



ODIN WEBSERVICE GRÆNSEFLADE

Version | 1.19

Udarbejdet af:

Søren Laurits Nielsen, Delegate



Indhold

1	Versions historik.....	4
2	Om dette dokument	5
3	Roller og adgange.....	5
3.1	Adgange.....	5
3.2	Låsning af konti.....	5
4	Miljø-oversigt.....	5
4.1	ODIN Miljøer	6
4.2	Login	6
5	Generelle krav til Web Service kald	7
5.1	Tekniske krav.....	7
5.2	ikke-tekniske krav.....	7
6	Indberetnings Web Service.....	7
6.1	Validering	9
6.1.1	Tips til valideringer	10
6.2	Statuskoder.....	12
6.3	Argumenter.....	12
6.4	Eksempler	13
7	Koblede rapporter.....	13
7.1	Eksempler	Fejl! Bogmærke er ikke defineret.
8	Stamdata Web Services.....	14
8.1	Stamdata2 metoder – vedligeholdelse af Stamdata.....	16
8.1.1	Køretøjer.....	16
8.1.2	Relation Køretøjer Station.....	17
8.1.3	Personer.....	18
8.1.4	Relation Person Station.....	18
8.1.5	Relation Person Kompetence	19
8.2	Stamdata eksempler	20
8.3	Værdilister eksempler	22
8.4	Hent kladder eksempel.....	24
9	Appendiks A: Indberet script.....	25
9.1	Eksempel 1.....	25
9.2	Eksempel 2.....	28

1 Versions historik

Version	Dato	Forfatter	Beskrivelse
1.0	22-09-2014	Søren Laurits Nielsen	Første samlede version
1.1	24-09-2014	Søren Laurits Nielsen	Revideret udgave efter møde med eksterne dataleverandører.
1.2	08-10-2014	Søren Laurits Nielsen	Tilføjet koblede rapporter
1.3	09-10-2014	Søren Laurits Nielsen	Rettet enkelte parametre på webservice kald
1.4	13-10-2014	Søren Laurits Nielsen	Rette stamdata webservice parametre
1.5	14-10-2014	Søren Laurits Nielsen	Fjernet GetSpecialMaterial web service – den erstattes af den tilsvarende pickliste.
1.6	29-10-2014	Søren Laurits Nielsen	Opdateret navn på Indberet webservice kald.
1.7	12-11-2014	Søren Laurits Nielsen	Tilføjet eksempel på opret indberetning
1.8	13-11-2014	Søren Laurits Nielsen	Rettet fejl i Delegate urls.
1.9	9-12-2014	Jacob Behrens	Tilføjet ”Aftale” i RapportTypeEnum pkt. 6.3
1.10	11-12-2014	Søren Laurits Nielsen	Opdateret metadata/pickliste returverdier.
1.11	22-12-2014	Søren Laurits Nielsen	Tilføjet XML versioner til stamdata webservices.
1.12	23-3-2014	Søren Laurits Nielsen	Tilføjet hent rapport webservice kald, og præciseret og tydeliggjort diverse valideringstips og fejl, der har vist sig at være problematiske.
1.13	5-10-2014	Søren Laurits Nielsen	Tilføjet informationer om hvordan man indberetter stationer i køretøjslisten. Tilføjet et eksempel PowerShell script til indberetning. Tilføjet en lang række nye stamdata opdateringsmetoder Tilføjet script til eksempel indberetning.
1.14	18-11-2015	Søren Laurits Nielsen	Tilføjet yderligere tekst til indberetningswebservice med hensyn til fejlhåndtering.
1.15	12-2-2016	Anders Dahl	Tilføjet dokumentation vedr. webservicekald til sletning af kladde rapporter
1.16	1-3-2016	Anders Dahl	Tilføjet dokumentation vedr. felterne meldingsadresseUdenVejnvan og skadestedUdenVejnavn
1.17	26-5-2016	Martin Reland	Tilføjet dokumentation vedrørende sikringDetektorType og sikringDetektorTypeAndet
1.18	17-10-2016	Søren Laurits Nielsen	Gennemgribende revideret.

1.19	30-11-2016	Lars Bjørn Andersen	Tilføjet max. rapportstørrelse
------	------------	---------------------	--------------------------------

2 Om dette dokument

Dette dokument er dokumentation for, hvordan Webservice laget i ODIN fungerer og hvordan det skal benyttes.

Dokumentet er tiltænkt personer med en vis teknisk indsigt i SOAP og REST protokollerne, der generelt bruges.

Bemærk: ”NYTIND” var arbejdstitlen for den version af ODIN der blev lanceret 1. Januar 2015.

3 Roller og adgange

I ODIN anvendes der windows authentication/Kerberos til alle webservice kald. De systemer der skal have adgang har derfor brug for at få oprettet en AD konto i Beredskabsstyrelsens eksterne AD (”DMZ”).

Kontakt Beredskabsstyrelsen / Martin Reland for information om dette. De brugere der oprettes i det eksterne AD, skal oprettes med en gyldig email adresse, hvis det f.eks. bliver nødvendigt at sende fejlbeskeder fra arbejdsprocesserne (f.eks. hvis en rapport ikke kan godkendes).

3.1 ADGANGE

En given webservice bruger kan have adgang til et eller flere beredskaber. Det er op til det eksterne system om der ønskes en enkelt generel konto med adgang til flere beredskaber, eller om der ønskes en særskilt konto pr. beredskab der skal kunne tilgås.

Det er ODIN administratorene der tildeler ”Indberetningsadgang”, evt. også ”lokal godkender” til de enkelte beredskaber.

3.2 LÅSNING AF KONTI

De enkelte konti er ikke begrænset med hensyn til IP adresser, og de er heller ikke underlagt krav om password skift.

Hvis der er mistanke om misbrug skal Beredskabsstyrelsen straks kontaktes, så kontoen enten kan spærres eller password skiftes.

4 Miljø-oversigt

Dette afsnit beskriver de forskellige ODIN miljøer der findes, og hvordan man logger ind.

4.1 ODIN MILJØER

Dette afsnit giver overblik over miljøerne der anvendes i ODIN:

Miljø	Beskrivelse	Adresser
Præprod / QA	<p>Miljø der hostes af Beredskabsstyrelsen.</p> <p>Dette miljø kan/skal bruges til at teste og verificere webservice kald. Data er ækvivalent med produktion, og der tildeles adgange på samme måde.</p> <p>NemLog-in adressen er koblet sammen med NemLog-in's PROD miljø, og man skal bruge gyldige NemLog-in medarbejdersignaturer.</p> <p>Der vil blive lagt kode og test data på dette miljø, ca. hver 1-2 uge.</p>	<p>NemLog-in: https://qa-odin.brs.dk Webservice: https://qa-odin-ws.brs.dk BRS Internt: https://qa-odin-brs.brs.dk</p>
Produktion	<p>Produktionsmiljø der hostes af Beredskabsstyrelsen.</p> <p>Må ikke bruges til test af webservices.</p> <p>Der vil kun blive lagt kode på dette miljø i planlagte servicevinduer, forventet hver 1-2 måneder.</p>	<p>NemLog-in: https://odin.brs.dk Webservice: https://odin-ws.brs.dk BRS Internt: https://odin-brs.brs.dk</p>

4.2 LOGIN

På alle ODIN miljøer findes der tre måder at logge ind på:

- **WebService login:** På webservice adresserne benyttes NTLM/Kerberos autentifikation, der er velegnet til webservice integrationer. Kontoerne er i "DMZ" domænet.

I nærværende dokument anvendes <odin url> som angivelse for denne adresse.

- **NemLog-in:** Man bliver som udgangspunkt sendt til NemLog-in, når man besøger et ODIN site.

NemLog-in adgangen kan ikke benyttes af webservices og beskrives ikke yderligere i dette dokument.

- **BRS internt:** Beredskabsstyrelsen kan logge ind med deres BRS konto. Det er beredskabsstyrelsens eksisterende ADFS miljø der sørger for login, derfor vil der være automatisk login for BRS ansatte. Kan ikke bruges til webservices.

5 Generelle krav til Web Service kald

5.1 TEKNISKE KRAV

Systemerne der skal kunne integrere med ODIN skal understøtte:

1. Windows autentifikation
2. SSL over port 443
3. Være i stand til at anvende REST eller SOAP webservices og fortolke svarkoder korrekt

Note: For .NET applikationer kan man valgfrit anvende klassisk SOAP (.NET 2.0) eller WCF, hvor første mulighed dog er simplest at anvende.

For simple test eller browser applikationer er det mest praktisk at anvende REST versionerne (note: Der er *ikke* et \$metadata endpoint)

For system integration er SOAP protokollen normalt at foretrække, da den giver mulighed for autogenerering af proxy klasser.

Teknisk detalje: Bemærk at webservices er bygget med WCF, og der for visse value type argumenter skal specificeres om de er angivet eller ej, f.eks. "NewStatusSpecified" i afsnit 6.4.

5.2 IKKE-TEKNISKE KRAV

Der indgår en lang række stamdata værdier i ODIN, f.eks. personer, køretøjer og diverse værdilister.

Afsnit 7 angiver hvordan disse værdier løbende kan hentes ud.

Beredskabsstyrelsen anbefaler at disse værdilister kun gemmes i kortere tid, for at optimere performance uden at gå på kompromis med datakvaliteten.

Disse stamdata ændres sjældent derfor vil det ofte være tilstrækkeligt, f.eks. at hente dem en gang i døgnet. Data mængden kan være flere hundrede KB, derfor anbefales det de så vidt muligt gemmes.

6 Indberetnings Web Service

Der er lavet webservice metoder, hvor man kan indberette nye rapporter, opdatere eksisterende, slette eller hente eksisterende ud.

Generelt anvendes et "beredskabsnummer" til at identificere de enkelte beredskaber, f.eks. "1007" for Gentofte beredskab, og et "brsId" felt til at identificere rapporterne, som er en systemgenereret streng, f.eks. "1007-34", der angiver at det er en rapport indenfor beredskabet med nummer "1007".

Alle nedenstående adresser præfikses med **<odin url>/_vti_bin/Delegate.BRS.NytInd/Services/**

Service	Service metode	Betydning
indberetning.svc (Type: REST)	Indberet	Indsend en fuld indberetning, inkl. status af rapport. Type: POST REST endpoint.
	UpdateIndberetning	Opdater en eksisterende indberetning evt. med et status skift Type: POST REST endpoint.
	FetchIndberetning/ <i>brsId</i>	Returnerer den rå xml fil for en konkret rapport. Nyttig til udviklingsformål. Eks: FetchIndberetning/1067-12 Type: GET REST endpoint.
	DeleteIndberetning/ <i>brsId</i>	Sletter kladden med det givne id. Fungerer kun på kladder. F.eks. DeleteIndberetning/1067-12 Type: GET REST endpoint.
	AddFile	Parametre: String rapportId: BRS id f.eks. 1067-12 String name: Filnavn byte[] file: fil indhold bool overwrite: Overskrive eksisterende, hvis den findes? Type: POST REST endpoint
indberetningSoap.svc (Type: SOAP)	mex?singleWsdL	Samme metoder som ovenstående, men med SOAP protokol.
Related.svc (Type: REST)	SearchForRelated/?easting= <i>easting</i> &northing= <i>northing</i> &received= <i>alarmReceived</i> &extendedSearch= <i>extendedSearch</i>	Søg efter rapporter der kan kobles sammen. Parametre: X = easting koordinat som heltal, hvis easting er 689976.401, skal der indættes "689976401", Y = northing koordinat som heltal, Received=Alarm tidspunkt som yyyy-MM-ddTHH-mm-ss, extendedSearch=true/false udvid søgekritererne for søgning (større tidspunkt, større geografisk boks) Bemærk: Der benyttes GPS koordinater i ETRS89 koordinater.
RelatedSoap.svc (Type: SOAP)	mex?singleWsdL	Samme metode som ovenstående, men med SOAP protokol. Bemærk der er enkelte ekstra web service metoder, der ikke er tænkt anvendt af dataleverandører.
ReportSearch.svc (Type: REST)	FindReports?beredskabsNummer= <i>brs_nummer</i> &rapportStatuses= <i>rapportstatus</i> &sortByPropertyname= <i>AlarmModtaget</i> &take= <i>antal</i> &skip= <i>springover</i>	Metode til at hente en liste af rapporter. Der returneres et JSON objekt, der indeholder alle rapporter der matcher de givne parametre.

		<p>Dette kald ligger til grund for siderne med lister af kladder og godkendte rapporter, og kan evt. benyttes til integrationer, hvis det ønskes.</p> <p>Argumenter:</p> <p><i>Brs_nummer</i>: 4 cifret nummer på beredskab</p> <p><i>Rapportstatus</i>: Liste af rapport status koder der skal returneres, f.eks. [”Kladde”] (som dog skal URI encoded til %5B%22Kladde%22%5D) (sortByPropertyName: Egenskab der skal sorteres efter, benyt ”AlarmModtaget” feltet.)</p> <p><i>antal</i>: Valgfrit. Antal der skal returneres</p> <p><i>springover</i>: Valgfrit. Antal der skal springes over. Kan bruges til at lave paging over et større antal elementer.</p>
ReportSearchSoap.svc (Type: SOAP)	mex?singleWsdl	<p>Samme metode som ovenstående, men med SOAP protokol.</p> <p>Bemærk der er enkelte ekstra web service metoder, der ikke er tænkt anvendt af dataleverandører.</p>

Det er vigtigt at bemærke at rapporterne består af en meget lang række felter, rækkefølge er ligegyldig, kun feltnavne er vigtige.

Data til værdilister- og stamdata felter findes i afsnit 7.

6.1 VALIDERING

De indsendte rapporter udsættes for samme validering, som der findes på web grænsefladen. Når valideringsreglerne justeres i web formularen, vil ændringen også slå igennem i webservice laget.

Data der indsendes som rapport accepteres som en *Kladde* med meget minimal validering; kun syntakst, værdilister og adgangsrettigheder kontrolleres. Hvis en af disse fejler returneres der en SOAP/http fejl.

Selvom rapporten accepteres som kladde, valider-es den alligevel, og valideringsresultatet returneres fra WebServicen. Disse valideringsfejl er rent informative.

Hvis rapporten ønskes markeret som ”Afsluttet” (eller senere statuskoder) valideres den mod alle valideringsregler, og der returneres evt. en SOAP/HTTP fejl, hvor fejlbeskeden angiver de nærmere valideringsregler der blev brudt.

6.1.1 TIPS TIL VALIDERINGER

Webservicen vil generelt returnere en liste af felter og tilhørende valideringsfejl, hvis der er nogen.

Det er værd at kende følgende detaljer:

- Systemet er generelt bygget, så der ikke må rapporteres data i ”lukkede” sektioner, hvis det f.eks. ikke er en redningsoperation, så er der ingen af redningsfelterne der må udfyldes. Hvis der derimod er tale om en redning, så skifter mange af disse felter til at være obligatoriske. Fejl beskederne for disse tilfælde er ”Må ikke udfyldes” og ”Skal udfyldes”.
- Det valideres at der er fortaget gyldige valg i værdilisterne, og man skal være opmærksom på at de gyldige muligheder af og til ændres ud fra rapport typen.

Som special tilfælde kan nævnes feltet ”Alarm modtaget fra” (felt: alarmModtagetFra), hvor den valgte værdi bruges til at filtrere de mulige værdier i Første Meldings Ordlyd (foersteMeldingsOrdlyd). I pickliste værdiern for første melding anvendes egenskaben ”brs-alarmmodtagetfra” til filtreringen. De mulige værdier er derfor dem der har den valgte alarm modtaget fra værdi i brs-alarmmodtaget fra feltet (kommasepareret liste af tal)

- Der er et antal af pickliste felterne, der inkluderer et felt til supplerende information. Hvis der derfor vælges en værdiliste værdi, hvor der findes en property ”visandetfelt” med værdien ”true”, så skal der yderligere indsendes et andet felt i tillæg til værdiliste værdien. Eksempel: Hvis der i Hændelses type (”haendelsestype”) indrapporteres ”20990” – ”Anden naturhændelse”, så skal feltet ”haendelsestypeAndet” også udfyldes – ellers kommer der en valideringsfejl på det felt.
- Der er ikke et krav om rækkefølge eller hierarki i de indrapporterede JSON data. Men hvis man vælger at indrapportere en hel sektion, så skal hele den pågældende sektion indsendes. Hvis man f.eks. i sin JSON indrapporterer ”... Skadestedet: {skadestedVejnavn: ”Datavej”} ...”, så skal man være opmærksom på at alle andre felter under skadested nu er blevet fjernet (null), og specielt boolean felter vil fejle valideringen, da de nu mangler. I det her tilfælde vil ”SkadestedLigMelding” (der har standard værdien ”true”) være blevet null, og give en fejl. Fejl beskeden er i dette tilfælde ”the enumeration constraint failed”, som måske ikke er helt indlysende. Anbefalingen er derfor *ikke* at indsende hierarkiske objekt, udover i de tilfælde hvor der er tale om tabel rækker, f.eks. personer, køretøjer m.fl.
- Der valideres på at både personer og køretøjer er gyldige. Dvs. at de pågældende personer/køretøjer er tilknyttet en station under det pågældende beredskab på det tidspunkt rapporten omhandler (alarm tidspunkt). Hvis stamdata ikke er opdateret kan man derfor opleve valideringsfejl på disse.

- Der anvendes en lang række system felter, f.eks. ”antalAfPersoner”, som beregnes af systemet. Man kan godt indrapportere dem uden at få fejl, men værdierne overskrives af systemet. Derfor kan man også opleve at få valideringsfejl på felter, der ikke er indrapporteret, f.eks. vil ”antalAfPersoner” fejle i valideringen hvis feltet er lig nul, dvs. at der ikke er indsendt nogen personer.
- Specielt for skadestedsadresser er der flere der har haft problemer med feltet ”skadestedLigMelding” der betyder at skadestedsadressen er den samme som meldingsadressen, og derfor kopierer systemet felterne fra meldingsadressen til skadestedsadressen. Hvis man så mangler et par felter i meldingsadressen, så kan man få valideringsfejl på at man mangler at udfylde enkelte felter i skadestedsadressen, selvom de er indsendt, da deres værdi bliver overskrevet af meldingsadressen.

6.2 FEJLRETURKODER

Bemærk at der er en række situationer, hvor metoderne returnerer en server fejl (http kode 5xx) i stedet for valid returdata. Hvis man får en fejl tilbage, så har webservicen ikke lavet ændringer og man kan ikke regne med at indberetningen er modtaget/opdateret.

De situationer der vil give fejl er bl.a.:

- Manglende IndberetningRequestDto request objekt
- Ugyldigt beredskabsnummer
- Manglende json værdi (null)
- Ingen adgang til nævnte beredskab
- Forkert json syntakst
- Manglende ”udrykningsrapport” i json objektet
- Forkert rapport type i forhold til beredskabet (kommunalt eller ikke)
- Manglende rettigheder til at lave en rapport med den ønskede status kode
- **Ønskede rapport status <> kladde og der er valideringsfejl**
- Evt. system fejl f.eks. hvis der er et problem med workflows etc. (meget sjældent), eller serveren ikke kan følge med etc.

Bemærk fejlen markeret med rødt – hvis man forsøger at indberette en rapport som ”GodkendtLokalt”, så skal den være uden valideringsfejl ellers kan den ikke godkendes. Webservicen vil derfor returnere en fejl, hvis det forsøges.

Hvis man, som kalder, har brug for at kende og håndtere valideringsfejlene, så skal man først indberette rapporten som en kladde (som vil give et rapport id og liste over evt. valideringsfejl). Hvis der ikke er valideringsfejl, så kan man opdatere rapportens status efterfølgende med *UpdateIndberetning*.

Uden for selve webservicen fås også fejl for:

- Manglende, ugyldig, udløbet webservice konto
- Fejl i SOAP formateret besked
- Fejl i url
- Server nede (f.eks. i deployment vinduer)
- Netværk nede (enten kalder eller server)
- M.fl.

6.3 STATUSKODER

En Rapport kan gennemløbe statusværdierne:

- **Kladde**, endnu ikke afsluttet eller godkendt.
- **Afsluttet**, Afsluttet af indberetter
- **GodkendtLokalt**, Godkendt af lokal administrator.
- **GodkendtCentralt**, Godkendt af central administrator.
- **Arkiveret**

Bemærk at der er rettighedsbegrænsninger på hvem der kan anvende visse statuskoder.

6.4 ARGUMENTER

Der anvendes en række objekter til at sende argumenter frem og tilbage i webservice funktionerne.

Dette afsnit knytter kommentarer til enkelte af disse, men er ikke udtømmende.

IndberetningRequestDto: Input objekt til Indsend

```
public class IndberetningRequestDto
{
    public string Json;
    public string BeredskabsNummer;
    public RapportTypeEnum Type;
    public RapportStatus Status;
}
```

Json er selve rapporten som et json objekt.

RapportTypeEnum er:

```
public enum RapportTypeEnum
{
    Assisterende,
    Aftale,
    IBR,
    Kemi,
    Nuc,
    StandardKommunalt,
    StandardStatsligt
}
```

IndberetningResponseDto: Svar objekt fra Indsend og OpdaterIndberetning.

```
public class IndberetningResponseDto
{
    Dictionary<string, List<string>> Errors;
    string RapportId;
    string Message;
}
```

Message er en besked til brugeren, og har p.t. værdien "Indberetning oprettet" eller "Kladde oprettet med fejl".

RapportId er det systemgenererede ID for rapporten, f.eks. "1020-321".

IndberetningOpdaterRequestDto: Input objekt til OpdaterIndberetning

```
public class IndberetningOpdaterRequestDto
{
    public string RapportId { get; set; }
    public string Json { get; set; }
    public RapportStatus NewStatus { get; set; }
    public bool NewStatusSpecified { get; set; }
}
```

NewStatusSpecified: Bemærk at dette felt skal være sandt, hvis der ønskes en specific status – ellers anvendes "kladde".

6.5 EKSEMPLER

Se appendiks A.

7 Koblede rapporter

Det er muligt at koble relaterede rapporter sammen i indberetnings webservices.

Denne webservice søger efter rapporter der "minder" om hinanden i geografisk- og tidsmæssig placering.

Note: Bemærk at der er en forsinkelse på op til 30 min, før en rapport kan fremsøges for at blive associeret. Hvis det er *et* system der indsender en stribe relaterede rapporter, så kan man benytte brsId returneret fra de enkelte indberetninger til at koble med.

Service	Service metode	Betydning
Related.svc/c/	<i>/SearchForRelated/?easting=easting&northing=northing&received=alarmReceived&extendedSearch=extendedSearch</i>	<p>Søg efter rapporter der kan kobles sammen.</p> <p>Parametre: X = easting koordinat som kommatall, hvis easting er 689976.401, skal der indættes ”0. 689976401”, Y = northing koordinat som kommatall, Received=Alarm tidspunkt som yyyy-MM-ddTHH-mm-ss, extendedSearch=true/false udvid søgekritererne for søgning (større tidspunkt, større geogradisk boks)</p>
RelatedSoap.svc	<i>/mex?singleWsdll</i>	<p>Samme metode som ovenstående, men med SOAP protokol.</p> <p>Bemærk der er enkelte ekstra web service metoder, der ikke er tænkt anvendt af dataleverandører.</p>

Returværdierne er en liste af ”RelatedItemDto” objekter

```
public RelatedItemDto {
    public string Title;
    public string Beredskabsnummer;
    public string ListId;
    public DateTime? AlarmModtaget;
    public string RoadName;
    public string HouseNo;
    public string Municipality;
    public string FirstReportWording;
    public string Station;
    public int RelatedCount;
    public string Id;
    public string Path;
    public string BrsId;
}
```

Hvor ”title” er en tekststreng, der kan vises til en bruger, der evt. kan vælge om rapporter skal være relaterede.

BrsId er et ID nummer, der skal anvendes til at koble rapporter sammen i indberetningswebservicen.

8 Stamdata Web Services

Dette afsnit beskriver de web services der er lavet til at hente stamdata værdier ud til brug for indberetning af rapporter.

Bemærk at der er to stamdata webservices, hvor begge er bibeholdt for at sikre kompatibilitet. Det anbefales at benytte ”Stamdata2” versionen.

8.1 STAMDATA METODER (VERSION 1)

Alle nedenstående adresser præfikses med **<odin url>/_vti_bin/Delegate.BRS.NytInd/Services/**

Service	Service metode	Betydning
Stamdata.svc/	GetVehicles?name= <i>stationsNavn</i> &queryDate= <i>Dato</i>	Hent køretøjer tilknyttet <i>stationsNavn</i> på <i>dato</i>
	GetVehiclesXml?name= <i>stationsNavn</i> &queryDate= <i>Dato</i>	Som ovenstående, returdata i XML i stedet for SOAP.
	GetVehiclesByLocalStation?brs= <i>beredskabsnummer</i> &localId= <i>localId</i> &queryDate= <i>Dato</i>	Hent køretøjer ud fra en stations local ID nummer på dato
	GetVehiclesByLocalStationXml?brs= <i>beredskabsnummer</i> &localId= <i>localId</i> &queryDate= <i>Dato</i>	Som ovenstående, returdata i XML i stedet for SOAP.
	UpdateVehicle	Opdaterer et køretøjs stamdata. Se eksempler nedefor for detaljer.
	GetPersons?name= <i>stationsNavn</i> &queryDate= <i>date</i>	LocalId er det lokale id nummer, som f.eks. Falck har tildelt til sine stationer.
	GetPersonsByLocalStation?brs= <i>beredskabsnummer</i> &localId= <i>localId</i> &queryDate= <i>Dato</i>	Hent personer ud fra en stations lokal ID nummer på dato
	GetPersonsByLocalStationXml?brs= <i>beredskabsnummer</i> &localId= <i>localId</i> &queryDate= <i>Dato</i>	Som ovenstående, returdata i XML i stedet for SOAP.
	UpdatePerson	Opdater en persons stamdata. Se eksempler nedefor for detaljer.
	GetStations?brs= <i>beredskabsNummer</i> ?queryDate= <i>dato</i>	Dato er datoen hvor der forspørges på relationen. Oftest lig dags dato
GetStationsXml?brs= <i>beredskabsNummer</i> ?queryDate= <i>dato</i>	Som ovenstående, returdata i XML i stedet for SOAP.	
StamdataSoap.svc	/mex?singleWsdL	Samme funktioner som i ”stamdata.svc”, men med SOAP protokol.
Metadata.svc/	GetPickLists	Henter alle værdilister (200kb)
	GetPickList/ <i>FieldName</i>	Henter en enkelt værdiliste
	GetPickListJson/ <i>FieldName</i>	Henter en enkelt værdiliste i json format
	GetSelectedPickLists/ <i>FieldName1,FieldName2,FieldName3</i>	Hent udvalgte værdilister
	GetSelectedPickListsJson / <i>FieldName1,FieldName2,FieldName3</i>	Hent udvalgte værdilister i json format
MetadataSoap.svc	/mex?singleWsdL	Samme funktioner som i ”Metadata.svc”, men med SOAP protokol.

8.2 STAMDATA2 METODER – VEDLIGEHOLDELSE AF STAMDATA

Denne sektion indeholder metoder der er tilføjet efter 1/10-2015 til nemmere vedligeholdelse af stamdata i ODIN.

Alle metoder følger samme principper, og der er metoder til at vedligeholde alle stamdata objekter.

Af hensyn til bagudkompatibilitet er der ikke ændret i eksisterende ”Stamdata.svc” metoder.

Adressen til webservicen er:

REST interface: `<odin url>/_vti_bin/Delegate.BRS.NytInd/Services/Stamdata2.svc`

SOAP interface: `<odin url>/_vti_bin/Delegate.BRS.NytInd/Services/Stamdata2Soap.svc?singleWsdll`

Det er valgfrit hvilket interface man vil benytte, dog skal det bemærkes at i REST interfacet kan Add/Update metoderne kun kaldes med en http POST operation, og derfor kan være lidt vanskeligere at benytte.

Generelle bemærkninger:

Objekterne følger tæt datamodellen i ODIN, og er alle lavet som simple CRUD operationer.

Metoderne arbejder ofte med ID felter, der alle er heltal. Hvis det ikke ønskes at angive et ID (f.eks. for en søgning), så anvendes ”-1”.

Alle opdateringsmetoderne opdaterer generelt kun de værdier der er indsendt, alle null / manglende værdier ignoreres. Eneste undtagelse fra denne regel er relationernes slut dato, der godt kan være null, og derfor har et tilknyttet boolsk flag om de skal opdateres.

Alle værdilisteværdier er angivet ved deres ”brs-nøgle” id.

Alle datostempler angives med en xml type dato, f.eks. ”2015-10-05T12:23:42” i lokal dansk tid. Hvis der ønskes GMT tid angives ”2015-10-05T12:23:42Z”

8.2.1 KØRETØJER

Dataobjekt:

`KoeretoejDto`

```
{
    int Id
    string Navn
    int KoeretoejTypeId
    string Stelnummer
    string Nummerplade
}
```



```

        int Standardbemanding
    }

```

(hvor -1 kan anvendes til "int" typerne for at angive null).

KoeretoejTypeId svarer til "KøretøjType" værdilisten.

Metoder:

Service	Service metode	Betydning
Stamdata2.svc/	HentKoeretoej?id={id}	Henter et køretøj fra id
Stamdata2Soap.svc/	mex?singleWsdl	Metadata for webservice
	KoeretoejDto HentKoeretoej(id)	Henter et køretøj fra id
	Void OpdaterKoeretoej(KoeretoejDto koeretoej)	Opdaterer et køretøj (der tidligere er hentet)
	int TilfoejKoeretoej(KoeretoejDto koeretoej)	Tilføjer et nyt køretøj.
	HentKoeretoejMedStelnummer	
	HentKoeretoejMedNummerplade	

8.2.2 RELATION KØRETØJER STATION

Dataobjekt:

RelationKoeretoejStationDto

```

{
    int Id
    int StationId
    int KoeretoejId
    string RelationStartDato
    string RelationSlutDato
    string LokalId
}

```

Metoder:

Service	Service metode	Betydning
Stamdata2.svc/	HentRelationKoeretoejStation?brs={beredskabsnummer}&koeretoejid={koeretoejId}&stationid={stationId}&queryDate={queryDate}	Henter relationer. Angive enten køretøj, station eller begge. Dato er frivillig.
Stamdata2Soap.svc/	mex?singleWsdl	Metadata for webservice
	RelationKoeretoejStationDto[] HentRelationKoeretoejStation(int koeretoejId, int stationId, DateTime? queryDate)	Angiv enten køretøj id eller stationid eller begge.
	Void OpdaterRelationKoeretoejStation(RelationKoeretoejStationDto	

	dto, bool opdaterSlutdato)	
	int TilfoejRelationKoeretoejStation(int beredskabsnummer, RelationKoeretoejStationDto dto)	Tilføj ny relation. Returnerer relationsid

8.2.3 PERSONER

Dataobjekt:

```
PersonDto
{
    int Id
    string Navn
    string Cpr
}
```

Bemærk at CPR nummer ikke returneres fra metoderne.

Metoder:

Service	Service metode	Betydning
Stamdata2.svc/	HentPerson?id= {id}	
	HentPersonMedCpr?cpr= {cpr}	
Stamdata2Soap.svc/	mex?singleWsdL	Metadata for webservice
	PersonDto HentPerson(int id)	
	PersonDto HentPersonMedCpr(string cpr)	
	void OpdaterPerson(PersonDto person)	
	int TilfoejPerson(PersonDto person)	

8.2.4 RELATION PERSON STATION

Dataobjekt:

```
RelationPersonStationDto
{
    int Id
    int StationId
    int PersonId
    string LokalId
    int StillingId
    int AnsaettelsesStatusId
    string RelationStartDato
    string RelationSlutDato
}
```

Hvor

StillingId svarer til "PersonStilling" værdilisten

AnsaettelsesStatusId svarer til "PersonAnsaettelsesStatus" værdilisten

Metoder:

Service	Service metode	Betydning
Stamdata2.svc/	HentRelationPersonStation?brs={beredskabsnummer}&Personid={PersonId}&stationid={stationId}&queryDate={queryDate}	Henter relationer. Angiv enten person, station eller begge. Dato er frivillig.
Stamdata2Soap.svc/	mex?singleWSDL	Metadata for webservice
	RelationPersonStationDto[] HentRelationPersonStation(int PersonId, int stationId, DateTime? queryDate)	
	Void OpdaterRelationPersonStation(RelationPersonStationDto dto, bool opdaterSlutdato)	
	int TilfoejRelationPersonStation(int beredskabsnummer, RelationPersonStationDto dto)	Tilføj ny relation. Returnerer relationsid

8.2.5 RELATION PERSON KOMPETENCE

Dataobjekt

`RelationPersonKompetenceDto`

```
{
    int Id
    int PersonId
    int KompetenceId
    string RelationStartDato
    string RelationSlutDato
}
```

Metoder:

Service	Service metode	Betydning
Stamdata2.svc/	HentRelationPersonKompetence?Personid={PersonId}&Kompetenceid={KompetenceId}&queryDate={queryDate}	Henter relationer. Angiv personid. KompetenceId og Dato er frivillige.
Stamdata2Soap.svc/	mex?singleWSDL	Metadata for webservice
	RelationPersonKompetenceDto[] HentRelationPersonKompetence(int PersonId, int KompetenceId, DateTime? queryDate)	Angiv personId. Evt. kompetenceid og/eller dato.
	Void OpdaterRelationPersonKompetence(RelationPersonKompetenceDto dto, bool opdaterSlutdato)	
	int TilfoejRelationPersonKompetence(RelationPersonKompetenceDto dto)	Tilføj ny relation. Returnerer relationsid

8.3 STAMDATA EKSEMPLER

[https://qa-odin-ws.brs.dk/_vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/GetVehicles?name=Beredskabsstyrelsen Midtjylland&queryDate=2014-10-10](https://qa-odin-ws.brs.dk/_vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/GetVehicles?name=Beredskabsstyrelsen%20Midtjylland&queryDate=2014-10-10)

```
[{"DefaultPersons":3,"FilterDate":"2014-10-10T00:00:00","Id":266,"LocalIds":[null],"Name":"TOYOTA INTLC 1 - 300332","Registration":null,"SerialNo":null,"VehicleType":null}, {"DefaultPersons":3,"FilterDate":"2014-10-10T00:00:00","Id":254,"LocalIds":[null],"Name":"BUS 1 - 300456","Registration":null,"SerialNo":null,"VehicleType":null...}
```

https://qa-odin-ws.brs.dk/_vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/UpdateVehicle

(Kræver en post, eller en SOAP request).

Input argumentet er et ”`KoeretoerjDetailsDto`” objekt, der indeholder:

```
[DataContract]
public class KoeretoerjDetailsDto
{
    [DataMember]
    public int Id;

    [DataMember]
    public string Navn;

    [DataMember]
    public KoeretoerjTypeDto KoeretoerjType;

    [DataMember]
    public string KoeretoerjStelNummer;

    [DataMember]
    public string KoeretoerjNummerplade;
}
```

Og det refererede `KoeretoerjTypeDto` er:

```
[DataContract]
public class KoeretoerjTypeDto
{
    [DataMember]
    public string Navn;

    [DataMember]
    public string Id;
}
```

Alle felter i KoeretoeyDetailsDto er påkrævede. Køretøjet findes ud fra dets ID (et tal), og alle de øvrige data felter opdateres derefter. Hvis køretøjet ikke findes, returneres en fejl.

https://qa-odin-ws.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/GetPersons?name=Sorø&queryDate=2014-10-10

```
[{"FilterDate":"2014-10-10T00:00:00","Id":1579,"LocalIds":[""],"Navn":"Johansen, Bo", "PersonCpr":-1},
{"FilterDate":"2014-10-10T00:00:00","Id":1589,"LocalIds":[""],"Navn":"John Bjørkman", "PersonCpr":-1},
{"FilterDate":"2014-10-10T00:00:00","Id":1554,"LocalIds":[""],"Navn":"Racher, Douglas", "PersonCpr":-1}...
```

https://qa-odin-ws.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/UpdatePerson

(Kræver en post, eller en SOAP request).

Der skal indsendes et ”PersonDTO” objekt af typen:

```
[DataContract]
public class PersonDto
{
    [DataMember]
    public int Id { get; set; }

    [DataMember]
    public string Navn { get; set; }

    [DataMember]
    public string PersonCpr { get; set; }
}
```

Personen identificeres med et ID, et tal, og der returneres en fejl, hvis denne ikke eksisterer. Navn og CPR nummer opdateres.

https://qa-odin-ws.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/stamdata.svc/GetStations?brs=1111&queryDate=2014-10-10

```
[{"FilterDate":"2014-10-10T00:00:00","Id":48,"LocalIds":[null],"Navn":"Falck-Hammel"}, {"FilterDate":"2014-10-10T00:00:00","Id":339,"LocalIds":[null],"Navn":"Beredskabskontoret Hammel"}, {"FilterDate":"2014-10-10T00:00:00","Id":106,"LocalIds":[null],"Navn":"Favrskov brandvæsen,
```

```
Hadsten"}, {"FilterDate": "2014-10-10T00:00:00", "Id": 202, "LocalIds": [null], "Navn": "Falck-Hinnerup"}]
```

8.4 VÆRDILISTER EKSEMPLER

Alle metoderne returnerer den samme datastruktur.

https://qa-odin-ws.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/metadata.svc/GetSelectedPickListsJson/BrandDirekteÅrsagsType,BrandMedvirkFaktor

```
[
  {
    "DT": "Menneskelig handling-disabled",
    "DTR": "Menneskelig handling",
    "Desc": "",
    "FN": "BrandDirekteÅrsagsType",
    "PRPS": [
      {
        "Key": "brs-gruppe",
        "Value": ""
      },
      {
        "Key": "brs-nøgle",
        "Value": "10000"
      },
      {
        "Key": "brs-odinnøgle",
        "Value": ""
      },
      {
        "Key": "brs-rapport-type",
        "Value": "standardkommunalt"
      }
    ],
    "TV": "10000"
  },
  {
    "DT": "-- Tilsigtet",
    "DTR": "Tilsigtet",
    "Desc": "Inkluderer bevidst misbrug af varmekilde eller en brand der har brandstiftende karakter.",
    "FN": "BrandDirekteÅrsagsType",
    "PRPS": [
      {
        "Key": "brs-gruppe",
        "Value": "10000"
      },
      {
        "Key": "brs-nøgle",
        "Value": "10100"
      },
      {
        "Key": "brs-odinnøgle",
        "Value": "20560000, 20562892, 20566096, 20567216"
      },
      {
        "Key": "brs-rapport-type",
        "Value": "standardkommunalt"
      }
    ],
  },
]
```

```

    "TV": "10100"
  },
  ...

```

Retur data objekterne er en liste af (skrevet som XML):

Data element	Forklaring
<PLI>	Pick List Item
<DT>Menneskelig handling-disabled</DT>	<p>Display Text – det der skal vises på skærmen</p> <p>BEMÆRK:</p> <ul style="list-style-type: none"> - Der tilføjes ”-disabled” hvis værdien skal vises, men ikke være valgbar (f.eks. en overskrift). - Der tilføjes ”-deprecated” hvis værdien ikke længere er brugbar, men bibeholdt af historiske årsager, f.eks. når en gammel rapport vises, som indeholder en værdi der senere er blevet ugyldig. - Både –disabled og –deprecated kan godt forekomme på samme værdi. <p>Det er ikke tilladt at indberette med disse værdier, selvom det dog er teknisk muligt for at imødekomme grænsetilfælde og historiske data.</p> <p>Beredskabsstyrelsen sletter IKKE værdier fremadrettet, de markeres som –disabled.</p>
<DTR>Menneskelig handling</DTR>	Display Text – det der skal vises på skærmen, dog uden de ekstra ”-disabled”/”-deprecated” markeringer.
<FN>BrandDirekteÅrsagsType</FN>	Field Name – Værdilistens navn
<PRPS>	Properties – liste af egenskaber
<PRP>	Property #1
<Key>brs-gruppe</Key>	Key – Navn på egenskab for pågældende værdiliste element
<Value/>	Value
</PRP>	
<PRP>	Property #2
<Key>brs-nøgle</Key>	Key – Navn på egenskab for pågældende værdiliste element
<Value>10000</Value>	Value – Værdi af egenskab
</PRP>	
</PRPS>	
<TV>10000</TV>	Text Value – værdien der skal gemmes i rapporten.
</PLI>	
<PLI>	Værdiliste element #2
<DT>-- Forsætlig</DT>	
<FN>BrandDirekteÅrsagsType</FN>	Navn på værdiliste
<PRPS>	
<PRP>	
<Key>brs-gruppe</Key>	
<Value>10000</Value>	

</PRP>	
<PRP>	
<Key>brs-nøgle</Key>	
<Value>10100</Value>	
</PRP>	
</PRPS>	
<TV>Forsætlig</TV>	Text Value – værdien der skal gemmes i rapporten. Disabled er lig ikke valgbar.
</PLI>	

8.5 HENT KLADDER EKSEMPEL

Hent rapporter i kladde status for beredskab 1000, hent rapport 41 og 42 ud af listen af kladde rapporter sorteret efter AlarmModtaget feltet.

Forespørgsel:

https://qa-odin.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/ReportSearch.svc/FindReports?beredskabsNummer=1000&rapportStatuses=%5B%22Kladde%22%5D&sortByPropertyName=AlarmModtaget&skip=40&take=2

Returdata:

```
{
  "Reports": [
    {
      "AarhusFakturering": false,
      "HasChanges": false,
      "Husnummer": null,
      "Id": 407,
      "Kommune": null,
      "ListId": "{cd8941eb-92b8-43a0-8054-a13b00f314f2}",
      "MeldingsOrdlyd": null,
      "RapportLink":
"/Udrykning/Sider/Rapport.aspx?XmlLocation=/Udrykning/Rapporter_1000/IGang/Udrykningsrapport
20150312 11-53-28/Udrykningsrapport 20150312 11-53-28.xml&blanketId=1&enhedsId=1000",
      "Status": "Kladde",
      "Tidspunkt": "",
      "Vejnavn": null
    },
    {
      "AarhusFakturering": false,
      "HasChanges": false,
      "Husnummer": null,
      "Id": 409,
      "Kommune": null,
      "ListId": "{cd8941eb-92b8-43a0-8054-a13b00f314f2}",
      "MeldingsOrdlyd": null,
      "RapportLink":
"/Udrykning/Sider/Rapport.aspx?XmlLocation=/Udrykning/Rapporter_1000/IGang/Udrykningsrapport
20150312 11-55-44/Udrykningsrapport 20150312 11-55-44.xml&blanketId=1&enhedsId=1000",
      "Status": "Kladde",

```



```

        "Tidspunkt": "",
        "Vejnavn": null
    },
    ],
    "TotalReportsCount": 61
}

```

Hvor det kan ses at første retur element f.eks. har id 407 og hvis man vil hente rapporten, f.eks. via `FetchIndberetning`, så skal der benyttes ”1000-407” som id.

Yderligere er der i dette eksempel 61 rapporter i alt med kladde status.

9 Appendiks A: Indberet script

Følgende PowerShell script udfører en fuld indberetning af en ny rapport.

Bemærk: Ved kopiering fra word skal man sikre sig at word ikke har ændret bindestreger til tankestreger, etc.

Placer nedenstående tekst i en .ps1 fil. Ret ”beredskabsnummer”. Scriptet kræver at der tastets credentials, når det køres.

9.1 EKSEMPEL 1

[...]

```

param(
    [string]$url = "https://qa-odin-ws.brs.dk"
)

try {
    Write-Host "Script '$($MyInvocation.MyCommand)' starting"

    Write-Host "Webservice test $url"

    $wsUrl =
"$url/_vti_bin/Delegate.BRS.NytInd/Services/indberetningSoap.svc/mex?singlewsdl"
    Write-Host "Creating web proxy at url $wsUrl"
    $cred = Get-Credential
    $ws = New-WebServiceProxy -credential $cred -Uri $wsUrl #To specify explicit
credentials.
    # $ws = New-WebServiceProxy -UseDefaultCredential -Uri $wsUrl

    if( !$ws ) {
        throw "Webservice proxy could not be created"
    }

    #NOTE: Kommentarer er ikke lovlige I JSON. Her fjernes de med et regex inden brug

```

#NOTE2: Apostrof skal escapes med \. For eksempel i adresse felter 'Christian 10\'s Vej', med mindre man omringer felter med gåsøjne, så kan man godt indberet med "Christian 10\'s Vej"

```

$jsonValuesWithPersons = "
    {'Udrykningsrapport' :{

<!-- ALARMERING -->
'rapportnummer': 'jb-test- KOM standard Komb. Redning/Evakuering',
'lokaltRapportnr': '1000-1-Redning',
'eksterntRapportnr': '112-1-Redning',
'nummer112': '112-4',
'alarmModtaget': '2014-10-27T10:00:00',
<!-- 112-PO -->
'alarmModtagetFra': '10200',
<!-- FUH fastklemte bil -->
'foersteMeldingsOrdlyd': 'rffa',
'foersteMeldingBemaerkning': 'test 112 - redning',
<!-- FUH færdselsuheld -->
'generiskMelding': '30100',
<!-- 112-politi -->
'datakilde': '20120',
'meldingsadresseKommune': 'Lyngby-Taarbæk Kommune',
'meldingsadresseKommuneKode': '173',
'meldingsadresseVejnavn': 'Jernbaneplassen',
'meldingsadresseVejkode': '0386',
'meldingsadresseHusnr': '45',
'meldingsadresseEtage': '',
'meldingsadressePostnr': '2800',
'meldingsadresseBy': 'Kgs.Lyngby',
<!-- x-koordinat -->
'meldingsadresseNorthing': '6185879',
<!-- y-koordinat -->
'meldingsadresseEasting': '719732',
'meldingsadresseStedUdenOfficielAdresse': 'Test til uoff. adresse - webservice',
'meldingsadresseUdenVejnavn':true,
<!-- Service niveau -->
'serviceniveauUdrykningstid': '10005',
'opgaveMoedeplan': 'true',
'oevelse': 'true',

<!-- SKADESTED -->
'skadesstedLigMelding': 'false',
'skadesstedKommune': 'Lyngby-Taarbæk Kommune',
'skadesstedKommuneKode': '173',
'skadesstedVejnavn': 'Jernbaneplassen',
'skadesstedVejkode': '0386',
'skadesstedHusnr': '22',
'skadesstedEtage': '',
'skadesstedPostnr': '2800',
'skadesstedBy': 'Kgs.Lyngby',
'skadesstedNorthing': '6185943',
'skadesstedEasting': '719673',
'skadesstedUdenVejnavn':false,
<!-- Reel -->
'alarmtype': '10100',
'opgaveLoestFoerAnkomst': 'false',
<!-- Skråning/Grøft -->
'haendelsesPlacering': '21400',
<!-- Trafikulykke -->
'haendelsestype': '10250',
'meldingPassende': 'true',
<!-- Nej -->
'brandsynsobjekt': 'Nej',

<!-- INDSATS -->
<!-- Personredning/Personevakuering -->
'opgaveHvadUdfoert': ['12100','12300'],
<!-- Bygning/Person -->
'objekt': '20100',

'Redning': {
<!-- Frigørelse teknisk - fastklemte-->
'redningsMetode': '11200',

```

```
'antalPersonerReddet': '2',
'personRedningBemaerkning': 'Test redning med halskrave',
'antalPersonerEvakueret': '3',
},
```

```
<!-- RESSOURCER KØRETØJER -->
'KoeretoerRepeater': [
{
'KoeretoerId': '56',
'KoeretoerStationId': '330',
'KoeretoerAntalPersoner': '1',
'KoeretoerUdrykningsArt': '12100',
'KoeretoerIndsatsOpgave': '12100',
'KoeretoerKoerselstype': '1',
'KoeretoerAlarmTidspunkt': '2014-10-27T10:00:01',
'KoeretoerAfgangTidspunkt': '2014-10-27T10:00:02',
'KoeretoerFremmeTidspunkt': '2014-10-27T10:00:03',
'KoeretoerFrigivetTidspunkt': '2014-10-27T10:00:04',
'KoeretoerKlarigenTidspunkt': '2014-10-27T10:00:05'
},
{
'KoeretoerId': '160',
'KoeretoerStationId': '330',
'KoeretoerAntalPersoner': '1',
'KoeretoerUdrykningsArt': '12100',
'KoeretoerIndsatsOpgave': '12100',
'KoeretoerKoerselstype': '1',
'KoeretoerAlarmTidspunkt': '2014-10-27T10:00:01',
'KoeretoerAfgangTidspunkt': '2014-10-27T10:00:02',
'KoeretoerFremmeTidspunkt': '2014-10-27T10:00:03',
'KoeretoerFrigivetTidspunkt': '2014-10-27T10:00:04',
'KoeretoerKlarigenTidspunkt': '2014-10-27T10:00:05'
}
],
```

```
<!-- RESSOURCER PERSONER -->
'PersonaleRepeater': [
{
<!-- GG - Alexander Murillo -->
'PersonId': '1617',
'PersonAlarmTidspunkt': '2014-10-27T10:00:01',
'PersonMoedtTidspunkt': '2014-10-27T10:00:02',
'PersonFrigivetTidspunkt': '2014-10-27T10:00:04',
'PersonOpgaveType': '12100',
'PersonFunktion': '10100',
'PersonStationId': '330',
'PersonKoeretoerId': '56',
},
{
<!-- CC - Orla Rasmussen -->
'PersonId': '1030',
'PersonAlarmTidspunkt': '2014-10-27T10:00:01',
'PersonMoedtTidspunkt': '2014-10-27T10:00:02',
'PersonFrigivetTidspunkt': '2014-10-27T10:00:04',
'PersonOpgaveType': '12100',
'PersonFunktion': '10100',
'PersonStationId': '330',
'PersonKoeretoerId': '160'
}
],
```

```
<!-- BESKRIVELSE -->
'beskrivelse': 'Beskrivelse test',
```

```
<!-- SPECIEL MATERIEL-->
'ResourcerSpecielMateriel': {
'specielMaterielTypeAnvendt': [
'20250', '20200', '20150' ]}
}"
```

```
$request = @{}
#This number MUST match your privileges
$request.BeredskabsNummer = 1061
$request.Status = "Kladde"
```

```

$request.Type = "StandardKommunalt"

#Remove the comments as they are not valid JSON
$jsonValues = $jsonValuesWithPersons -replace "<!--.*-->", ""
$request.ValuesJson = $jsonValues

Write-Host "Calling indberet webservice"
$response = $ws.Indberet($request)

Write-Host "Got a success response. ID of rapport: $($response.RapportId),
message: $($response.Message)"

if( $response.Errors -and $response.Errors.Count -gt 0 ){
    $e = ($response.Errors | out-string -width 400 )
    Write-warning "Got the following validation errors: $e"
}
}
catch {
    Write-Host "An exception occurred. Aborting. Exception : " $_
}

```

[...]

9.2 EKSEMPEL 2

Følgende script indberetter en rapport, hvor der er valgt både køretøjs og stations ressourcer.

Specielt bemærkes at ”køretøjsid’er” for stationer sættes til ”stationens id+900000”.

Bemærk yderligere at ID’er og beredskabsnummer skal justeres efter miljø der testes imod, og hvilke rettigheder der haves.



WebServiceTest-102
7-K-Ressourcer.ps1

10 Appendiks B: En fuld rapport

En liste af afsluttede rapporter kan f.eks. hentes me: https://qa-odin.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/ReportSearch.svc/SearchReports?beredskabsNummer=1000&rapportStatuses=%5B%22Afsluttet%22%5D&sortByPropertyName=&sortDirection=0&refiners=&take=20

```

{
  "Reports": [
    {
      "AarhusFakturering": false,
      "HasChanges": false,
      "Husnummer": "2",
      "Id": 341,
      "Kommune": "Slagelse Kommune",
      "ListId": "{cd8941eb-92b8-43a0-8054-a13b00f314f2}",

```

```

        "MeldingsOrdlyd": null,
        "RapportLink":
"/Udrykning/Sider/Rapport.aspx?XmlLocation=/Udrykning/Rapporter_1000/IGan
g/Udrykningsrapport 20150305 10-28-18/Udrykningsrapport 20150305 10-28-
18.xml&blanketId=1&enhedsId=1000",
        "Rapportnummer": "jb-test - STAT IBR International indsats",
        "Status": "Afsluttet",
        "Tidspunkt": "2015-03-01T21:00:00",
        "Vejnavn": "Slagelse Landevej"
    },
    {
        ...
    },
    {
        ...
    },
    {
        "AarhusFakturering": false,
        "HasChanges": false,
        "Husnummer": "24",
        "Id": 1950,
        "Kommune": "Rudersdal Kommune",
        "ListId": "{cd8941eb-92b8-43a0-8054-a13b00f314f2}",
        "MeldingsOrdlyd": null,
        "RapportLink":
"/Udrykning/Sider/Rapport.aspx?XmlLocation=/Udrykning/Rapporter_1000/IGan
g/Udrykningsrapport 20160701 15-06-19/Udrykningsrapport 20160701 15-06-
19.xml&blanketId=1&enhedsId=1000",
        "Rapportnummer": "Test Afgang- udrykningstider Nonnemann",
        "Status": "Afsluttet",
        "Tidspunkt": "2016-07-01T15:11:31",
        "Vejnavn": "Datavej"
    }
],
    "TotalReportsCount": 4
}

```

Og en konkret rapport kan hentes med "FetchIndberetning" med (1000 = beredskabsnummer, 341 = rapportens lokale id) https://qa-odin.brs.dk/vti_bin/Delegate.BRS.NytInd/Services/indberetning.svc/FetchIndberetning/1000-341

```

{
  "Metadata": {
    "BrsId": "1000-341",
    "BeredskabNumber": "1000",
    "BeredskabName": "Beredskabsstyrelsen",
    "BeredskabType": "Statsligt",
    "RapportType": "IBR",
    "RapportStatus": "Afsluttet",
    "ForrigeRapportStatus": "Kladde",
    "BeredskabTypeLowercase": "statsligt",
    "RapportTypeLowercase": "ibr",
    "IsStatsligt": "true",

```

```

    "IsKommunalt": "false",
    "IsStandardKommunalt": "false",
    "IsAssisterende": "false",
    "IsAftale": "false",
    "IsAssisterenceOrAftale": "false",
    "IsNuc": "false",
    "IsIBR": "true",
    "IsKemi": "false",
    "DateFilterForResources": "2015-03-01T21:00:00",
    "KoeretoeyIDs": "543,",
    "StationIDs": "607,",
    "Responstid": null,
    "Afgangstid": null,
    "ResponstidSecs": null,
    "AfgangstidSecs": null,
  },
  "Stamdata": null,
  "Alarmering": {
    "Meldingsadresse": {
      "mkort": null,
      "madresse": {
        "meldingsadresseKommune": "Slagelse Kommune",
        "meldingsadresseKommuneKode": "330",
        "meldingsadresseVejnavn": "Slagelse Landevej",
        "meldingsadresseVejKode": "1534",
        "meldingsadresseHusnr": "2",
        "meldingsadresseBogstav": null,
        "meldingsadresseEtage": null,
        "meldingsadressePostnr": "4200",
        "meldingsadresseBy": "Slagelse",
        "meldingsadresseEasting": "6514077",
        "meldingsadresseNorthing": "61390752",
        "meldingsadresseStedUdenOfficielAdresse": "Test til uoff. adresse -
webservice",
        "meldingsadresseUdenVejnavn": "false"
      },
      "mSkjulVejnavn": null
    },
    "fremmedDaekning": "false",
    "rapportKategori": "10100",
    "Oevelse": "false",
    "betalingInternational": "true",
    "IVAssistance": "false",
    "planlagtAssistance": "true",
    "opgaveAfgangskrav": "10005",
    "opgaveMoedeplan": "false",
    "serviceniveauUdrykningstid": "10060",
    "datakilde": "20400",
    "generiskMelding": null,
    "brsGruppeGeneriskMelding": null,
    "generiskMeldingAndet": null,
    "foersteMeldingBemaerkning": "test IBR webservice",
    "foersteMeldingsOrdlyd": null,
    "brsGruppeFoersteMeldingsOrdlyd": null,
    "rekvirent": "30050",
    "alarmModtagetFra": null,
    "beredskabsniveau": "11100",
    "rekvisitionsTidspunkt": "2013-12-15T23:55:00",
    "alarmModtaget": "2015-03-01T21:00:00",
  }
}

```

```

"Advarsel": null,
"nummer112": null,
"eksterntRapportnr": null,
"lokaltRapportnr": null,
"rapportnummer": "jlb-test - STAT IBR International indsats"
},
"Skadestedet": {
  "kort": null,
  "adresse": {
    "skadestedKommune": "Sønderborg Kommune",
    "skadestedKommuneKode": "0540",
    "skadestedVejnavn": "Hagensig",
    "skadestedVejkode": "0678",
    "skadestedHusnr": "3",
    "skadestedBogstav": null,
    "skadestedEtage": null,
    "skadestedPostnr": "6470",
    "skadestedBy": "Sydals",
    "skadestedEasting": "559651",
    "skadestedNorthing": "6080294",
    "skadestedStedUdenOfficielAdresse": "Test til uoff. adresse -
webservices",
    "skadestedUdenVejnavn": "false"
  },
  "SkadesstedLigMelding": "false",
  "Sikringsanlaeg": {
    "sikringAnlaegNr": null,
    "sikringDetektorGruppeNr": null,
    "sikringDetektorNr": null,
    "sikringDetektorType": null,
    "sikringUdloestAarsag": null,
    "sikringUdloestAarsagAndet": null,
    "sikringDetektorTypeAndet": null
  },
  "sSkjulVejnavn": null,
  "brandsynsobjekt": null,
  "meldingPassende": null,
  "haendelsestype": "10150",
  "haendelsestypeAndet": null,
  "haendelsesPlacering": "21400",
  "brsGroupHaendelsesPlacering": "21000",
  "haendelsesPlaceringAndet": null,
  "alarmtype": "10100",
  "opgaveLoestFoerAnkomst": "false"
},
"Indsats": {
  "MetadataIndsats": null,
  "Brand": {
    "Bygningsbrand": {
      "brandTekniskUdstyrAnvendt": null,
      "brandTekniskUdstyrVirkede": null,
      "brandTekniskUdstyrBrugt": null,
      "brandTekniskUdstyrBrugtAndet": null,
      "brandSpredningAarsag": null,
      "brandSpredningAndet": null,
      "brandSpredning": null,
      "brandStartRum": null,
      "brandStartRumAndet": null,
      "brandForloebAnkomst": null
    }
  }
}

```

```

    },
    "brandKlassificering": null,
    "brandSaerligeForhold": "false",
    "brandVandkilde": null,
    "brandVandkildeAndet": null,
    "brandVandforbrug": null,
    "brandVarmeKilde": null,
    "brandTypeAfMaterialeFoerstAntaendt": null,
    "brandObjektMaterialeAnvendelse": null,
    "brandUdstyrInvolveretIAntaendelse": null,
    "brandVarmeKildeAndet": null,
    "brandUdstyrInvolveretMaerkeVareNummer": null,
    "brandUdstyrInvolveretIAntaendelseAndet": null,
    "brandObjektMaterialeAnvendelseAndet": null,
    "brandTypeAfMaterialeFoerstAntaendtAndet": null,
    "brandFormDirekteAarsagsType": null,
    "hideBrandMedvirkFaktor": "2",
    "hideMenneskeligeFaktorer": "3",
    "hideBrandMedvirkFaktorAndet": "1",
    "hideBrandMenneskeligeFaktorerAarsagTilAntaendelseAndet": "4",
    "brandMenneskeligeFaktorerAarsagTilAntaendelse": null,
    "brandMenneskeligeFaktorerAarsagTilAntaendelseAndet": null,
    "brandMedvirkFaktor": null,
    "brandMedvirkFaktorAndet": null
  },
  "Redning": {
    "antalDyrReddet": null,
    "antalPersonerEvakueret": null,
    "redningsMetode": null,
    "antalPersonerReddet": null,
    "personRedningBemaerkning": null,
    "redningsMetodeAndet": null
  },
  "Miljoe": {
    "StofRepeater": {
      "farligeStofferUnNr": null,
      "forureningStofNavn": null,
      "forureningMaengdeEnhed": null,
      "forureningMaengdeUdstroemmet": null,
      "forureningMaengdeOpsamlet": null,
      "forureningStofNavnAndet": null
    },
    "forureningAarsagTilUheld": null,
    "forureningAarsagTilUheldAndet": null
  },
  "antalBrandOmkomne": "0",
  "countOmkomne": "0",
  "BrandOmkomneRepeater": {
    "skadeartBrandOmkomne": null,
    "omkommetKendtAfBeredskabet": null,
    "bemaerkningOmkommet": null,
    "skadeartOmkomneAndet": null
  },
  "antalBrandTilskadekomne": "0",
  "countTilskadekomne": "0",
  "BrandTilskadekomneRepeater": {
    "skadeartBrand": null,
    "kendtAfBeredskabet": null,
    "bemaerkning": null,

```



```

        "tilskadekommenAndet": null
    },
    "objekt": "70100",
    "objektAndet": null,
    "opgaveHvadUdfoert": "18400",
    "opgaveHvadUdfoertAndet": null
},
"Beskrivelse": {
    "beskrivelse": "Beskrivelse test JB Beskrivelse test JB Beskrivelse test
JB",
    "fileUpload": {
        "group1": {
            "group2": {
                "fil": null,
                "field6": null
            }
        }
    },
    "aarhusFakturering": "false",
},
"Ressourcer": {
    "KoeretoerRepeater": {
        "KoeretoerId": "543",
        "KoeretoerName": "11.000 - Passat",
        "KoeretoerAntalPersoner": "1",
        "KoeretoerStation": "Beredskabsstyrelsen Nordjylland",
        "KoeretoerStationId": "607",
        "KoeretoerType": null,
        "KoeretoerTypeId": null,
        "KoeretoerGruppe": null,
        "KoeretoerUdrykningsArt": "12100",
        "KoeretoerIndsatsOpgave": "12100",
        "KoeretoerKoerselsType": "1",
        "KoeretoerAlarmTidspunkt": "2016-05-18T00:00:00",
        "KoeretoerAfgangTidspunkt": "2016-05-18T01:00:00",
        "KoeretoerFremmeTidspunkt": "2016-05-18T02:00:00",
        "KoeretoerFrigivetTidspunkt": "2016-05-18T03:00:00",
        "KoeretoerKlarigenTidspunkt": "2016-05-18T04:00:00",
        "KoeretoerResponstid": "02:00:00",
        "KoeretoerAfgangstid": "01:00:00",
        "KoeretoerAfgangstidSecs": "3600",
        "KoeretoerResponstidSecs": "7200",
        "KoeretoerSupplerendeTekst": null
    }
},
"RessourcerPersoner": {
    "antalAfPersoner": "1",
    "indsatsleder": null,
    "holdleder": "1752",
    "PersonaleRepeater": {
        "PersonKoeretoerName": "11.000 - Passat",
        "PersonId": "1752",
        "PersonNavn": null,
        "PersonAlarmTidspunkt": "2016-05-19T02:00:00",
        "PersonMoedtTidspunkt": "2016-05-19T02:30:00",
        "PersonFrigivetTidspunkt": "2016-05-19T02:32:00",
        "PersonOpgaveType": "10100",
        "PersonFunktion": "10150",
        "PersonStationId": "607",
    }
}

```

```
    "PersonKoeretoejId": "543",
    "KTTid": "2016-05-18T00:00:00",
    "KTIndsatsOpgave": "12100",
  }
},
"ResourcerSpecialMateriel": {
  "specialMaterielTypeAnvendt": [
    "20250",
    "20150",
    "20200"
  ],
},
"Kobling": {
  "Koblede": {
    "KobledeRapporter": null
  },
  "koblingMessageNothingFound": null,
},
"FrivilligeFelter": {
  "felt10": null,
  "felt9": null,
  "felt8": null,
  "felt7": null,
  "felt6": null,
  "felt5": null,
  "felt4": null,
  "felt3": null,
  "felt2": null,
  "felt1": null
},
"Opsummering": null,
"Underskrift": null
}
```

Hvortil det skal bemærkes at alle felter under ”Metadata” ikke kan indberettes, og at filer heller ikke bør indberettes – de skal tilføjes med AddFile metoden.

11 Appendiks C: Alle felter i rapporten

Beskrivelse af de enkelte felterne i en rapport findes i et regneark, der kan hentes fra: [TODO].