsens offentlige FTP server.

SOR indeholder data om mange af sundhedsvæsnets organisationer. Her importeres kun:

* Læger, speciallæger og lægepraksis
* Apoteker

Sygehuse og afdelinger hvor hele hierarkiet har SKS koder.

### Taksten

Taksten hentes fra LMS FTP server. Takst filerne er opdelt i en hovedtakst der kommer hver 14 dag, samt daglige opdateringstakster hvor kun prisoplysninger bliver ændret.

### Yderregistret

Der gælder ingen særlige forhold for yderregistret.

### ”Sikrede”

Hver record indeholder både information om nuværende forhold om information om fremtidige forhold. Er en person ved at skifte læge vil dette fremgå som en fremtidig relation.

Ikke hele registret er tilgængeligt. Information som allerede er tilgængeligt fra f.eks. CPR registret er ikke inkluderet.

### Bemyndigelser

Bemyndigelsesfiler, kan leveres både som fuldt udtræk og som delta udtræk, filerne skrives lokalt på bemyndigelsessservice serverens disk, og et FTP job i bemyndigelsesservicen, kan derefter ”skubbe” filerne op på stamdata serveren i en relevant folder.

### NPI-SOR Relationer

Der leveres et fuldt udtræk ved hver opdatering, og der føres ikke historik på disse tabeller, da de kun indeholder aktuelle og gyldige data.

# Enkeltopslag i registre

Stamdataservicen tilbyder en service til såkaldte enkeltopslag i følgende registre:

* Autorisationsregistret
* CPR registret

Enkeltopslag i et af de ovennævnte registre foregår som et webservice kald til stamdataservicen på NSP med angivelse af søgekriterier. De konkrete parametre og eventuelle særlige forhold der gør sig gældende for kald til enkeltopslags-snitfladen på stamdataservicen er beskrevet i de følgende afsnit.

## Forespørgsler og dataformat

Enkeltopslags-snitfladen er udstillet som en DGWS 1.0.1 identitetsbaseret webservice. Adgang til enkeltopslagsservices forudsætter en aftale med NSP operatøren (adgang styres på CVR-nummer). Metoderne for de udstillede registre er beskrevet i separate WSDL filer, der kan rekvireres ved henvendelse til NSP operatøren.

Fælles for de udstillede enkeltopslagsservices er at der kræves STS-signerede system IDkort (dvs. DGWS niveau 3).

## Enkeltopslag i autorisationsregistret

Det er muligt at lave onlineopslag i stamdataservicens kopi af Sundhedsstyrelsens autorisationsregister til fremsøgning af eventuelle autorisationer for en given sundhedsfaglig person. Metoden tager personens CPR nummer som input, og returnerer personens navn og samtlige autorisationer der er i kraft for vedkommende.[[2]](#footnote-2)

En person kan have flere gyldige autorisationer. Hver autorisation som bliver returneret fra webservicen er forbundet med en uddannelseskode. Tabellen nedenfor viser en liste over kendte uddannelseskoder og deres tilknyttede uddannelse. Der henvises til Sundhedsstyrelsens løbende vedligeholdte liste af uddannelser, hvortil der udstedes sundhedsfaglige autorisationer for yderligere information (<http://autregwebservice.sst.dk/autregservice.asmx>). Faggruppenavnet bliver også returneret såfremt uddannelseskoden er kendt af enkeltopslagsservicen.

|  |  |
| --- | --- |
| **Uddannelseskode** | **Faggruppenavn** |
| 4498 | Optiker |
| 5151 | Fysioterapeut |
| 5153 | Ergoterapeut |
| 5155 | Fodterapeut |
| 5158 | Radiograf |
| 5159 | Bioanalytiker |
| 5166 | Sygeplejerske |
| 5175 | Jordemoder |
| 5265 | Kiropraktor |
| 5431 | Tandplejer |
| 5432 | Klinisk Tandtekniker |
| 5433 | Tandlæge |
| 5451 | Klinisk diætist |
| 7170 | Læge |
| 9495 | Bandagist |

Hvis CPR-nummeret fra forespørgselen er tilknyttet en eller flere autorisationer vil også for- og efternavn for personen blive returneret. Forespørges der på et ugyldigt CPR-nummer eller en person uden gyldige autorisationer returneres der af sikkerhedshensyn tomme navnefelter i svaret – også selv om CPR-nummeret er et korrekt CPR-nummer for en eksisterende person.

### Eksempel: Én eller flere autorisationer

Nedenfor ses et ”solskinsscenarie”, hvor der forespørges på en sundhedsfaglig, der har en enkelt gyldig autorisation.

**Forespørgsel:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Header>

…

</S:Header>

<S:Body>

<AuthorizationRequestStructure xmlns="http://nsi.dk/-/stamdata/3.0" xmlns:ns2="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#" xmlns:ns4="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:ns5="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:ns6="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd">

<cpr>1111122222</cpr>

</AuthorizationRequestStructure>

</S:Body>

</S:Envelope>

**Svar:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Header>

…

</S:Header>

<S:Body>

<AuthorizationResponseStructure xmlns="http://nsi.dk/-/stamdata/3.0" >

<cpr>1111122222</cpr>

<firstName>Peter</firstName>

<lastName>Andersen</lastName>

<authorization>

<authorizationCode>B1114</authorizationCode>

<educationCode>2131</educationCode>

</authorization>

</AuthorizationResponseStructure>

</S:Body>

</S:Envelope>

I eksemplet er Peter Andersen tilknyttet CPR 1111122222, og han har en autorisation.

### Eksempel: Ingen gyldige autorisationer

Forespørges der på en ikke-eksisterende person eller en person, der ikke har en gyldig autorisation, svares der blot med en tom AutorizationResponseStructure.

I eksemplet nedenfor er headere udeladt, og der henvises til eksemplet i afsnit 5.2.1 for en tilsvarende forespørgsel hvor headere er vist.

**Forespørgsel:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

<S:Header>

…

</S:Header>

<S:Body>

<AuthorizationRequestStructure xmlns="http://nsi.dk/-/stamdata/3.0" xmlns:ns2="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns3="http://www.w3.org/2000/09/xmldsig#" xmlns:ns4="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:ns5="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:ns6="http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd">

<cpr>1111122244</cpr>

</AuthorizationRequestStructure>

</S:Body>

</S:Envelope>

**Svar:**

<soapenv:Envelope …>

<soapenv:Header>…</soapenv:Header>

<soapenv:Body>

<AuthorizationResponseStructure xmlns="http://nsi.dk/-/stamdata/3.0">

<cpr>1111122222</cpr>

</AuthorizationResponseStructure>

</soapenv:Body>

</soapenv:Envelope>

Da der ikke er nogen tilknyttede autorisationer, returneres en tom AuthorizationResponseStructure uden angivelse af for- og efternavn.

## CPR registret: Det Gode CPR Opslag

Stamdataservicen udstiller mulighed for enkeltopslag i CPR registret gennem en implementering af ’Det Gode CPR Opslag’ som specificeret af MedCom (<http://www.medcom.dk/wm110491>).

### WSDL

Wsdl filer for servicen kan findes på medcoms svn her :

<http://svn.medcom.dk/svn/drafts/CPR-opslag/trunk/CPR-opslag/wsdl/>

Hver opmærksom på at der tilføjes Den Gode WebService felter på, så vil nok være nemmere at hente wsdl direkte fra servicen, det gøres ved at kalde endpoint med ?wsdl tilføjes som sådan <http://endpointurl?wsdl> da denne indeholder de tilføjede felter.

Sidst men ikke mindst kan de også hentes med dgws felter direkte fra leverandørens svn her:

<https://fisheye.nspop.dk/browse/public/components/sdm/latest/code/nsp/cpr-ws/src/main/webapp/WEB-INF/wsdl>

### Endpoints

Det gode CPR opslag er udstillet i version 1.0.0 og version 1.0.2 som er tilgængelig på hver deres endpoints.

For 1.0.0:

<http://stamdatahost:8080/stamdata-cpr-ws/service/DetGodeCPROpslag>

For 1.0.2

<http://stamdatahost:8080/stamdata-cpr-ws/service/DetGodeCPROpslag-1.0.2>

Det anbefales at køre på version 1.0.2 da ikke alle personer kan hentes i 1.0.0 uden validerings fejl.

### Navne- og adressebeskyttelse

Generelt gælder der at hvis en record i CPR registret er markeret med navne- og adressebeskyttelse, da vil hele output være beskyttet ud fra følgende regler

|  |  |
| --- | --- |
| Søgning via | Hvilke data er beskyttet |
| CPR-nummer | Alle output felter på nær CPR-nummeret |
| Liste af CPR- numre | Alle output felter på nær CPR-nummeret |
| Navn | Alle output felter |
| Fødselsdato | Alle output felter |

## CPR registret: getPersonDetails

Stamdataservicen udstiller endvidere en variant af ’Det Gode CPR Opslag’, der kan anvendes til søgninger i CPR registret med følgende input.

* CPR-nummer
* liste af CPR-numre
* fødselsdato
* navn

Servicen har en enkelt snitflade: **getPersonDetails**, der er implementeret i forskellige udgaver der muliggør forskellige typer af søgninger. I de følgende afsnit er kaldmulighederne beskrevet med tilhørende eksempler.

Servicen er formelt beskrevet i WSDL og XSD skemaer der kan findes her:

<src-root>/Dokumentation/bilag/StamdataPersonLookupService/

Servicen tager hensyn til navnebeskyttelse, og det anbefales at undersøge elementet PersonInformationProtectionIndicator for om værdien er true da personen i så fald er navnebeskyttet. Værdierne som de forskellige elementer er erstattet med er afhængig af hvilken XSD type elementet har (dvs. der tages hensyn til de navnebeskyttede felters skema-definition).

### Eksempel 1: getPersonDetails(<Personnummer>)

I denne variant af getPersonDetails søges på et specifikt CPR nummer. Findes personen i CPR registret returneres de tilhørende CPR informationer. Findes personen ikke i CPR registret returneres et tomt svar.

I dette eksempel vises et kald, og en række mulige svar, afhængigt af søgeresultatet.

**Kald 1: (Opslag vha. CPR-nummer)**

<stam:PersonLookupRequest>

<CivilRegistrationNumberPersonQuery>

1234567890

</CivilRegistrationNumberPersonQuery>

</stam:PersonLookupRequest>

**Svar 1: (Personen blev fundet)**

En søgning, hvor den fremfundne person *ikke* er adressebeskyttet, giver anledning til følgende svar:

<stam:PersonLookupResponse>

<ns:PersonInformationStructure>

<ns1:RegularCPRPerson>

<ns1:SimpleCPRPerson>

<ns2:PersonNameStructure>

<ns3:PersonGivenName>**Peter**</ns3:PersonGivenName>

<ns3:PersonSurnameName>**Andersen**</ns3:PersonSurnameName>

</ns2:PersonNameStructure>

<ns1:PersonCivilRegistrationIdentifier>

1234567890

</ns1:PersonCivilRegistrationIdentifier>

…

</ns1:SimpleCPRPerson>

…

<ns6:PersonInformationProtectionIndicator>

**false**

</ns6:PersonInformationProtectionIndicator>

</ns1:RegularCPRPerson>

<ns:PersonAddressStructure>…</ns:PersonAddressStructure>

</ns:PersonInformationStructure>

</stam:PersonLookupResponse>

**Svar 2: (Personen blev fundet, men har navnebeskyttelse)**

En søgning, hvor den fremfundne person er navnebeskyttet, giver et partielt svar. Personen bliver stadig returneret men beskyttede data udelades i svaret og erstattes af en generisk tekst:

<stam:PersonLookupResponse>

<ns:PersonInformationStructure>

<ns1:RegularCPRPerson>

<ns1:SimpleCPRPerson>

<ns2:PersonNameStructure>

<ns3:PersonGivenName>**Navnebeskyttet**</ns3:PersonGivenName>

<ns3:PersonSurnameName>**Navnebeskyttet**</ns3:PersonSurnameName>

</ns2:PersonNameStructure>

<ns1:PersonCivilRegistrationIdentifier>

1234567890

</ns1:PersonCivilRegistrationIdentifier>

…

</ns1:SimpleCPRPerson>

…

<ns6:PersonInformationProtectionIndicator>

**true**

</ns6:PersonInformationProtectionIndicator>

</ns1:RegularCPRPerson>

<ns:PersonAddressStructure>…</ns:PersonAddressStructure>

</ns:PersonInformationStructure>

</stam:PersonLookupResponse>

**Svar 3: (Personen blev ikke fundet)**

Hvis søgningen ikke giver et positivt resultat, returneres der et tomt svar:

<stam:PersonLookupResponse />

### Kald 2: (Opslag vha. navn)

getPersonDetails understøtter også søgninger på personnavn. Svar ved kald til denne variant er strukturelt identiske med svar ved søgninger på CPR nummer eller liste af CPR numre, og der henvises til disse eksempler for svarmuligheder.

<stam:PersonLookupRequest>

<NamePersonQuery>

<ns:PersonGivenName>Anne</ns:PersonGivenName>

<!-- Det er ikke nødvendigt at angive mellemnavn -->

**<ns:PersonMiddleName>Søgaard</ns:PersonMiddleName>**

<ns:PersonSurnameName>Petersen</ns:PersonSurnameName>

</NamePersonQuery>

</stam:PersonLookupRequest>

Angives mellemnavnet ikke, ignoreres det og records med matchende for- og efternavn returneret. Se kaldet med en liste af CPR-numre for at se scenariet hvor flere person-elementer returneres.

### Kald 3: (Opslag vha. fødselsdato)

getPersonDetails understøtter også søgninger på fødselsdato. Svar ved kald til denne variant er strukturelt identiske med svar ved søgninger på CPR nummer eller liste af CPR numre, og der henvises til disse eksempler for svarmuligheder.

<stam:PersonLookupRequest>

<BirthDatePersonQuery>1991-09-22</BirthDatePersonQuery>

</stam:PersonLookupRequest>

### Kald 4: (Opslag vha. en liste af CPR numre)

getPersonDetails understøtter også søgninger med en liste af CPR numre som input. For hvert CPR nummer på listen returneres CPR informationerne (betinget af eventuel navnebeskyttelse) eller et tomt svar, hvis CPR nummeret ikke findes.

<stam:PersonLookupRequest>

<CivilRegistrationNumberListPersonQuery>

<CivilRegistrationNumber>1111111111</CivilRegistrationNumber>

<CivilRegistrationNumber>**2222222222**</CivilRegistrationNumber>

<CivilRegistrationNumber>3333333333</CivilRegistrationNumber>

</CivilRegistrationNumberListPersonQuery>

</stam:PersonLookupRequest>

**Svar 4: (Resultat med flere records)**

<stam:PersonLookupResponse>

<ns:PersonInformationStructure>

…

<ns:PersonCivilRegistrationIdentifier>

1111111111

</ns:PersonCivilRegistrationIdentifier>

…

</ns:PersonInformationStructure>

<ns:PersonInformationStructure>

…

<ns:PersonCivilRegistrationIdentifier>

3333333333

</ns:PersonCivilRegistrationIdentifier>

…

</ns:PersonInformationStructure>

</stam:PersonLookupResponse>

## CPR-Abonnementsservice

Stamdata tilbyder mulighed for at man for en liste af CPR-numre kan abonnere på ændringer i CPR-registret. Brugen af abonnementsservicen er skarpt opdelt i to snitflader:

* Vedligeholdelse af abonnementslisten (tilføj og slet CPR-numre)
* Udtræk af ændringer for CPR-numrene på listen

Vedligeholdelsen af abonnementslisten er implementeret ved hjælp af den Generiske Opsamlingsservice (GOS), der er en NSP baseret service til opsamling af informationer af forskellig art. Til brug for abonnementsservicen anvendes en særlig opsamlingstype, der giver mulighed for dels at tilføje, dels at slette CPR-numre fra listen. I afsnit 5.4.1 nedenfor er det angivet hvordan dette gøres i praksis.

Udtræk af ændringer for CPR-numrene på abonnementslisten fås ved kald til abonnementsservicen, der overvåger CPR registret og returnerer en liste af CPR-numre, der har fået opdateret oplysninger i CPR registret siden sidste udtræk. I kaldet kan det også angives at svaret skal indeholde de komplette CPR-oplysninger for de berørte CPR-numre.



Figur – Skitse af det samlede flow ved brug af abonnementsservicen

Detaljeret information om indholdet af forespørgsler og svar for de enkelte kald findes i de følgende afsnit.

### Vedligehold af abonnementslisten

Tilføjelse eller sletning af et CPR-nummer på abonnementslisten håndteres ved et kald til GOS med et request, hvor payload overholder dette skema:

<src-root>/common/src/main/resources/xsd/stamdata/subscription.xsd

Ved tilføjelse af et CPR-nummer til abonnementslisten benyttes elementet **subscribe** som payload. Ved sletning af et CPR-nummer fra abonnementslisten benyttes elementet **unsubscribe**.

Et eksempel på payload ved tilføjelse af CPR-nummer 1234567890 ser således ud:

<subscribe xmlns="http://nsi.dk/stamdata/2011/09/">

<cpr>1234567890</cpr>

</subscribe>

Et eksempel på payload ved sletning af CPR-nummer 1234567890 ser således ud:

<unsubscribe xmlns="http://nsi.dk/stamdata/2011/09/">

<cpr>1234567890</cpr>

</unsubscribe>

Generel information om, hvordan GOS’ens services kaldes, herunder reference til den tilhørende WSDL, findes i dette dokument:

<src-root-til-Behandlingsrelations-servicen>/doc/gos/Guide til Anvendere.docx

### Udtræk af ændringer i CPR-data

Der findes to metoder til at få ændringer i CPR-data for CPR-numrene på abonnementslisten:

* *getChangedCprs*, der returnerer en liste af CPR-numre, for hvilke CPR-data er ændret
* *getSubscribedPersonDetails*, der returnerer samme liste som *getChangedCprs*, men som også indeholder CPR-oplysninger for hvert CPR-nummer på listen

Metoderne tager samme input, nemlig en valgfri angivelse af et tidspunkt, hvorfra ændringer skal registreres, men har forskellig output som beskrevet ovenfor. Angives der *ikke* et tidspunkt, holder abonnementsservicen selv regnskab med hvornår metoderne er kaldt sidst. Hvis der ved første udtræk af ændringer ikke angives et tidspunkt, sætter servicen selv tidspunktet til en fortidig værdi (1. januar 1970).

**Bemærk**: De to metoder har fælles lagring af seneste tidspunkt for et kald til en given abonnementsliste. Et kald til den ene metode vil derfor medføre at ved et efterfølgende kald uden tidsangivelse til den anden metode vil tidsstemplet fra det første kald blive anvendt automatisk. Det anbefales derfor at anvendere af abonnementsservicen enten selv holder regnskab med ”siden sidst”, alternativt udelukkende anvender den ene af de to metoder.

Det bemærkes endvidere, at både abonnementslister og tidspunkterne for kald til udtræksmetoderne deles på tværs af samtlige NSP-instanser ved brug af platformens replikeringsmekanisme. Da disse to replikeringer ikke foretages øjeblikkeligt, skal man som klient være forberedt på, at fremtidige requests der foretages tidsmæssigt tæt på det oprindelige, kan returnere cpr-numre, man har set ved tidligere kald.

Fælles for begge metoder er at abonnementslisten automatisk identificeres på basis af det medsendte system IDkort. Cvr-nummer angives derfor *ikke* eksplicit.

Følgende WSDL’er gælder for de to metoder:

* *getChangedCprs*: cprabbs.wsdl
* *getSubscribedPersonDetails*: StamdataPersonLookupWithSubscription.wsdl

De to WSDL’er samt relaterede skemadefinitioner kan rekvireres ved henvendelse til NSP operatøren.

I Bilag A: Eksempel på kald til abonnementsservicen ses et eksempel på request og response ved et kald til getSubscribedPersonDetails, hvor relevante headere er inkluderet i eksemplet. Nedenfor følger et eksempel på et kald til getChangedCprs, hvor kun body er inkluderet i forespørgsel og svar:

**Forespørgsel**:

<CprAbbsRequest xmlns=”http://nsi.dk/cprabbs/2011/10”>

<since>2008-09-29T03:49:45</since>

</CprAbbsRequest>

**Svar**:

<CprAbbsResponse xmlns==”http://nsi.dk/cprabbs/2011/10”>

<changedCprs>1234567890</cpr>

<changedCprs>2345678901</cpr>

</CprAbbsResponse>

# Ændringslog

Kilden til dette dokument kan fås ved henvendelse til NSP-operatøren.

| Version | Dato | Ændring | Ansvarlig |
| --- | --- | --- | --- |
| 1.0 | 2011-04-28 | Initielt Dokument | Trifork |
| 1.1 | 2011-09-26 | Opdateret med cpr tjenester | Trifork |
| 1.2 | 2011-12-01 | Kvalitetssikring | Lakeside |
| 1.3 | 2012-06-19 | Opdateret med bemyndigelse og NPI-SOR relationer | Trifork |
| 1.4 | 2013-05-14 | Tilføjet information om krs endpoint url | Trifork |
| 1.5 | 2013-05-21 | Opdateret det gode cpr opslag med wsdl og endpoints | Trifork |

# Bilag A: Eksempel på kald til abonnementsservicen

Nedenfor ses et eksempel på et kald til abonnementsservicen og det tilhørende svar. Eksemplet indeholder komplette headers, inklusiv IDkort, dog er signaturen og certifikatet erstattet med ”[...]”.

**Forespørgsel:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="<http://schemas.xmlsoap.org/soap/envelope/>">

  <S:Header>

    <ns2:Security xmlns:ns10="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/05/13/>" xmlns:ns11="<http://rep.oio.dk/xkom.dk/xml/schemas/2006/01/06/>" xmlns:ns12="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/09/01/>" xmlns:ns13="<http://rep.oio.dk/ois.dk/xml/schemas/2006/04/25/>" xmlns:ns14="<http://rep.oio.dk/medcom.sundcom.dk/xml/schemas/2007/02/01/>" xmlns:ns15="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/09/01/>" xmlns:ns16="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/11/24/>" xmlns:ns17="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" xmlns:ns18="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>" xmlns:ns19="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>" xmlns:ns2="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>" xmlns:ns20="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/01/17/>" xmlns:ns21="<http://rep.oio.dk/itst.dk/xml/schemas/2005/02/22/>" xmlns:ns22="<http://nsi.dk/cprabbs/2011/10>" xmlns:ns23="<http://nsi.dk/2011/09/23/StamdataCpr/>" xmlns:ns3="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns4="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2003/02/13/>" xmlns:ns5="<http://rep.oio.dk/itst.dk/xml/schemas/2006/01/17/>" xmlns:ns6="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/>" xmlns:ns7="<http://rep.oio.dk/itst.dk/xml/schemas/2005/06/24/>" xmlns:ns8="<http://rep.oio.dk/xkom.dk/xml/schemas/2005/03/15/>" xmlns:ns9="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/03/15/>">

      <wsu:Timestamp xmlns:wsu="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>">

        <wsu:Created>2011-12-08T09:10:31Z</wsu:Created>

      </wsu:Timestamp>

      <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion" IssueInstant="2011-12-08T09:05:31Z" Version="2.0" id="IDCard">

        <saml:Issuer>TheSOSILibrary</saml:Issuer>

        <saml:Subject>

          <saml:NameID Format="medcom:other">123</saml:NameID>

          <saml:SubjectConfirmation>

            <saml:ConfirmationMethod>urn:oasis:names:tc:SAML:2.0:cm:holder-of-key</saml:ConfirmationMethod>

            <saml:SubjectConfirmationData>

              <ds:KeyInfo xmlns:ds="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)">

                <ds:KeyName>OCESSignature</ds:KeyName>

              </ds:KeyInfo>

            </saml:SubjectConfirmationData>

          </saml:SubjectConfirmation>

        </saml:Subject>

        <saml:Conditions NotBefore="2011-12-08T09:05:31Z" NotOnOrAfter="2011-12-09T09:05:31Z"/>

        <saml:AttributeStatement id="IDCardData">

          <saml:Attribute Name="sosi:IDCardID">

            <saml:AttributeValue>BWMWK+OPPwTlBMxLxucS6w==</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="sosi:IDCardVersion">

            <saml:AttributeValue>1.0.1</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="sosi:IDCardType">

            <saml:AttributeValue>system</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="sosi:AuthenticationLevel">

            <saml:AttributeValue>3</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="sosi:OCESCertHash">

            <saml:AttributeValue>pFQXTvhBFAsCwPy2VkYI6GDTnFQ=</saml:AttributeValue>

          </saml:Attribute>

        </saml:AttributeStatement>

        <saml:AttributeStatement id="SystemLog">

          <saml:Attribute Name="medcom:ITSystemName">

            <saml:AttributeValue>ACME Pro</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="medcom:CareProviderID" NameFormat="medcom:cvrnumber">

            <saml:AttributeValue>12345678</saml:AttributeValue>

          </saml:Attribute>

          <saml:Attribute Name="medcom:CareProviderName">

            <saml:AttributeValue>dk</saml:AttributeValue>

          </saml:Attribute>

        </saml:AttributeStatement>

        <ds:Signature xmlns:ds="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" id="OCESSignature">

          <ds:SignedInfo>

            <ds:CanonicalizationMethod Algorithm="[http://www.w3.org/2001/10/xml-exc-c14n#](http://www.w3.org/2001/10/xml-exc-c14n)"/>

            <ds:SignatureMethod Algorithm="[http://www.w3.org/2000/09/xmldsig#rsa-sha1](http://www.w3.org/2000/09/xmldsig" \l "rsa-sha1)"/>

            <ds:Reference URI="#IDCard">

              <ds:Transforms>

                <ds:Transform Algorithm="[http://www.w3.org/2000/09/xmldsig#enveloped-signature](http://www.w3.org/2000/09/xmldsig" \l "enveloped-signature)"/>

                <ds:Transform Algorithm="[http://www.w3.org/2001/10/xml-exc-c14n#](http://www.w3.org/2001/10/xml-exc-c14n)"/>

              </ds:Transforms>

              <ds:DigestMethod Algorithm="[http://www.w3.org/2000/09/xmldsig#sha1](http://www.w3.org/2000/09/xmldsig" \l "sha1)"/>

              <ds:DigestValue>IVexeSZomox+Mq9cNYlSp1L+GX8=</ds:DigestValue>

            </ds:Reference>

          </ds:SignedInfo>

          <ds:SignatureValue>[...]</ds:SignatureValue>

          <ds:KeyInfo>

            <ds:X509Data>

              <ds:X509Certificate>[...]</ds:X509Certificate>

            </ds:X509Data>

          </ds:KeyInfo>

        </ds:Signature>

      </saml:Assertion>

    </ns2:Security>

    <ns18:Header xmlns:ns10="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/05/13/>" xmlns:ns11="<http://rep.oio.dk/xkom.dk/xml/schemas/2006/01/06/>" xmlns:ns12="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/09/01/>" xmlns:ns13="<http://rep.oio.dk/ois.dk/xml/schemas/2006/04/25/>" xmlns:ns14="<http://rep.oio.dk/medcom.sundcom.dk/xml/schemas/2007/02/01/>" xmlns:ns15="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/09/01/>" xmlns:ns16="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/11/24/>" xmlns:ns17="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" xmlns:ns18="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>" xmlns:ns19="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>" xmlns:ns2="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>" xmlns:ns20="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/01/17/>" xmlns:ns21="<http://rep.oio.dk/itst.dk/xml/schemas/2005/02/22/>" xmlns:ns22="<http://nsi.dk/cprabbs/2011/10>" xmlns:ns23="<http://nsi.dk/2011/09/23/StamdataCpr/>" xmlns:ns3="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns4="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2003/02/13/>" xmlns:ns5="<http://rep.oio.dk/itst.dk/xml/schemas/2006/01/17/>" xmlns:ns6="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/>" xmlns:ns7="<http://rep.oio.dk/itst.dk/xml/schemas/2005/06/24/>" xmlns:ns8="<http://rep.oio.dk/xkom.dk/xml/schemas/2005/03/15/>" xmlns:ns9="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/03/15/>">

      <medcom:SecurityLevel xmlns:medcom="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>">3</medcom:SecurityLevel>

      <medcom:Linking xmlns:medcom="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>">

        <medcom:FlowID>ccfd937c-91bb-4d7e-8772-ea72573cc96c</medcom:FlowID>

        <medcom:MessageID>42foobar</medcom:MessageID>

      </medcom:Linking>

      <medcom:RequireNonRepudiationReceipt xmlns:medcom="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>">no</medcom:RequireNonRepudiationReceipt>

    </ns18:Header>

  </S:Header>

  <S:Body>

    <ns22:CprAbbsRequest/>

  </S:Body>

</S:Envelope>

**Svar:**

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="<http://schemas.xmlsoap.org/soap/envelope/>">

  <S:Header>

    <ns3:Security xmlns:ns2="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" xmlns:ns3="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>" xmlns:ns4="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns5="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2003/02/13/>" xmlns:ns6="<http://rep.oio.dk/itst.dk/xml/schemas/2006/01/17/>" xmlns:ns7="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/>" xmlns:ns8="<http://rep.oio.dk/itst.dk/xml/schemas/2005/06/24/>" xmlns:ns9="<http://rep.oio.dk/xkom.dk/xml/schemas/2005/03/15/>" xmlns:ns10="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/03/15/>" xmlns:ns11="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/05/13/>" xmlns:ns12="<http://rep.oio.dk/xkom.dk/xml/schemas/2006/01/06/>" xmlns:ns13="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/09/01/>" xmlns:ns14="<http://rep.oio.dk/ois.dk/xml/schemas/2006/04/25/>" xmlns:ns15="<http://rep.oio.dk/medcom.sundcom.dk/xml/schemas/2007/02/01/>" xmlns:ns16="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/09/01/>" xmlns:ns17="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/11/24/>" xmlns:ns18="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>" xmlns:ns19="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>" xmlns:ns20="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/01/17/>" xmlns:ns21="<http://rep.oio.dk/itst.dk/xml/schemas/2005/02/22/>" xmlns:ns22="<http://nsi.dk/cprabbs/2011/10>" xmlns:ns23="<http://nsi.dk/2011/09/23/StamdataCpr/>">

      <ns19:Timestamp>

        <ns19:Created>2011-12-08T09:10:32Z</ns19:Created>

      </ns19:Timestamp>

    </ns3:Security>

    <ns18:Header xmlns:ns2="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" xmlns:ns3="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>" xmlns:ns4="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns5="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2003/02/13/>" xmlns:ns6="<http://rep.oio.dk/itst.dk/xml/schemas/2006/01/17/>" xmlns:ns7="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/>" xmlns:ns8="<http://rep.oio.dk/itst.dk/xml/schemas/2005/06/24/>" xmlns:ns9="<http://rep.oio.dk/xkom.dk/xml/schemas/2005/03/15/>" xmlns:ns10="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/03/15/>" xmlns:ns11="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/05/13/>" xmlns:ns12="<http://rep.oio.dk/xkom.dk/xml/schemas/2006/01/06/>" xmlns:ns13="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/09/01/>" xmlns:ns14="<http://rep.oio.dk/ois.dk/xml/schemas/2006/04/25/>" xmlns:ns15="<http://rep.oio.dk/medcom.sundcom.dk/xml/schemas/2007/02/01/>" xmlns:ns16="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/09/01/>" xmlns:ns17="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/11/24/>" xmlns:ns18="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>" xmlns:ns19="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>" xmlns:ns20="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/01/17/>" xmlns:ns21="<http://rep.oio.dk/itst.dk/xml/schemas/2005/02/22/>" xmlns:ns22="<http://nsi.dk/cprabbs/2011/10>" xmlns:ns23="<http://nsi.dk/2011/09/23/StamdataCpr/>">

      <ns18:Linking>

        <ns18:FlowID>ccfd937c-91bb-4d7e-8772-ea72573cc96c</ns18:FlowID>

        <ns18:MessageID>743e57ca-f3c2-462a-9fd3-f54e8414dfbd</ns18:MessageID>

        <ns18:InResponseToMessageID>42foobar</ns18:InResponseToMessageID>

      </ns18:Linking>

      <ns18:FlowStatus>flow\_finalized\_succesfully</ns18:FlowStatus>

    </ns18:Header>

  </S:Header>

  <S:Body>

    <ns23:PersonLookupResponse xmlns:ns2="[http://www.w3.org/2000/09/xmldsig#](http://www.w3.org/2000/09/xmldsig)" xmlns:ns3="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>" xmlns:ns4="urn:oasis:names:tc:SAML:2.0:assertion" xmlns:ns5="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2003/02/13/>" xmlns:ns6="<http://rep.oio.dk/itst.dk/xml/schemas/2006/01/17/>" xmlns:ns7="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/03/18/>" xmlns:ns8="<http://rep.oio.dk/itst.dk/xml/schemas/2005/06/24/>" xmlns:ns9="<http://rep.oio.dk/xkom.dk/xml/schemas/2005/03/15/>" xmlns:ns10="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/03/15/>" xmlns:ns11="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/05/13/>" xmlns:ns12="<http://rep.oio.dk/xkom.dk/xml/schemas/2006/01/06/>" xmlns:ns13="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/09/01/>" xmlns:ns14="<http://rep.oio.dk/ois.dk/xml/schemas/2006/04/25/>" xmlns:ns15="<http://rep.oio.dk/medcom.sundcom.dk/xml/schemas/2007/02/01/>" xmlns:ns16="<http://rep.oio.dk/ebxml/xml/schemas/dkcc/2005/09/01/>" xmlns:ns17="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2005/11/24/>" xmlns:ns18="<http://www.medcom.dk/dgws/2006/04/dgws-1.0.xsd>" xmlns:ns19="<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd>" xmlns:ns20="<http://rep.oio.dk/cpr.dk/xml/schemas/core/2006/01/17/>" xmlns:ns21="<http://rep.oio.dk/itst.dk/xml/schemas/2005/02/22/>" xmlns:ns22="<http://nsi.dk/cprabbs/2011/10>" xmlns:ns23="<http://nsi.dk/2011/09/23/StamdataCpr/>">

      <ns15:PersonInformationStructure>

        <ns15:CurrentPersonCivilRegistrationIdentifier>0201270145</ns15:CurrentPersonCivilRegistrationIdentifier>

        <ns20:RegularCPRPerson>

          <ns20:SimpleCPRPerson>

            <ns6:PersonNameStructure>

              <ns5:PersonGivenName>Peter</ns5:PersonGivenName>

              <ns5:PersonMiddleName>Sigurd</ns5:PersonMiddleName>

              <ns5:PersonSurnameName>Andersen</ns5:PersonSurnameName>

            </ns6:PersonNameStructure>

            <ns7:PersonCivilRegistrationIdentifier>0101822231</ns7:PersonCivilRegistrationIdentifier>

          </ns20:SimpleCPRPerson>

          <ns21:PersonNameForAddressingName>Peter,Andersen</ns21:PersonNameForAddressingName>

          <ns5:PersonGenderCode>male</ns5:PersonGenderCode>

          <ns17:PersonInformationProtectionIndicator>false</ns17:PersonInformationProtectionIndicator>

          <ns17:PersonBirthDateStructure>

            <ns10:BirthDate>2011-12-06+01:00</ns10:BirthDate>

            <ns17:BirthDateUncertaintyIndicator>false</ns17:BirthDateUncertaintyIndicator>

          </ns17:PersonBirthDateStructure>

          <ns20:PersonCivilRegistrationStatusStructure>

            <ns17:PersonCivilRegistrationStatusCode>1</ns17:PersonCivilRegistrationStatusCode>

            <ns20:PersonCivilRegistrationStatusStartDate>2011-12-07+01:00</ns20:PersonCivilRegistrationStatusStartDate>

          </ns20:PersonCivilRegistrationStatusStructure>

        </ns20:RegularCPRPerson>

        <ns15:PersonAddressStructure>

          <ns8:CareOfName>Søren Petersen</ns8:CareOfName>

          <ns12:AddressComplete>

            <ns9:AddressAccess>

              <ns7:MunicipalityCode>0461</ns7:MunicipalityCode>

              <ns7:StreetCode>0234</ns7:StreetCode>

              <ns5:StreetBuildingIdentifier>10A</ns5:StreetBuildingIdentifier>

            </ns9:AddressAccess>

            <ns12:AddressPostal>

              <ns10:StreetName>Ørstedgade</ns10:StreetName>

              <ns7:StreetNameForAddressingName>Østergd.</ns7:StreetNameForAddressingName>

              <ns5:StreetBuildingIdentifier>10A</ns5:StreetBuildingIdentifier>

              <ns5:FloorIdentifier>12</ns5:FloorIdentifier>

              <ns5:SuiteIdentifier>tv</ns5:SuiteIdentifier>

              <ns10:DistrictSubdivisionIdentifier>Birkely</ns10:DistrictSubdivisionIdentifier>

              <ns10:PostCodeIdentifier>6666</ns10:PostCodeIdentifier>

              <ns10:DistrictName>Überwald</ns10:DistrictName>

            </ns12:AddressPostal>

          </ns12:AddressComplete>

          <ns14:CountyCode>1083</ns14:CountyCode>

        </ns15:PersonAddressStructure>

      </ns15:PersonInformationStructure>

      <ns15:PersonInformationStructure>

        <ns15:CurrentPersonCivilRegistrationIdentifier>1302642907</ns15:CurrentPersonCivilRegistrationIdentifier>

        <ns20:RegularCPRPerson>

          <ns20:SimpleCPRPerson>

            <ns6:PersonNameStructure>

              <ns5:PersonGivenName>Peter</ns5:PersonGivenName>

              <ns5:PersonMiddleName>Sigurd</ns5:PersonMiddleName>

              <ns5:PersonSurnameName>Andersen</ns5:PersonSurnameName>

            </ns6:PersonNameStructure>

            <ns7:PersonCivilRegistrationIdentifier>0101821234</ns7:PersonCivilRegistrationIdentifier>

          </ns20:SimpleCPRPerson>

          <ns21:PersonNameForAddressingName>Peter,Andersen</ns21:PersonNameForAddressingName>

          <ns5:PersonGenderCode>male</ns5:PersonGenderCode>

          <ns17:PersonInformationProtectionIndicator>false</ns17:PersonInformationProtectionIndicator>

          <ns17:PersonBirthDateStructure>

            <ns10:BirthDate>2011-12-06+01:00</ns10:BirthDate>

            <ns17:BirthDateUncertaintyIndicator>false</ns17:BirthDateUncertaintyIndicator>

          </ns17:PersonBirthDateStructure>

          <ns20:PersonCivilRegistrationStatusStructure>

            <ns17:PersonCivilRegistrationStatusCode>1</ns17:PersonCivilRegistrationStatusCode>

            <ns20:PersonCivilRegistrationStatusStartDate>2011-12-07+01:00</ns20:PersonCivilRegistrationStatusStartDate>

          </ns20:PersonCivilRegistrationStatusStructure>

        </ns20:RegularCPRPerson>

        <ns15:PersonAddressStructure>

          <ns8:CareOfName>Søren Petersen</ns8:CareOfName>

          <ns12:AddressComplete>

            <ns9:AddressAccess>

              <ns7:MunicipalityCode>0461</ns7:MunicipalityCode>

              <ns7:StreetCode>0234</ns7:StreetCode>

              <ns5:StreetBuildingIdentifier>10A</ns5:StreetBuildingIdentifier>

            </ns9:AddressAccess>

            <ns12:AddressPostal>

              <ns10:StreetName>Ørstedgade</ns10:StreetName>

              <ns7:StreetNameForAddressingName>Østergd.</ns7:StreetNameForAddressingName>

              <ns5:StreetBuildingIdentifier>10A</ns5:StreetBuildingIdentifier>

              <ns5:FloorIdentifier>12</ns5:FloorIdentifier>

              <ns5:SuiteIdentifier>tv</ns5:SuiteIdentifier>

              <ns10:DistrictSubdivisionIdentifier>Birkely</ns10:DistrictSubdivisionIdentifier>

              <ns10:PostCodeIdentifier>6666</ns10:PostCodeIdentifier>

              <ns10:DistrictName>Überwald</ns10:DistrictName>

            </ns12:AddressPostal>

          </ns12:AddressComplete>

          <ns14:CountyCode>1083</ns14:CountyCode>

        </ns15:PersonAddressStructure>

      </ns15:PersonInformationStructure>

    </ns23:PersonLookupResponse>

  </S:Body>

</S:Envelope>

1. Det er forventningen, at det er i nær fremtid bliver muligt at adgangsstyre på system-niveau. [↑](#footnote-ref-1)
2. Servicen giver et øjebliksbillede af personens autorisationer, dvs. udløbne og fremtidige autorisationer returneres ikke. [↑](#footnote-ref-2)