

## Tillate oppdatering av entiteter med PATCH

---

Prosjekt	NOARK 5 Tjenestegresesnitt
Kategori	Versjon 1.0 beta
Alvorlighet	kommentar
Meldingstype	utelatt
Brukerreferanse	thomas.sodring@hioa.no
Dokumentdel	6.1.1.5
Sidenummer	19
Linjenummer	n/a
Innsendingsdato	2017-03-26

---

### Beskrivelse

Når det gjelder komposisjoner som er beskrevet på side 18 så står det «Heleide objekter (komposisjoner) kan opprettes sammen med hovedobjektet».

Det må også være mulig å opprette/oppdatere slike komposisjoner på et senere tidspunkt. Valget dokumentert i spesifikasjonen står da mellom en PUT på hele objektet der objektet inngår eller at det er egne relasjonslenker for slike operasjoner.

Slik vi forstår tjenestegresesnittet er det mening at hele objektet skal lastes fram og tilbake og kjernen må ta stilling til om felter er blitt endret. Dette kompliseres enda mer ved at det er endel felter det ikke er lov å endre etter at objektet er opprettet/avsluttet.

Man løper en stor dataintegritetsrisiko hvis man tvinger klienten til å holde hele objektet med alle komposisjoner i minne og sende hele objektet tilbake til kjernen ved mindre oppdateringer. Dette mener vi er unødvendig og en dårlig tilnærming. Man må huske at Riksarkivet godkjenner Noark 5 komplett/kjerne, ikke forsystemene som skal integreres med en Noark 5 kjerne. Her er et eksempel for å forklare problemet.

La oss anta følgende situasjon. Et fagsystem for behandling av barnehagesøknader er integrert med et Noark 5 saksbehandlingsystem. Søknaden mottas via postmottak og det opprettes en saksmappe. Fagsystemet går da gjennom disse saksmappene og behandler søknader. Følgende data sendes fra kjernen til fagsystemet (GET):

```
{
  "systemId" : "ad6d2092-180f-46d7-a631-ba679f875fd0" ,
  "opprettetDato" : "2017-03-03T13:21:34",
  "opprettetAv" : "admin",
  "offentligTittel": "Public title of the test file",
  "mappeID": "2017/01",
```

```

"saksstatus" : "Opprettet",
"tittel": "Title of the test file",
"beskrivelse": "Søknad om bhg plass i Eksempel kommune",
"noekkelord": ["keyword 1", "keyword 2","keyword 3" ],
"dokumentmedium": "Elektronisk arkiv"
}

```

Men fagsystemet som utgjør klienten er dårlig laget. Fagsystemet er ikke er i stand til å forstå feltet «noekkelord», som fører til at når JSON-innholdet mottas av fagsystemet så droppes feltet «noekkelord» med tilhørende verdi. Senere laster fagsystemet opp følgende data til kjernen med PUT:

```

{
  "systemId" : "ad6d2092-180f-46d7-a631-ba679f875fd0" ,
  "opprettetDato" : "2017-03-03T13:21:34",
  "opprettetAv" : "admin",
  "offentligTittel": "Public title of the test file",
  "mappeID": "2017/01",
  "saksstatus" : "Avsluttet",
  "tittel": "Title of the test file",
  "beskrivelse": "Søknad om bhg plass i Eksempel kommune",
  "dokumentmedium": "Elektronisk arkiv"
}

```

Nå er feltet «noekkelord» med tilhørende verdi forsvunnet. Kjernen har ingen grunn til å tro at det ikke var meningen at «noekkelord» skulle forsvinne. Dette er ikke bare et problem for komposisjoner, men kan også gjelde felter. Poenget er at kjernen tvinger klienten å være inneforstått med hele datamodellen til Noark 5, noe som er unødvendig. Riksarkivet godkjenner ikke forsystemene som skal integreres til en frittstående kjerne, slik at det ikke er åpning for riksarkivet å oppdage slike problemer før slike klienter tas i bruk.

Et relatert eksempel er med feltet «opprettetDato» der «opprettetDato» kan være forskjellig for kjernen og fagsystemet. Det er legitimt at fagsystemet velger å sette «opprettetDato» til tidspunktet mappen ble opprettet i fagsystemet.

```

{
  "systemId" : "ad6d2092-180f-46d7-a631-ba679f875fd0" ,
  "opprettetDato" : "2017-03-04T09:23:14",
  "opprettetAv" : "fagsystemadmin",
  "offentligTittel": "Public title of the test file",
  "mappeID": "2017/01",
  "saksstatus" : "Opprettet",
  "tittel": "Title of the test file",
  "beskrivelse": "Plassen er tildelt i duestien barnehage",
  "oppbevaringsted" : [ "location 1", "location2", "location3" ],
  "dokumentmedium": "Elektronisk arkiv"
}

```

Det er ikke mulig å endre «opprettetDato» og «opprettetDatoAv» i Noark 5 så denne forespørslen ville måtte avvises. Det er allikevel en del logikk som må bygges inn i kjernen for å sjekke hvilken felte er blitt endret og om det er lov å tillate en slik endring. En dårlig Noark 5 kjerne ville kanskje til og med tillate en slik endring!

Det ville være mye enklere å be klienten angi hvilken felte som skal endres. I eksemplet over er det beskrivelse som ble endret. Da kunne det være en PATCH forespørsel der kun beskrivelse inngikk.

En PATCH forespørsel for å endre «beskrivelse» for følgende mappe-instans [contextPath][api]/arkivstruktur/mappe/ad6d2092-180f-46d7-a631-ba679f875fd0 ville da bestå av følgende JSON

```
{ "op": "replace", "path": "/beskrivelse",  
  "value": "Plassen er tildelt i duestien barnehage" }
```

Dette er en veldig tydelig og ryddig måte å angi hva som skal endres og med en slik strategi vil det være mye raskere og enklere å avvise uønskete endringer samtidig som klienten tvinges kun til forholde seg til de feltene den har behov for å vite noe om.

IETF har [standardisert PATCH-forespørsler av JSON i RFC 6902](#) og dette er noe som med fordel kunne brukes i tjenestegrensesnittet.

Tjenestegrensesnitt i sitt nåværende form virker å være utviklet utifra et «Noark 5-komplett»-synspunkt, framfor synspunktet, «frittstående kjerne» som kan integreres med sak/arkiv og fagsystemer. Et Noark 5-komplett system vil ha full kontroll både på klienten og kjernen og derfor vil nok en del av problemene over ikke være relevant. Når det gjelder en Noark 5-kjerne med integrasjoner til fagsystem vil Riksarkivet kun godkjenne kjernen, ikke klienter, og vi tror det er viktig å ta inn over seg forskjellen. Eksisterende Noark 5-komplett systemer blir godkjent som en helhet. En frittstående kjerne med integrasjoner til fagsystemer vil stå over mange flere utfordringer når det gjelder datakvaliteten hvis data unødvendig flyttes fram og tilbake mellom klienten og kjernen. Ulike klienter vil ikke nødvendigvis forstå viktigheten og betydningen av Noark 5-metadata og sammenhenger mellom entiteter!

For å understøtte argumentet at kjernen må støtte PATCH. Tjenestegrensesnittet sier selv at noen steder bygges det på OData. [OData-retningslinjene](#) sier følgende for oppdateringer:

Update an Entity

*The OData services SHOULD support PATCH as the preferred means of updating an entity. But also services MAY additionally support PUT.*

Vi mener prinsippet som ligger til grunn for en slik anbefaling er at du skal isolere og kun jobbe med data som trenger oppdatering, framfor å risikere uønsket hendelser på data.

Hvis vi forstår OData anbefalingene riktig så anbefales det følgende strategi. Entiteter opprettes / oppdateres (i sin helhet) med POST/PUT mens felter oppdateres med en PATCH, der feltets attributtnavn og ny verdi oppgis.

Det må da tas stilling til hvordan komposisjoner opprettes og oppdateres. I tjenestegrensesnittet opprettes komposisjoner sammen med entiteten.

Vi ser ikke noe problem med nøstet komposisjoner når entiteter opprettes og det står ingenting om det i [Odata-anbefalingene](#), men når vi ser på [OData anbefalinger for oppdateringer](#) står det følgende:

The entity MUST NOT contain related entities as inline content. It MAY contain binding information for navigation properties.

Hvis komposisjoner skal omfattes av OData anbefalingene så må det utvikles nye relasjons-URL-er for alle entiteter med komposisjoner. Slik vi forstår det, vil en tilknytting av en komposisjon til en entitet være en oppdatering av entiteten og da vil det måtte brukes en PATCH forespørsel.

Nye REL som trengs da for feks klasse vil være:

---

REST_REL	<a href="http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-skjerming">http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-skjerming</a>
REST_REL	<a href="http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-gradering">http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-gradering</a>
REST_REL	<a href="http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-sletting">http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-sletting</a>
REST_REL	<a href="http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-kassasjon">http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-kassasjon</a>
REST_REL	<a href="http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-kassasjonutfoert">http://rel.kxml.no/noark5/v4/api/arkivstruktur/ny-kassasjonutfoert</a>

---

## Ønsket endring

Det er veldig omfattende å skrive et ord-for-ord beskrivelse av alle endringene som trengs. Vi nøyer oss derfor i denne omgang med en kort oppsummering.

Den første endringen vi ønsker er at tjenestegrensesnittet bruker PATCH og tillater feltoppdateringer. Den andre endringen er at RFC-6902 brukes som metode for å støtte oppdateringer. Den tredje endringen er at tjenestegrensesnittet følger den overnevnt identifiserte beste praksis for CRUD. Selv om tjenestegrensesnittet ikke defineres utelukkende som en OData kilde så kan OData-standarden brukes som veiledende. Vi foreslår at det utvikles en beskrivelse av hvilke beste praksis-spesifikasjoner tjenestegrensesnittet forholder seg til, eventuelt om det er avvik til etablerte tilnærminger og hvorfor.

OData som beste praksis gir en generell endring i hvordan CRUD beskrives:

1. En entitet opprettes med en POST.
2. En entitet oppdateres, i sin helhet, med en PUT.
3. Enkeltfelt eller utvalgte felt oppdateres med en PATCH.
4. Komposisjoner nøstes ikke ved oppdateringer
5. Komposisjoner knyttes til entiteter via PATCH