

Dokumentlog

Dato	Version	Beskrivelse	Applikation version	Reference	Forfatter	Godkender
2015.11.30	1.0	Ny godkendelses-proces	MDS 4.1	K15 Transition	Oprydning CPRDOK	Godkendt af leverandør ifb. med Transition
2020.01.02	2.0	Ref til CPR adresse mv fjernet		SER-3762	Søren Bækdal	Karin Schøtt



**Håndbog
Til
CPR services**

**Bilag 6
Anvendelse af
CPR Søger-
services,
programme-
ringsvejledning**

CPR-kontoret
Hjemmeside: www.cpr.dk

Indholdsfortegnelse

1. Indledning	4
2. Forudsætninger for læseren	4
3. Forudsætninger før kald af services	4
4. Sikkerhed ved søgninger	4
5. Generel beskrivelse af CPR-søgeservices	5
6. Simple opslag	5
7. Opslag med flere rækker	5
8. Begrænsede søgninger	6
9. Brede søgninger	6
10. Eksempel 1: Vis nuværende adresse (simpelt opslag)	7
11. Eksempel 2: Vis personens nuværende og tidligere adresser (opslag med liste)	9
12. Eksempel 3: Myndighedssøgning (begrænset søgning)	12
13. Eksempel 4: Fødselsdatosøgning (bred søgning)	14

1. Indledning

2. Forudsætninger for læseren

For at forstå eksemplerne er det en fordel at have læst GCTP-standard

Standard vedrørende Generel GCTP format

GCTP er den standard for kommunikation mellem server og client som CPR har valgt at bruge, blandt andet fordi denne sikrer en dynamisk grænseflade mellem server og client.

3. Forudsætninger før kald af services

Før man kan begynde at kalde CPR-services skal brugeren have en fysisk forbindelse til CPR og være logget på CPR-systemet som beskrevet i

bilag 4 : Etablering af netforbindelse til CPR og

bilag 5 : Etablering af forbindelse til og logon på CPR-serveren.

I eksemplerne beskrives kun GCTP-delen af kommunikationen, ikke eventuelle HTTP-headere og lignende. Dette anses for at være en del af transport-laget, som er irrelevant for den egentlige applikation.

Det bemærkes også, at der i dette dokument ikke er vist XML-header, XML-commandstatements og blokken der henviser til XML-namespace. De fremgår af bilag 5 om logon og generel brug af CPR-services.

Alle søgninger hører GCTP-mæssigt til systemet CprSoeg.

Søgninger er beskrevet i en **Servicespecifikation**, som beskriver søgningen, hvilke nøgler klienten skal/kan sende til serveren og hvilke data serveren returnerer.

4. Sikkerhed ved søgninger

Adgangen til at udføre søgninger i CPR er styret af den enkelte brugerkode.

Forskellige personer i samme myndighed kan have forskellige adgange (profiler) til at søge i CPR.

5. Generel beskrivelse af CPR-søgeservices

Som tidligere beskrevet er der to typer CPR-søgeservices : Opslag og søgninger.

Opslagene kan af natur opdeles i to undertyper :

1. Simple opslag med kun ét hit
F.eks. "Nuværende adresse"
(Visning af personens nuværende adresserække samt diverse markeringer)
2. Opslag med flere rækker
F.eks. "Tidligere adresser"
(Viser en liste over en persons nuværende og tidligere adresser)

De deciderede søgninger returnerer alle flere rækker - men også her er der fundamentalt to typer :

1. Begrænsede søgninger med kvalificerede nøgler eller på små datamængder
F.eks. "Myndigheds søgning" (søgning på kommunekoder og lignende)
2. Brede søgninger på forholdsvis store datamængder
F.eks. "Fødselsdatosøgning" eller "Navnesøgning"

De fire typer er beskrevet nærmere i det følgende.

6. Simple opslag

Ved simple opslag er kommunikationen ret enkel.

Af servicespecifikationen fremgår det tydeligt hvilke nøgler klienten skal kalde med og hvilke felter serveren sender retur.

Når klienten har fået data retur kan man glemme alt om opslaget.

Serveren har kontrolleret sikkerheden (har brugeren adgang til servicen) og har logget opslaget (denne bruger har slået op på denne/disse personer).

7. Opslag med flere rækker

Af performance hensyn sender CPR-serveren maksimalt 20 rækker retur til klienten ved denne slags opslag. Klienten kan, hvis det ønskes, sætte dette maksimum endnu længere ned (angives med nøglen MAXA). Det kan f.eks. bruges hvis man kun vil have vist personens forrige adresse (første tidligere adresse).

Hvis klienten har brug for det (angives med nøglen AIA) kan serveren fortælle hvor mange rækker der er i alt. Med denne viden kan klienten lave korrekt opsætning af scrollbars osv.

Hvis der er flere rækker end de returnerede, returnerer serveren også en "restart-række" der kun indeholder en nøgle som klienten kan kalde med for at få de næste rækker.

Klienten skal altså her huske at gemme restart-nøglen til evt. senere brug.

Af servicespecifikationen fremgår det tydeligt hvilke nøgler klienten skal kalde med og hvilke felter serveren sender retur. Denne type opslag returnerer i visse tilfælde nogle enkelt-felter til at starte med, f.eks. bopælsadressen ved "Vis alle beboere på personens bopæl".

8. Begrænsede søgninger

De begrænsede søgninger virker præcist som "Opslag med flere rækker", og er ofte blot "brede søgninger", men på en meget lille datamængde.

9. Brede søgninger

Ved brede søgninger er opslaget i databasen potentielt så tungt, at vi gemmer nøglerne til de enkelte rækker i et midlertidigt søgedatalager på serveren.

Ved en navnesøgning gemmes f.eks. alle personnumre på dem der opfylder søgekriterierne. Når/hvis klienten beder om at få de næste 20 personer på listen skal serveren blot hente nøglerne i søgedatalageret og læse personernes data.

Restart virker ellers helt som ved "Opslag med flere rækker" og "Begrænsede søgninger".

Nøglen til søgedatalageret er en del af restart-nøglen, således at serveren kan bruge denne til at læse resultatet af søgningen, hvis klienten kalder for at læse flere end de returnerede rækker.

Når klienten initierer en sådan søgning, returnerer serveren også nøglen til søgedatalageret som et enkelt-felt (feltreference "SEARCH_ID").

Når brugeren forlader listen **skal** klienten kalde en speciel CPR utility-service (SOEGDONE) for at få slettet søgedatalageret.

Bemærk at hvis det antal rækker der findes er mindre end det ønskede max antal (MAXA - evt. defaultet til 20) vil der **ikke** blive gemt et søgedatalager, da klienten jo allerede har fået alle data.

10. Eksempel 1: Vis nuværende adresse (simpelt opslag)

Servicen til at få vist aktuel bopæl er ADRESSE3 : Nuværende adresse

Af servicespecifikationen fremgår det at klienten SKAL angive et personnummer (PNR) og at klienten derudover med AKX-nøglen kan angive om man ønsker at få evt. fortrudte og rettede rækker med i opslaget.

I dette opslag angiver AKX om de fortrudte/rettede rækker skal medtages ved opsætningen af TIDLADMRK.

Hvis vi antager at klienten vil have de fortrudte/rettede rækker med i betragtning ved opsætningen af TIDLADMRK, så ser klientens GCTP til serveren sådan ud :

```
<Gctp v="1.0" ><System r="CprSoeg"><Service r="ADRESSE3"><CprServiceHeader r="ADRESSE3"><Key><Field r="PNR" v="12121212"/><Field r="AKX" v="X"/></Key></CprServiceHeader></Service></System></Gctp>
```

Lidt mere læsevenligt :

```
<Gctp v="1.0" >
  <System r="CprSoeg">
    <Service r="ADRESSE3">
      <CprServiceHeader r="ADRESSE3">
        <Key>
          <Field r="PNR" v="12121212"/>
          <Field r="AKX" v="X"/>
        </Key>
      </CprServiceHeader>
    </Service>
  </System>
</Gctp>
```

Bemærk at

- hvis klienten ikke ønsker at tag hensyn til eventuelle fortrudte/rettede rækker skal **<Field r="AKX" v="X"/>** blot udelades. (Ikke alle søgninger/opslag bruger AKX)
- nøglerne behøver ikke at stå i den rækkefølge hvori de er beskrevet i servicespecifikationen. Blot GCTP-strukturen følges, er de enkelte felters interne placering underordnet.
- Det her angivne personnummer (12121212) er angivet som eksempel. En rigtig søgning med dette personnummer vil resultere i en fejl. Alle nøgler valideres før søgningerne aktiveres og validereingen på PNR er blandt andet at dette skal findes i CPR.

Serverens svar vil, i vores tænkte eksempel, se sådan ud :

```
<Gctp v="1.0 env="Test" ><System r="CprSoeg"><Service r="ADRESSE3">
<CprServiceHeader r="ADRESSE3" ts="20010123201713711036"/><CprData u="O"><Rolle r="HovedRolle"><Field r="TIDLADMRK" v="X"/><Field r="TIDLKOMMRK" v="X"/><Field r="SUPADMRK" v="X"/><Field r="ADRKOD" v="1"/><Field r="TIMSTP" v="20000525142701794751"/><Field r="SYSTEM_TS" v="20000525141925862898"/><Field
```

```
r="ADRTS" v="20000525141953813707"/><Field r="MYNKOD" v="0201" t="Allerød"
tl="Allerød"/><Field r="VEJKOD" v="2073" t="Aarupvej" tm="Aarupvej"/><Field r="HUSNR"
v="002"/><Field r="ETAGE" v="st"/><Field r="SIDEDOER"/>
<Field r="TILFLYKOMDATO" v="199902020000"/><Field r="BOLIGTYPKOD" v="0" t="Ukendt
boligtype"/><Field r="POSTNR" v="3520" t="Farum"/><Field r="STARTDATO"
v="200005201414"/><Field r="KOMKOD" v="0207" t="Farum" tl="Farum"/><Field
r="FLYTPAMRK" v="X"/></Rolle></CprData></Service></System></Gctp>
```

Den læsevenlige version :

```
<Gctp v="1.0" env="Test" >
<System r="CprSoeg">
<Service r="ADRESSE3">
<CprServiceHeader r="ADRESSE3" ts="20010123201713711036"/>
<CprData u="O">
<Rolle r="HovedRolle">
<Field r="TIDLADRMRK" v="X"/>
<Field r="TIDLKOMMRK" v="X"/>
<Field r="SUPADRMRK" v="X"/>
<Field r="ADRKOD" v="1"/>
<Field r="TIMSTP" v="20000525142701794751"/>
<Field r="SYSTEM_TS" v="20000525141925862898"/>
<Field r="ADRTS" v="20000525141953813707"/>
<Field r="MYNKOD" v="0201" t="Allerød" tl="Allerød"/>
<Field r="VEJKOD" v="2073" t="Aarupvej" tm="Aarupvej"/>
<Field r="HUSNR" v="002"/>
<Field r="ETAGE" v="st"/>
<Field r="SIDEDOER"/>
<Field r="TILFLYKOMDATO" v="199902020000"/>
<Field r="BOLIGTYPKOD" v="0" t="Ukendt boligtype"/>
<Field r="POSTNR" v="3520" t="Farum"/>
<Field r="STARTDATO" v="200005201414"/>
<Field r="KOMKOD" v="0207" t="Farum" tl="Farum"/>
<Field r="FLYTPAMRK" v="X"/>
</Rolle>
</CprData>
</Service>
</System>
</Gctp>
```

Bemærk især feltet **SIDEDOER**. Dette felt er kun medtaget her for at vise hvordan et tomt felt **kan** se ud.

Normalt vil sådanne felter nemlig slet ikke blive sendt, da de jo netop ikke indeholder noget. Som det ses i servicespecifikationen er der mange flere felter i ADRESSE3 – men flere af dem kan ikke komme samtidigt (da personen jo kun kan bo **enten** i udlandet **eller** i Danmark **eller** være forsvundet).

I visse situationer kan serveren beslutte alligevel at sende tomme felter med i GCTP-strengen, oftest til debuging-formål. Blandt andet derfor er det nødvendigt at clienten kan reagere fornuftigt på en **dynamisk GCTP-streng**.

11. Eksempel 2: Vis personens nuværende og tidligere adresser (opslag med liste)

Initiering af en liste foregår på stort set samme måde, blot kan man i denne situation bede om den før nævnte tæller, der fortæller hvor mange rækker opslaget vil generere i alt. I denne situation har vi med AIA=X bedt om at få tælleren med ud.

```
<Gctp><System r="CprSoeg"><Service r="ADRESSE1"><CprServiceHeader r="ADRESSE1">
<Key><Field r="PNR" v="1212121212"/><Field r="AKX" v="X"/><Field r="MAXA" v="10"/>
<Field r="AIA" v="X"/></Key></CprServiceHeader></Service></System> </Gctp>
```

Læsevenligt :

```
<Gctp>
<System r="CprSoeg">
<Service r="ADRESSE1">
<CprServiceHeader r="ADRESSE1">
<Key>
<Field r="PNR" v="1212121212"/>
<Field r="AKX" v="X"/>
<Field r="MAXA" v="10"/>
<Field r="AIA" v="X"/>
</Key>
</CprServiceHeader>
</Service>
</System>
</Gctp>
```

Serverens svar : (her vises kun den læservenlige udgave)

```
<Gctp v="1.0" env="Test"
<System r="CprSoeg">
<Service r="ADRESSE1">
<CprServiceHeader r="ADRESSE1" ??????????????????????????/>
<CprData u="O">
<Rolle r="HovedRolle">
<Table r="ADRESSE" aia="20">
<Row k="19430730120019950508100017300852">
<Field r="TIMSTP" v="19950508100017300852"/>
<Field r="SYSTEM_TS" v="19950508115740600010"/>
<Field r="ADRTS" v="19950508100017300852"/>
<Field r="MYNKOD" v="0267" t="Skovby" tl="Skovby"/>
<Field r="ADRKOD" v="1"/>
<Field r="STARTDATO" v="194307301200"/>
<Field r="STARTDATOUSH" v="*/>
<Field r="HUSNR" v="020"/>
<Field r="VEJKOD" v="2650" t="Roskildevej" tm="Roskildevej"/>
<Field r="TILFLYKOMDTO" v="194307210000"/>
<Field r="CONVN" v="c/o Olga Persersen"/>
<Field r="CONVNTS" v="19981113090000000000"/>
<Field r="KOMKOD" v="0265" t="Roskilde" tl="Roskilde"/>
<Field r="POSTNR" v="4000" t="Roskilde"/>
<Field r="BOLIGTYPKOD" v="1" t="Egentlig beboelse"/>
```

```
<Field r="STADR" v="Roskildevej 20"/>
</Row>
<Row k="19430730120019910214163730671007194307301200A">
  <Field r="TIMSTP" v="19910214163730671007"/>
  <Field r="SYSTEM_TS" v="19910214122531900086"/>
  <Field r="ADRTS" v="19910214163730671007"/>
  <Field r="AKM" v="A" t="Fortrudd"/>
  <Field r="MYNKOD" v="0267" t="Skovby" tl="Skovby"/>
  <Field r="ADRKOD" v="2"/>
  <Field r="UDRLANDEKOD" v="5205" t="Atlantis" tl="Atlantis"/>
  <Field r="STARTDATO" v="194307301200"/>
  <Field r="SLUTDATO" v="194307301200"/>
  <Field r="SLUT_TS" v="19910214163730671007"/>
  <Field r="INDRLANDEKOD" v="0000" t="Ukendt Myndighed" tl="Ukendt Myndighed"/>
  <Field r="UDLANDSADR1" v="O-Adr-1"/>
  <Field r="UDLANDSADR2" v="O-Adr-2"/>
  <Field r="UDLANDSADR3" v="O-Adr-3"/>
  <Field r="UDLANDSADR4" v="O-Adr-4"/>
  <Field r="UDLANDADR_MYNKOD" v="0000"
    t="Ukendt Myndighed" tl="Ukendt Myndighed"/>
  <Field r="UDLANDADR_TS" v="194307300000000000000000"/>
  <Field r="STADR" v="I udlandet"/>
</Row>
<Row k="19430730120019910112130549528586194307301200K">
  <Field r="TIMSTP" v="19910112130549528586"/>
  <Field r="SYSTEM_TS" v="19910112115740600004"/>
  <Field r="ADRTS" v="19910112130549528586"/>
  <Field r="AKM" v="K" t="Rettet"/>
  <Field r="MYNKOD" v="0267" t="Skovby" tl="Skovby"/>
  <Field r="ADRKOD" v="1"/>
  <Field r="STARTDATO" v="194307301200"/>
  <Field r="SLUTDATO" v="194307301200"/>
  <Field r="HUSNR" v="030"/>
  <Field r="VEJKOD" v="2650" t="Roskildevej" tm="Roskildevej"/>
  <Field r="TILFLYKOMDTO" v="194307210000"/>
  <Field r="KOMKOD" v="0265" t="Roskilde" tl="Roskilde"/>
  <Field r="POSTNR" v="4000" t="Roskilde"/>
  <Field r="BOLIGTYPKOD" v="1" t="Egentlig beboelse"/>
  <Field r="STADR" v="Roskildevej 30"/>
</Row>
.
.
<7 andre rækker>
.
.
<Row u="REST" k="19301122120019910115100912813115193507220000"/>
</Table>
</Rolle>
</CprData>
<Kvit r="Ok" t="" v="0"/>
</Service>
</System>
</Gctp>
```

Læg mærke til at hver enkelt række har sin egen nøgle(k=), og at den sidste række har usage = Restart (U=REST). Denne nøgle skal bruges hvis klienten vil læse de sidste 10 rækker, vist i dette eksempel :

```
<Gctp>
<System r="CprSoeg">
  <Service r="ADRESSE1">
    <CprServiceHeader r="ADRESSE1">
      <Key>
        <Field r="PNR" v="1212121212"/>
        <Field r="AKX" v="X"/>
        <Field r="MAXA" v="10"/>
        <Field r="REST" v="19301122120019910115100912813115193507220000"/>
      </Key>
    </CprServiceHeader>
  </Service>
</System>
</Gctp>
```

I eksemplet er AIA-nøglen fjernet, da vi allerede fra første kald ved at der er 20. I stedet er restart-nøglen angivet.

Det er imidlertid vigtigt at angive kriterie-nøglerne igen (her PNR og AKX).

Hvis man i restart-kaldet **ikke** angav AKX=X vil dette kald ikke indeholde de fortrudte/rettede rækker og dermed måske ikke returnere 10 rækker som forventet.

12. Eksempel 3: Myndighedssøgning (begrænset søgning)

I det her viste eksempel ønsker klienten en liste over myndigheder der starter med "farum". Der bedes om at få returneret 15 hit af gangen, samt at få antal i alt (AIA=X) returneret.

```
<Gctp v="1.0" >
<System r="CprSoeg">
<Service r="MYN2">
<CprServiceHeader r="MYN2">
<Key>
<Field r="MYTE" v="farum"/>
<Field r="AIA" v="X"/>
<Field r="MAXA" v="15"/>
</Key>
</CprServiceHeader>
</Service>
</System>
</Gctp>
```

Serverens svar kan se sådan ud :

```
<Gctp v="1.0" env="Test" >
<System r="CprSoeg">
<Service r="MYN2">
<CprServiceHeader r="MYN2" ts="20010130200740753336"/>
<CprData u="O">
<Rolle r="HovedRolle">
<Table r="MYN" aia="2">
<Row k="0207Farum">
<Field r="CMYN_MYNKOD" v="0207" t="Farum" tl="Farum"/>
<Field r="CMYN_MYNTYP" v="05"/>
<Field r="CMYN_SLUTDATO"/>
</Row>
<Row k="7414Farum,Farum">
<Field r="CMYN_MYNKOD" v="7414"
t="Farum,Farum" tl="Farum Sogn, Farum Kommune"/>
<Field r="CMYN_MYNTYP" v="25"/>
<Field r="CMYN_SLUTDATO"/>
</Row>
</Table>
</Rolle>
</CprData>
</Service>
</System>
</Gctp>
```

Hvis klienten kun var interesseret i at søge efter kommuner (og altså i dette tilfælde få eet unikt hit) skulle klienten tilføje MYNT-nøglen således :

```
<Gctp v="1.0" >  
<System r="CprSoeg">  
<Service r="MYN2">  
<CprServiceHeader r="MYN2">  
<Key>  
<Field r="MYTE" v="farum"/>  
<Field r="MYNT" v="05"/>  
<Field r="AIA" v="X"/>  
<Field r="MAXA" v="15"/>  
</Key>  
</CprServiceHeader>  
</Service>  
</System>  
</Gctp>
```

Svaret ville næsten se ud som overstående, blot ville AIA være 1 og rækken med "Farum sogn" ville være væk.

13. Eksempel 4: Fødselsdato søgning (bred søgning)

Søgning på fødselsdato er en potentielt "tung" søgning.

Derfor vil serveren gemme nøglerne på de fundne rækker (her personnumre), således at man blot kan bruge de allerede fundne nøgler til at slå data op med hvis klienten beder om flere end de første 20 personer.

Det betyder omvendt at serveren har et stykke lagerplads liggende, som umiddelbart blot vil fylde op når brugeren er færdig med at bruge resultatet.

Derfor skal klienten sørge for at fortælle serveren det, når brugeren er færdig med listen.

Dette gælder i alle tilfælde hvor klienten har fået én eller flere rækker retur, men er selvsagt ikke nødvendig hvis søgningen ikke giver noget resultat.

Clientens forespørgsel :

```
<Gctp v="1.0" >
<System r="CprSoeg">
  <Service r="FODSOG2">
    <CprServiceHeader r="FODSOG2">
      <Key>
        <Field r="AIA" v="X"/>
        <Field r="FODT" v="19540322"/>
        <Field r="KOMK" v="0265"/>
      </Key>
    </CprServiceHeader>
  </Service>
</System>
</Gctp>
```

Man ønsker at finde personer som bor i Roskilde (kommunekode 265) og som er født den 22. marts 1954.

Serverens svar kan se sådan ud :

```
<Gctp v="1.0" env="Test" >
<System r="CprSoeg">
  <Service r="FODSOG2">
    <CprServiceHeader r="FODSOG2" ts="20010131155558793795"/>
    <CprData u="O">
      <Rolle r="HovedRolle">
        <Field r="SEARCH_ID" v="B553C318E0B9C000"/>
        <Table r="FOEDSOEG" aia="39">
          <Row k="1212121212B553C318E0B9C000">
            <Field r="PNR" v="1212121212"/>
            <Field r="ADRNVN" v="Riesen,Rie" t="Rie Riesen"/>
            <Field r="FOEDMYNTXT" v="Farum,Farum"/>
            <Field r="STADR" v="Grønnemosen 5"/>
            <Field r="POSTNR" v="4000" t="Roskilde"/>
            <Field r="KOEN" v="K"/>
          </Row>
          <Row k="1313131313B553C318E0B9C000">
            <Field r="PNR" v="1313131313"/>
          </Row>
        </Table>
      </Rolle>
    </CprData>
  </Service>
</System>
</Gctp>
```

```

<Field r="ADRNVN" v="Rolfesen,Rolf" t="Rolf Rolfesen"/>
<Field r="FOEDMYNTXT" v="Farum,Farum"/>
<Field r="STADR" v="Grønnemosen 5 A"/>
<Field r="POSTNR" v="4000" t="Roskilde"/>
<Field r="KOEN" v="M"/>
</Row>
<Row k="1414141414B553C318E0B9C000">
<Field r="PNR" v="1414141414"/>
<Field r="ADRNVN" v="Piasen,Pia" t="Pia Piasen"/>
<Field r="FOEDMYNTXT" v="Farum,Farum"/>
<Field r="ADRBESKYT" v="1"
    t="Beskyt" tm="Navne- og adresse besk." tl="Navne- og adresse beskyttelse"/>
<Field r="STADR" v="Roskildevej 20"/>
<Field r="POSTNR" v="4000" t="Roskilde"/>
<Field r="KOEN" v="K"/>
</Row>
<Row k="1515151515B553C318E0B9C000">
<Field r="PNR" v="1515151515"/>
<Field r="ADRNVN" v="Persen,Per" t="Per Persen"/>
<Field r="FOEDMYNTXT" v="Farum,Farum"/>
<Field r="STADR" v="Roskildevej 21 B"/>
<Field r="POSTNR" v="4000" t="Roskilde"/>
<Field r="KOEN" v="M"/>
</Row>
<Row k="1616161616B553C318E0B9C000">
<Field r="PNR" v="1616161616"/>
<Field r="ADRNVN" v="Oliassen,Ole" t="Ole Oliassen"/>
<Field r="CNVN_STATUS" v="80" ts="UDR" tl="Udrejst" t="Udrejst"/>
<Field r="FOEDMYNTXT" v="Farum,Farum"/>
<Field r="STADR" v="Roskildevej 25"/>
<Field r="POSTNR" v="4000" t="Roskilde"/>
<Field r="KOEN" v="M"/>
</Row>
.
.
<16 andre rækker>
.
.
<Row u="REST" k="2222222222B553C318E0B9C000"/>
</Table>
</Rolle>
</CprData>
</Service>
</System>
</Gctp>

```

Klienten skal sørge for at gemme værdien af SEARCH_ID. Denne skal bruges til at fortælle serveren når det er ok at slette søgedatalageret. Det kan f.eks. være fordi brugeren forlader listen eller hvis klienten har "læst i bund".

Bemærk at hvis det antal rækker der findes er mindre end MAXA (evt. defaultet til 20) vil der **ikke** blive gemt et søgedatalager, da klienten jo allerede har fået alle data. SEARCH_ID vil i dette tilfælde være blank, og vil derfor som regel ikke blive sendt ud.