Datafordeler

Dataleverancespecifikation

Implementeringsdetaljer

CPR



# Versions historik

|  |  |  |
| --- | --- | --- |
| **Dato** | **Version** | **Bemærkning** |
| 04-04-2017 | 1.0 | Første version af dokumentet |
| 02-05-2015 | 1.1 | Tilrettet version af dokument med yderligere detaljer |

Indholdsfortegnelse

[Versions historik 2](#_Toc485037814)

[Introduktion 4](#_Toc485037815)

[Om dette dokument 4](#_Toc485037816)

[Implementerings detaljer 5](#_Toc485037817)

[Diverse 5](#_Toc485037818)

[Historik 5](#_Toc485037819)

[Civilstand 5](#_Toc485037820)

[Personnummer 8](#_Toc485037821)

[Implementeringsdetaljer specifikt gældende for Private tjenester 8](#_Toc485037822)

[Generelt 8](#_Toc485037823)

[Adresser på døde personer 8](#_Toc485037824)

[Søgning på navne 8](#_Toc485037825)

[Implementeringsdetaljer specifikt gældende for Offentlige tjenester 9](#_Toc485037826)

[Implementeringsdetaljer gældende specifikt for RobinsonListeDownload filudtræk 10](#_Toc485037827)

# Introduktion

### Om dette dokument

Dette dokument udvider de logiske beskrivelser i de øvrige bilag, især bilag 2, med mere tekniske forklaringer af potentielt særligt komplekse aspekter. Herudover så kan dette dokument også indeholde eksempler på, hvordan en fysisk implementering af logikken fra de andre bilag kan realiseres i henhold til de logiske beskrivelser og de fysiske data.

# Implementerings detaljer

## Diverse

- Feltet foraeldremyndighedshaverrolle kan indeholde værdierne ’mor’, ’far\_medmor’ og ’anden’

- Tabellen MorFar indeholder forældre til personen, hvis disse er kendt. En MorFar række blive til to Foraeldre strukturer hvis begge forældre er kendt.

- Ved join med MorFar tabellen og afgrænsning på virkningfra skal oplysningen med såfremt blot én af de to virkningfra datoer opfylder søgekriteriet (virkningfrafar/virkningframor)

- Søgning med adresselinie parameteren skal give hit hvis bare en af de fem adresselinier giver hit.

## Historik

Metoderne i CPRPerson har historik, medmindre der indgår ordet aktuel i navnet på metoden. Alle offentlige har dog altid fuld historik på pnr.

Der er visse undtagelers for historik, det gælder når der er referencer til andre personer, så som ægtefælde, forældre m.m. I disse tilfælde vil det være den aktuelle person der returneres.

## Civilstand

Civilstand er specielt kompliceret og for at tilsikre forståelse af logikken er der medtaget nedenstånde forklaring af begrebet samt praktisk anvendelse i CPR systemet.

En civilstand i CPR systemet dækker ikke kun over det man først tænker over nemlig at man er GIFT eller FRASKILT, men dækker også en række andre tilstande såsom UGIFT , REGISTRERET PARTNER, DOED, så der er således ikke nødvendigvis en anden part med i en civilstand.

Civilstande har en livscykles som hedder : UGIFT -> (GIFT, FRASKILT, REGISTRERET PARTNER, OPHØRT REGISTRERET PARTNERSKAB ) -> DOED

Man kan dermed selvfølgelig gå fra UGIFT til DOED uden at have nogle af de andre status.

Er man GIFT eller REGISTRERET PARTNER kan man så også være SEPARERET som ikke er civilstandsstatus men en tilstand på netop disse to status.

I de status hvor der forekommer 2 parter : GIFT og REGISTRERET PARTNER vil der kunne forekomme et CIVILSTANDSID. Dette CIVILSTANDSID vil forekomme en eller 2 gange afhængig om man i CPR kender den anden part.

Hvis man er gift vil der således være to civilstande i systemet, en set fra den ene parts side og en fra den anden parts side.

CIVILSTANDSID er et timestamp, så der vil godt kunne forekomme flere end 2 af det civilstandsid, og derfor kan man ikke nøjes med dette ID for at finde parterne.

CIVILSTANDEN skal altid ses i konteksten af personen man kigger på.

I det XML der sendes over til datafordeleren kommer der altid en hel person ad gangen, og hvis denne person f.eks. er GIFT med en anden person, vil der komme en Civilstand med over hvor PersonID er netop denne person, hvis den anden part i forholdet er kendt i CPR vil der også være udfyldt et ægtefælle ID som er PersonID’et på ægtefællen.

Det betyder, at når man fremsøger en person i datafordeleren, så skal den civilstand der vises være fra denne persons side og ikke partnerens side. Man joiner således Personid fra personen der vises med personid fra civilstand.

Det betyder så at når man fremsøger en person i datafordeleren, så skal vise civilstanden set fra den person man viser og altså fra partnerens side, så man skal have Personid med i joinet op mod civilstand svarende til den person der vises. Her er CIVILSTANDSID i virkelighed irrelevant, man skal bare vise de civilstande der passer på Personid.

Separation leveres på samme måde som civilstand til datafordeleren, i form af en separation pr. part ( hvis parten altså er kendt i systemet )

Når man kigger på separationer så joines de på både civilstandsid og på persond, så man viser så at sige den person der vises Separation til en given civilstand.

Herunder kommer eksempel på XML som der sendes for 2 forskellige personer som er gift med hinanden og separeret: (demo data)

<cmn:DataEntity><cmn:Action>Create</cmn:Action><cmn:CreateEvent>CreateEvent</cmn:CreateEvent><cpr:Civilstand><cpr:aegtefaelle>0820e44b-bf00-4c75-a6ba-090b3b437a28</cpr:aegtefaelle><cpr:aegtefaellefoedselsdato xsi:nil="true"/><cpr:aegtefaellefoedselsdatousikkerhedsmarkering xsi:nil="true"/><cpr:aegtefaellenavn xsi:nil="true"/><cpr:aegtefaellenavnemarkering xsi:nil="true"/><cpr:civilstandsid>**201408121241**</cpr:civilstandsid><cpr:civilstandstype>gift</cpr:civilstandstype><cpr:personid>0489129f-1955-4f99-9cd0-f485d701f571</cpr:personid><cpr:status>aktuel</cpr:status><cpr:virkningfra>2005-02-02T06:48:00.000+01:00</cpr:virkningfra><cpr:virkningfrausikkerhedsmarkering>false</cpr:virkningfrausikkerhedsmarkering><cpr:virkningtil xsi:nil="true"/><cpr:virkningtilusikkerhedsmarkering xsi:nil="true"/></cpr:Civilstand></cmn:DataEntity>

<cmn:DataEntity><cmn:Action>Create</cmn:Action><cmn:CreateEvent>CreateEvent</cmn:CreateEvent><cpr:Separation><cpr:civilstandsid>201408121241</cpr:civilstandsid><cpr:personid>0489129f-1955-4f99-9cd0-f485d701f571</cpr:personid><cpr:status>aktuel</cpr:status><cpr:virkningfra>2014-08-01T00:00:00.000+02:00</cpr:virkningfra><cpr:virkningfrausikkerhedsmarkering>false</cpr:virkningfrausikkerhedsmarkering><cpr:virkningtil xsi:nil="true"/><cpr:virkningtilusikkerhedsmarkering xsi:nil="true"/></cpr:Separation></cmn:DataEntity>

Og her den anden person

<cmn:DataEntity><cmn:Action>Create</cmn:Action><cmn:CreateEvent>CreateEvent</cmn:CreateEvent><cpr:Civilstand><cpr:aegtefaelle>0489129f-1955-4f99-9cd0-f485d701f571</cpr:aegtefaelle><cpr:aegtefaellefoedselsdato xsi:nil="true"/><cpr:aegtefaellefoedselsdatousikkerhedsmarkering xsi:nil="true"/><cpr:aegtefaellenavn xsi:nil="true"/><cpr:aegtefaellenavnemarkering xsi:nil="true"/><cpr:civilstandsid>**201408121241**</cpr:civilstandsid><cpr:civilstandstype>gift</cpr:civilstandstype><cpr:personid>0820e44b-bf00-4c75-a6ba-090b3b437a28</cpr:personid><cpr:status>aktuel</cpr:status><cpr:virkningfra>2005-02-02T06:48:00.000+01:00</cpr:virkningfra><cpr:virkningfrausikkerhedsmarkering>false</cpr:virkningfrausikkerhedsmarkering><cpr:virkningtil xsi:nil="true"/><cpr:virkningtilusikkerhedsmarkering xsi:nil="true"/></cpr:Civilstand></cmn:DataEntity>

<cmn:DataEntity><cmn:Action>Create</cmn:Action><cmn:CreateEvent>CreateEvent</cmn:CreateEvent><cpr:Separation><cpr:civilstandsid>**201408121241**</cpr:civilstandsid><cpr:personid>0820e44b-bf00-4c75-a6ba-090b3b437a28</cpr:personid><cpr:status>aktuel</cpr:status><cpr:virkningfra>2014-08-01T00:00:00.000+02:00</cpr:virkningfra><cpr:virkningfrausikkerhedsmarkering>false</cpr:virkningfrausikkerhedsmarkering><cpr:virkningtil xsi:nil="true"/><cpr:virkningtilusikkerhedsmarkering xsi:nil="true"/></cpr:Separation></cmn:DataEntity>

Læg mærke til at i første XML udpeges partneren med aegtefaelle og der er et civilstandsid med som identificerer civilstanden.

Har man således fremsøgt den første person, så vil man join civilstand ved at sige :

SELECT \* FROM PERSON JOIN CIVILSTAND ON PERSON.PERSONID = CIVILSTAND.PERSONID

Så får man altid kun vist civilstanden set fra denne persons syn, man er ikke interesseret i civilstanden set fra den andens persons synspunkt

Tilsvarende får man de tilhørende separationer ud ved :

SELECT \* FROM PERSON JOIN CIVILSTAND ON PERSON.PERSONID = CIVILSTAND.PERSONID

                                                  JOIN SEPARATION ON PERSON.PERSONID = SEPARATION.PERSONID AND CIVILSTAND.CIVILSTANDSID = SEPARATION.CIVILSTANDSID

Når man så skal vise oplysninger om den anden part, så joiner man videre på AEGTEFAELLE for at finde data om denne person.

Så for at finde data om ægtefællen:

SELECT \* FROM PERSON JOIN CIVILSTAND ON PERSON.PERSOND = CIVILSTAND.AEGTEFAELLE

En person kan jo sagtens IKKE have en anden part hvis man f.eks. er UGIFT eller DOED, FRASKILT osv. og der skal så ikke joines videre. Også hvis der reelt ikke er en AEGTEFAELLE id, det kan godt forekomme selvom man er gift, det er udtryk for at det ikke er en person der er registreret i CPR man er gift med.

Status UGIFT findes ikke som historisk da denne status altid slettes første gang en person skifter civilstand.

## Personnummer

En person kan have flere personnumre, hvoraf det aktuelle personnummer altid vil have status ’aktuel’ og de resterende vil have status ’historisk’.

De offentlige tjenester har altid samtlige personnumre med ud i output, hvor de private tjeneste udelukkende har det aktuelle personnummer med ud i output.

## Implementeringsdetaljer specifikt gældende for Private tjenester

### Generelt

Generelt må private tjenester udelukkende udstille aktuelle data, men der skal altid søges både i aktuelle og historiske data.

Der må aldrig søges på personer der har status = ’nedlagt’.

Adresser, UdrejseIndrejse og Navn må kun komme med i output såfremt der ikke er forefindes en aktuel navn\_og\_adresse beskyttelse på personen der forespørges på.

### Adresser på døde personer

Døde personer har aldrig en aktuel adresse, udrejseindrejse eller forsvinding, men personens seneste historiske adresse eller udrejse/indrejse skal med i output såfremt personen ikke har navne og adressebeskyttelse på forespørgselstidspunktet.

Forsvindinger må altid komme med i output.

KontaktAdresse skal altid med ud for personer som er døde, udrejst eller forsvundet, såfremt denne er til stede.

### Søgning på navne

Man skal kunne søge på aktuelle og historiske navnedele i tjenesten, men man kan kun søge på en enkelt navnedel ud over efternavn.

Man skal angive efternavn.

Når man søger i navnedele søger man både i fornavne og mellemnavn.

Navnet der søges på skal matche et af navnene fra hhv fornavne eller mellemnavn

Der skal kunne søges historisk på tværs af navne.

Navnedelene deles op med mellemrumstegn

Eksempelvis en person der har følgende navne over tid :

|  |  |  |  |
| --- | --- | --- | --- |
| Fornavne | Mellemnavne | Efternavn | Dato |
| Svend Peter | Hans | Hansen | 09-01-2016 |
| Lars Henrik Poul |  | Nielsen | 09-01-2014 |
| Thorbjørn-Erik | Palle | Poulsen | 09-01-2013 |
| Peter Niels | Kalle | Karlsson | 09-01-2014 |

Søgning

|  |  |  |
| --- | --- | --- |
| Fornavne / Mellemnavne | Efternavn | Hit |
| Peter | Karlsson | √ |
| Svend | Karlsson | √ |
| Thorbjørn | Poulsen | ÷ |
| Svend Peter | Hansen | ÷ |

Foretages der søgning fornavn: Peter og efternavn Karlsson, får man hit fordi person hedder Peter og Karlsson, man får også hit når man søger: fornavn: Svend og efternavn: Karlsson da der er match historisk på navnedele og efternavn uagtet at personen ikke hed dette samlet på noget tidspunkt.

Søger man Fornavn: Thorbjørn og efternavn Poulsen, får man intet hit, da personen hedder Thorbjørn-Erik

Søger man Fornavn: Svend Peter og efternavn: Hansen, får man ligeledes ikke noget hit da personens navnedele ikke matcher Svend Peter, men Svend og Peter hver for sig.

Navne må kun komme ud i output såfremt der ikke er en aktuel navne\_og\_adresse beskyttelse.

## Implementeringsdetaljer specifikt gældende for Offentlige tjenester

Offentlige tjenester er inddelt i 3 kategorier af output: Small, Medium og Full.

Disse kategorier afgør mængden af data der kommer ud omkring en person.

Hver kategori kan yderligere inddeles med en metode der kun returnerer aktuelle data (current) og en metode der returnerer aktuelle og historiske data.

Small tjenesterne returnerer kun basis oplysninger om personen og ingen oplysninger om personer som denne person refererer til.

Medium tjenesterne returnerer de samme data som Small tjenesterne, men ydermere også referencer til personer som personen refererer til. Personens: Ægtefæller, Forældre, Forældremyndige samt Børn. Tjenesterne returner PNR og PersonID for reference personerne såfremt de er til stede. Ægtefæller og forældre kan også returneres som PersonUdenCpr, jvnf. Json-schema

Full tjenesterne returnerer de samme data som Medium tjenesterne, men ud over at returnere PNR og PersonId for reference personerne såfremt de er til stede, returneres også data om person referencens aktuelle navn og ophold: UdrejseIndrejse, Adresseoplysninger, Forsvinding samt KontaktAdresse.

## Implementeringsdetaljer gældende specifikt for RobinsonListeDownload filudtræk

Robinsonlisten udstiller udelukkende en række simple oplysninger om personer som har beskyttelse mod markedsføring.

Der udstilles således ikke personnummer lister, eller fulde navne eller fulde adresser inkl. Virkningsperioder.

Robinsonlisten udstiller alle personer som ikke har status : nedlagt, annulleret eller ikke har en bopæl

Der skal altid udstilles den aktuelle adresse med mindre der er tale om en person med status forsvundet, udrejst eller død. I disse tilfælde udstilles altid sidst kendte danske adresse, men mindre personens status er ældre end 3 år fra forespørgselstidspunktet. Hvis personens status er ældre end dette udstilles personen ikke i filudtrækket.

Det er altid personen aktuelle navn der udstilles.