

Tekniska förbättringar apotek

Förbättrad läsårskomst NLL

2023-05-30

Version 1

Inledning

Agenda

- Bakgrund
- Lösningssidéer
- Diskussion
- Avslutning

Bakgrund

- Fördjupad dialog apotek avseende genomgång av möjliga förbättringar hämtatjänster.
- Vi har tidigare presenterat möjliga förbättringar avseende hämtatjänster till NLL. Vi behöver få er återkoppling utifrån era behov och prioritering när det gäller dessa.
- På detta möte kommer vi göra en fördjupad genomgång i vad dessa möjliga förbättringar innebär ur tekniskt och nyttoperspektiv.
- Inbjudan är öppen för alla intresserade men skicka gärna inbjudan vidare till lämpliga personer i er organisation.
- Det finns olika sätt att hämta information och det är upp till varje systemleverantör att välja det sätt som passar bäst.

Bakgrund

Det har framförts önskemål om att göra tekniska förändringar för att:

- Minska antalet anrop som behövs för att skapa en patientöversikt.
- Bättre sökfunktioner för att slippa hämta onödig information.

Vid val av lösning bör följande principer beaktas.

- Bygga vidare på existerande arkitektur.
- Återanvända standardfunktioner i FHIR om möjligt.
- Vara generiskt och återanvändningsbart.
- Medföra en rimlig arbetsinsats och leveranstid.

Lösningssidéer

- Hämta allt via patientresursen.
 - Sökning på Patient med reverse include för att hämta övriga resurser i ett anrop.
- Hämta allt via en utökad operation.
 - Sökning på patientidentifierare via en utökad operation som returnerar patientinfo samt utvalda underliggande resurser.
- Hämta resurs samt överordnad resurs och händelse.
 - Sökning på en resurs med include för att hämta Patient / MedicationRequest och reverse include för att hämta Provenance.

Hämta allt via patientresursen

Om man utgår från existerande lösning i NLL-FHIR och använder "reverse include" för att hämta mer än en resurstyp skulle man i teorin kunna hämta all information i ett anrop med utgångspunkt från patientresursen.

Problemet med det är bl a att det inte går att filtrera ut vilka underliggande resurser man vill ha tillbaka. Man får alla resurser som har en koppling till patienten.

Fungerar inte

Hämta allt via patientresursen

```
GET [base]/Patient?identifier=http://electronichealth.se/identifier/  
personnummer|[pnr]
```

```
&_revinclude=Consent:patient
```

```
&_revinclude=Basic:subject
```

```
&_revinclude=MedicationRequest:patient
```

```
&_revinclude:iterate=Provenance:target:MedicationRequest
```

```
&_revinclude=MedicationDispense:patient
```

```
&_revinclude:iterate=Provenance:target:MedicationDispense
```



Fungerar inte

Hämta allt via en utökad operation

Om man skapar en utökad operation som tar en patientidentifierare som indata och returnerar utvalda underliggande resurser går det att skraddarsy en lösning.

Fördelar

- Skräddarsydd lösning för vårdens behov som reducerar antalet anrop.

Bör undvikas
om möjligt

Nackdelar

- Oflexibel lösning.
 - Svårare att anpassa för olika behov och olika användningsområden.
 - Svårare att vidareutveckla.
 - Kan vara svårt att enas om ett gemensamt upplägg.
- Etablerar ett nytt mönster som inte går att kombinera med existerande lösning.
- Motverkar syftet att endast relevant information ska hämtas.

Hämta resurs samt händelse ...

Om man använder include kan man reducera antalet anrop. Det går att filtrera på den resurs sökning baseras på.

Fördelar

- Generell lösning som kan återanvändas för olika användningsområden.
- Bygger på standardfunktioner i FHIR vilket underlättar anslutning och integration mellan system.
- Bygger vidare på befintlig lösning vilket kortar ner leveranstiden och arbetsmängden.
- Antalet anrop till NLL-FHIR reduceras.
- Det går att lägga till fler sökvillkor för att möta olika behov.

Förstahandsvalet

Nackdelar

- Det krävs mer än ett anrop.

Hämta resurs samt händelse ...

Hämta patient

All informationshämtning som rör en patient bygger på att man använder patientens logiska ID. Det får man tag på genom att hämta patientresursen.

Inga förändringar.

```
GET [base]/Patient?identifier=http://electronichealth.se/  
identifier/personnummer|[pnr]
```

Hämta resurs samt händelse ...

Sök samtycken

Sök samtycken samt inkludera patient och händelsen som hör ihop med respektive samtycke.

GET [base]/Consent?patient._id=[id]&[params]

GET [base]/Consent?patient=[id]&[params] ny

GET [base]/Patient/[id]/Consent?[params] ny för åtkomstsamtycke

Valfria parametrar:

&category=INFA,consent-dose-disp

ändra så parametern är valfri

&status=[string]

ny för dossamtycke

&actor.type=[string]

ändra till generell parameter (endast åtkomstsamtycken)

&practitioner.identifier=[string]

ändra till generell parameter (endast åtkomstsamtycken)

&_include=Consent:patient

ny

&_reinclude=Provenance:target:Consent

ny

Hämta resurs samt händelse ...

Hämta samtycken (read)

GET [base]/Consent/[id] ...

/_history

/_history/[vid]

?_include=Consent:patient

?_revinclude=Provenance:target:Consent

ny för dossamtycke

ny

ny

Hämta resurs samt händelse ...

Sök "övrigt" (Basic)

Sök "övrigt" dvs spärr, läkemedelslista, dosunderlag (Basic) samt inkludera patient och händelsen som hör ihop med respektive "övrigt" resurs.

```
GET [base]/Basic ...
```

```
?patient._id=[id]&[params]
```

```
?patient=[id]&[params]
```

ny

```
?nllPrescriptionChain=[UUID]&[params]
```

ändra till generell parameter (endast spärr)

```
GET [base]/Patient/[id] ...
```

```
/MedicationListInfo
```

utgår och ersätts av Basic

```
/Basic?[params]
```

ersätter MedicationListInfo

Hämta resurs samt händelse ...

Sök "övrigt" (Basic) forts.

Valfria parametrar:

`&code=multidosedispbasis,medlistinfo,datalock` ändra till valfri parameter

`&nllStatus=[string]`

`&nllPrescriptionChain=[id]` ändra till generell parameter (endast spärr)

`&nllLockedSubjectType=[string]` ändra till generell parameter (endast spärr)

`&nllDataLockLegalGround=[string]` ändra till generell parameter (endast spärr)

`&_include=Basic:patient` ny

`&_reinclude=Provenance:target:Basic` ny

Hämta resurs samt händelse ...

Hämta "övrigt" (read Basic)

GET [base]/Basic/[id] ...

/_history

används ej för läkemedelslistan

/_history/[vid]

ny för dosunderlag, ej läkemedelslistan

?_include=Basic:patient

ny

?_revinclude=Provenance:target:Basic

ny

Hämta resurs samt händelse ...

Sök förskrivningar

Sök förskrivningar samt inkludera patient och händelsen som hör ihop med respektive förskrivning.

```
GET [base]/MedicationRequest ...
```

```
?patient._id=[id]&[params]
```

```
?patient=[id]&[params] ny
```

```
?nllPrescriptionChain=[UUID]&[params]
```

```
?nllReceivingPharmacy.identifier=[GLN]&
```

```
patient.birthdate=[date]&[params] ändra namn från birthdate till patient.birthdate
```

```
?prescriptionSetId=[?]&[params] ändra så man kan använda alla valfria parametrar
```

```
?provenance._id=[id]&[params]
```

```
?nllOrdinationsId=[id]&[params] ändra så man kan använda alla valfria parametrar
```

```
GET [base]/Patient/[id]/MedicationRequest?[params]
```

```
GET [base]/Provenance/[id]/MedicationRequest
```

Hämta resurs samt händelse ...

Sök förskrivningar forts.

```
GET [base]/MedicationRequest?  
nllReceivingPharmacy.identifier=[GLN]&authoredon=ge[date] ta bort*
```

* Dolda sökbegrepp/filter appliceras när nllReceivingPharmacy.identifier kombineras med authoredon. En bättre lösning är att skapa en utökad operation för hämta recept för födelsedatumpatient.

Hämta resurs samt händelse ...

Sök förskrivningar forts.

Valfria parametrar:

`&nllprescriptionChain=[UUID]`

`&status=[code]`

`&authoredon={ge|le|eq}[date]`

`&_include=MedicationRequest:patient`

`&_revinclude=Provenance:target:MedicationRequest`

lägg till så eq stöds

ny

ny

Hämta resurs samt händelse ...

Hämta förskrivningar (read)

GET [base]/MedicationRequest/[id] ...

/_history

/_history/[vid]

?_include=Basic:patient ny

?_revinclude=Provenance:target:MedicationRequest ny

?_revinclude=MedicationDispense:prescription ny

Hämta resurs samt händelse ...

Sök uttag

Sök uttag samt inkludera patient och händelsen som hör ihop med respektive uttag.

GET [base]/MedicationDispense ...

?patient._id=[id]&[params]

?patient=[id]&[params]

variant av föregående

?medicationrequest._id=[id]&[params]

utgår och ersätts av prescription

?prescription._id=[id]&[params]

ersätter medicationrequest

?prescription=[id]&[params]

variant av föregående

GET [base]/Patient/[id]/MedicationDispense?[params] ny

GET [base]/MedicationRequest/[id]/MedicationDispense?[params]

Hämta resurs samt händelse ...

Sök uttag forts.

Valfria parametrar:

<code>&whenhandedover=ge[date]</code>	eventuellt borde även le och eq stödjas
<code>&prescription.status=[string]</code>	ny
<code>&hasAuthorizingPrescription=true false</code>	utgår och ersätts av prescription:missing
<code>&prescription:missing=true false</code>	ersätter hasAuthorizingPrescription
<code>&_count=[number]</code>	ny
<code>&_sort=[search parameter]</code>	ny, sortering på whenhandedover och status
<code>&_include=MedicationDispense:patient</code>	ny
<code>&_include=MedicationDispense:prescription</code>	ny
<code>&_include:iterate=MedicationRequest:patient</code>	ny
<code>&_revinclude=Provenance:target:MedicationDispense</code>	ny

Hämta resurs samt händelse ...

Hämta förskrivningar (read)

```
GET [base]/MedicationDispense/[id] ...
```

```
  /_history
```

```
  /_history/[vid]
```

```
  ?_include=MedicationDispense:patient ny
```

```
  ?_revinclude=Provenance:target:MedicationDispense ny
```

Hämta resurs samt händelse ...

Sök händelse

Sök händelser

```
GET [base]/Provenance/[id]
```

```
GET [base]/Provenance?target=[ref]{&activity-type=[code]}
```

Sök händelser via Compartment.

```
GET [base]/Consent/[id]/Provenances
```

```
GET [base]/Basic/[id]/Provenance
```

inkl ej läkemedelslistan

```
GET [base]/MedicationRequest/[id]/Provenance
```

```
GET [base]/MedicationDispense/[id]/Provenance
```

Slut