



Det Fælles Medicinkort

Ideoplæg til versionering med events

(1.2.6 funktionalitet)

2011-09-16



Indhold

Formål.....	3
Events.....	3
Event: source, action og result	3
Eventliste.....	4
Services.....	5
Eksempler på EventRequest.....	5
Eksempel på EventResponse.....	6
Ekstensibilitet.....	7
Source- og Result-elementer.....	8

Formål

I dag består FMKs versionsnummering af at versionsnummeret tælles op når der sker en klinisk ændring på medicinkortet eller lægemiddelordinationen. Oprettelser eller ændringer på recepter, dvs. effektueringer eller lignende påvirker ikke versionsnummeret. Desuden er det ikke utænkeligt, at der i fremtiden ville kunne foretages ændringer i andre systemer, der heller ikke medfører en klinisk ændring af medicinkortet, eksempelvis fra hjemmesygeplejen.

For at give klientsystemerne mulighed for at uddrage information omkring hvad der er sket af ændringer relateret til en patients medicinkort, kan der udstilles en række "events".

Events

Ved hver ændring af data på et CPR-nummer skrives der en række data til en eventlog knyttet til CPR-nummeret. Eksempelvis:

- Tidspunkt for ændring
- Type af ændring (lægemiddelordination oprettet, lægemiddelordination opdateret, recept oprettet, recept status ændret, ...)
- Hvilke data der er anvendt som udgangspunkt (medicinkortets versionsstreng, evt. lægemiddelordinationens id og versionsstreng, ...)
- Hvilke data der er fremkommet som resultat af ændringen (medicinkortets nye versionsstreng, evt. lægemiddelordinationens id og nye versionsstreng, ...)

Klientsystemet kan hente eventloggen på et CPR-nummer. Klientsystemet kan filtrere eventloggen for de events, der gør at systemet vil hente den tidligere udgave af lægemiddelordinationen eller medicinkortet.

Ud over events på FMK vil eventloggen indholde events for oprettelse af recept, oprettelse af effektuering, ændring af receptstatus m.v. Hermed opnås yderligere en funktionalitet, der har været efterspurgt, men som FMK indtil nu ikke har kunne tilbyde: I dag er det forud bestemt hvilke ændringer der medfører at medicinkortets versionsnummer opdateres, og det er stort set den eneste måde, hvorpå FMK kan signalere at der er sket en ændring på medicinkortet, således at klientsystemet skal hente en ny version. Med eventloggen er det muligt at give en meget mere detaljeret besked herom, og også på oprettelser og ændringer af f.eks. recepter, effektueringer og senere også andre dataelementer, der ikke kendes i dag.

Event: source, action og result

I forbindelse med forberedelse af FMK til distribueret drift foreslås det at FMK udvides med en liste af events pr. CPR-nummer. Et event giver modtageren information omkring en handling der er sket på f.eks. en lægemiddelordination, og hvad resultatet af handlingen er. Et event indeholder tre dele, source, action og result. Source og action beskriver i videst mulig kontekst hvad "sundhedspersonen" har anvendt som udgangspunkt for opdateringen, og hvad resultatet er. Eksempel 1 viser et event for en receptoprettelse.

```

<Event>
  <EventId>131436572900010005</EventId>
  <Source>
    <MedicineCard>
      <VersionId>131436541300020002</VersionId>
      <DrugMedication>
        <DrugMedicationIdentifier>2349166</DrugMedicationIdentifier>
        <VersionId>2011083014591200030002</VersionId>
      </DrugMedication>
    </MedicineCard>
  </Source>
  <Action>
    <Type>CreatePrescriptionMedication</Type>
    <Timestamp>2011-08-30T14:59:12</Timestamp>
  </Action>
  <Result>
    <MedicineCard>
      <VersionId>131436572800010002</VersionId>
      <DrugMedication>
        <DrugMedicationIdentifier>2349166</DrugMedicationIdentifier>
        <VersionId>131436572700010001</VersionId>
        <PrescriptionMedication>
          <PrescriptionMedicationIdentifier>33315129</PrescriptionMedicationIdentifier>
          <Status>Åben</Status>
        </PrescriptionMedication>
      </DrugMedication>
    </MedicineCard>
  </Result>
</Event>

```

Eksempel 1: Event for receptoprettelse, i eksemplet er namespaces udeladt.

I eksemplet herover har ”sundhedspersonen” (dvs. læge m.v.) taget udgangspunkt i et medicinkort for en person. Personens CPR-nummer er ikke indeholdt i event-dokumentet, idet dokumentet herved ikke indeholder nogen form for følsomme oplysninger.

Eventets id er unikt, og er sammensat af et tidsstempel, et procesid og et løbenummer.

Ud fra source-elementet ses hvilken medicinkort-version der er taget udgangspunkt i, og ud fra hvilken lægemiddelordination og lægemiddelordination-version recepten er oprettet (dvs. handlingens kontekst).

Action-elementet viser typen af handling, i dette tilfælde en receptoprettelse (”CreatePrescription-Medication”), og tidspunktet handlingen er registreret i FMK.

Result-elementet indeholder information omkring hvad handlingen har medført af ændringer, igen i videst mulige kontekst. I dette tilfælde er der oprettet en recept med recept-id (”PrescriptionMedicationIdentifier”), recepten har ved oprettelsen status ”Åben”. Ved receptoprettelsen er hverken medicinkortets eller lægemiddelordinationens versionsnummer ændret (jvf. FMKs regler for klinisk versionering af medicinkortet). Recepter har ingen versionsnumre, men har i stedet en status, der indgår i PrescriptionMedication-elementet.

Eventliste

Der udstilles en service, der giver mulighed for at hente en liste af events for et CPR-nummer. Typiske brugsscenarier er:

Opslag på tidsstempel for seneste kendte event: Et klientsystem har tidligere hentet eventlisten. Ud fra tidsstemplet foretager systemet et opslag på eventlisten for at hente events der er opstået herefter.

Opslag på event-id for seneste kendte event: Et klientsystem har tidligere hentet eventlisten. Ud fra event-id foretager systemet et opslag på eventlisten for at hente events der er opstået herefter.

Services

Der udstilles en service til at hente eventlisten. Idet listen af events for et CPR-nummer kan blive meget lang, skal der i requestet angives et interval. Intervallet angives som to af nedenstående, idet både From... og To... skal angives:

- FromEventId
- FromEvent + optionel liste af eventtyper
- FromTimestamp

- ToEventId
- ToEvent + optionel liste af eventtyper
- ToTimestamp

Det er muligt at filtrere listen af events, enten for events der ønskes returneres eller for events der ønskes udeladt. Anvendes elementet IncludeTypes vil kun events med de listede typer af actions blive returneret. For elementet ExcludeTypes vil events med de listede typer af actions blive udeladt, alle andre typer vil blive returneret. Det er ikke muligt at kombinere de to filtre, idet betydningen i så fald ikke er entydig.

Endeligt er det muligt med elementet Limit at begrænse hvor mange events der ønskes returneret. Der vil dog aldrig blive returneret flere events end en "global" limit.

Eksempler på EventRequest

Herunder er vist en række eksempler på event requests. I eksemplerne er namespaces og namespacealiases udeladt.

```
<EventRequest>
  <CPR>111111118</event20120101:CPR>
  <FromTimestamp>2011-08-02T12:34:00</FromTimestamp>
  <ToTimestamp>2011-09-07T12:37:00</ToTimestamp>
</EventRequest>
```

Eksempel 2: Request der kan anvendes hvis der tidligere er hentet et medicinkort, men ikke et event, eller hvis tidligere hentede events ikke gennemses. Der spørges på et tidsinterval fra oprettelsesdatoen for medicinkort-versionen og til "nu". Resultatet vil give information, der kan anvendes til at afgøre om der skal hentes et nyt medicinkort m.v.

```
<EventRequest>
  <CPR>111111118</CPR>
  <FromEventId>131436572900010005</FromEventId>
  <ToTimestamp>2011-09-07T12:37:00</ToTimestamp>
</EventRequest>
```

Eksempel 3: Request der anvendes hvis der tidligere er hentet et event, og der vil undersøges hvad der er sket siden. Der spørges på intervallet fra sidst kendte event til "nu".

I begge ovenstående eksempler vil samtlige events blive returneret, dog kan antallet af returnerede events begrænses til f.eks. de seneste 1000 (se limit længere nede).

Det skal være muligt at filtrere på events. Anvendes elementet IncludeTypes vil kun events med de listede typer af actions blive returneret. For elementet ExcludeTypes vil events med de listede typer af actions blive udeladt, alle andre typer vil blive returneret. Det er ikke muligt at kombinere de to filtre, idet betydningen i så fald ikke er entydig.

```

<EventRequest>
  <CPR>111111118</CPR>
  <FromEventId>131436572900010005</FromEventId>
  <ToTimestamp>2011-09-07T12:37:00</ToTimestamp>
  <IncludeTypes>
    <Type>CreatePrescriptionMedication</Type>
    <Type>UpdatePrescriptionMedication</Type>
    <Type>WithdrawPrescriptionMedication</Type>
    <Type>CreateEffectuation</Type>
  </IncludeTypes>
</EventRequest>

```

Eksempel 4: Request på event-id med en liste af typer der ønskes returneret. I dette tilfælde et opslag på events omkring receptudstedelse, statusskift på recepter, evt. seponeringer og ekspeditioner.

```

<EventRequest >
  <CPR>111111118</CPR>
  <EventId>131436572900010005</EventId>
  <ExcludeTypes>
    <Type>CreatePrescriptionMedication</Type>
    <Type>UpdatePrescriptionMedication</Type>
    <event20120101:Type>WithdrawPrescriptionMedication</Type>
  </ExcludeTypes>
</EventRequest>

```

Eksempel 5: Request på event-id med en liste af typer der ønskes ikke returneret

Med elementet Limit angives hvor mange events der maksimalt ønskes returneret. Der vil dog aldrig blive returneret flere events end en "global" limit på f.eks. 1000 events. I praksis bør den globale limit være tilstrækkelig høj til at resultatet sjældent begrænses i normale tilfælde.

```

<EventRequest>
  <CPR>111111118</CPR>
  <FromTimestamp>2000-01-01T00:00:00</FromTimestamp>
  <ToTimestamp>2999-01-01T00:00:00</ToTimestamp>
  <IncludeTypes><Type>CreateDrugMedication</Type></IncludeTypes>
  <Limit>100</Limit>
</EventRequest>

```

Eksempel 6: Request på alle oprettelser af lægemiddelordinationer (slettede undtaget), dog maksimalt 100.

Eksempel på EventResponse

Eksempel 7 viser et eksempel på et event response. Source og event delen er udeladt i dette eksempel. Elementet MoreAvailable indikerer at ikke alle data er returneret, og at det vil være nødvendigt med yderligere requests for at få alle events returneret. Ikke alle data er returnes hvis Limit elementet er sat lavere end antal events side sidste event eller timestamp, og desuden kan servicen have en "global" limit på f.eks. 1000 events. I praksis bør den globale limit være tilstrækkelig høj til at resultatet sjældent begrænses i normale tilfælde.

```

<EventResponse>
  <Event>
    <EventId>1314369871200010005</EventId>
    <Source>...</Source>
    <Action>
      <Type>CreateDrugMedication</Type>
      <Timestamp>2011-09-02T10:20:32</Timestamp>
    </Action>
    <Result>...</Result>
    ...
  </Event>
  <MoreAvailable/>
</EventResponse>

```

Eksempel 7: Eksempel på response, der indeholder en liste af events. Ikke alle detaljer er vist, og namespace er udeladt. Elementet MoreAvailable indikerer at ikke alle data er returneret, og at det vil være nødvendig med yderligere requests for at få alle events returneret.

Ekstensibilitet

Idet eventmodellen skal kunne håndtere fremtidigt indhold er det nødvendigt at det er muligt at udvide data et event beskriver. F.eks. idet source- eller result-elementet endnu ikke er defineret (eksempelvis et element fra hjemmesygeplejens integration til FMK, der endnu ikke er defineret, eller ved at der sker et event af en ny type (f.eks. at apoteket tilføjer en lægemiddelordination til patientens dosiskort). Ekstensibiliteten opnås ved brug af "any" i skemadefinitionen, se eksemplet herunder.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:event20120101="http://www.dkma.dk/event/xml.schema/2012/01/01"
  xmlns:medicinecard20080601="http://www.dkma.dk/medicinecard/xml.schema/2008/06/01"
  targetNamespace="http://www.dkma.dk/event/xml.schema/2012/01/01"
  elementFormDefault="qualified"
  attributeFormDefault="qualified">
  <xs:import
    namespace="http://www.dkma.dk/medicinecard/xml.schema/2008/06/01"
    schemaLocation="../../../2008/06/01/DKMA_MedicineCardVersionIdentifier.xsd"/>
  <xs:element name="MedicineCard" type="event20120101:MedicineCardType"/>
  <xs:complexType name="MedicineCardType">
    <xs:sequence>
      <xs:element
        name="VersionId"
        type="medicinecard20080601:MedicineCardVersionIdentifierType"/>
      <xs:any minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

Eksempel 8: Skemadefinition for MedicineCard.

Det er ikke muligt at angive indholdet i MedicineCard-elementet mere præcist end at det som minimum skal indeholde et versionsnummer (MedicineCardVersionIdentifier), idet et eller flere optionelle elementer kombineret med "any" gør at skemadefinitionen ikke er deterministisk. Tilsvarende gælder for lægemiddelordinationen.

XML skemadefinitionen vil indeholde de nuværende kendte typer af handler samt source og result dokumenter, men vil også kunne indeholde nye typer, således at en liste af events kan komme til at indeholde eventtyper der ikke er kendt på design tidspunktet. Dvs. at klientsystemet skal kunne håndtere at ignorere ukendte elementer. Nye snitfladeversioner kan tilføje de "ukendte" typer til skemadefinitionen.

Source- og Result-elementer

En event skal indeholde en angivelse af hvad ”sundhedspersonen” har anvendt som udgangspunkt for opdateringen, og hvad resultatet er. Hertil anvendes Source- og Result-elementerne. På nuværende tidspunkt kendes to kilder til events: Ændringer på patientens medicinkort m.v. og ændringer på recepter. Begge typer af elementer vil have nedenstående strukturer:

For handlinger foretaget på FMK, dvs. ud fra en kendt medicinkort-version:

```
Medicinkort (CPR-nummer, medicinkort-version)
Lægemeddelordination (lægemeddelord.-id, lægemeddelord.-version)
Receptordination (receptordination-id, status)
```

For handlinger foretaget på receptserveren:

```
Receptordination (receptordination-id, status)
```

For eksempelvis en opdatering af en lægemeddelordination vil både source og result indeholde en lægemeddelordination:

```
<Event>
  <EventId>1314369871200010005</EventId>
  <Source>
    <MedicineCard>
      <VersionId>131436572700010001</VersionId>
      <DrugMedication>
        <DrugMedicationId>131431112100020002</DrugMedicationId>
        <VersionId>131436597200010001</VersionId>
      </DrugMedication>
    </MedicineCard>
  </Source>
  <Action>
    <Type>UpdateDrugMedication</Type>
    <Timestamp>2001-12-17T09:30:47.0Z</Timestamp>
  </Action>
  <Result>
    <MedicineCard>
      <VersionId>131451211400050003</VersionId>
      <DrugMedication>
        <DrugMedicationId>131431112100020002</DrugMedicationId>
        <VersionId>131451232100050003</VersionId>
      </DrugMedication>
    </MedicineCard>
  </Result>
</Event>
```

Eksempel 9: Eksempel på opdatering af en lægemeddelordination.

Specielt for receptoprettelser uden om FMK vil source elementet ikke være tilstede, idet receptoprettelsen ikke foretages ud fra noget kendt medicinkort eller andet.

```
<Event>
  <EventId>1314369871200010005</EventId>
  <Source/>
  <Action>
    <Type>CreateDrugMedication</Type>
    <Timestamp>2001-12-17T09:30:47.0Z</Timestamp>
  </Action>
  <Result>
    <PrescriptionMedication>
      <PrescriptionMedicationIdentifier>79847239</PrescriptionMedicationIdentifier>
      <PrescriptionMedicationStatus>Åben</PrescriptionMedicationStatus>
    </PrescriptionMedication>
  </Result>
</Event>
```

Eksempel 10: Event for en oprettelse af en løs recept.

I forbindelse med et drift i flere clustre, eller såfremt der ”samtidigt” opdateres ud fra samme medicinkort-version kan der senere opstå en situation hvor der haves to source versioner. Skal to medicinkort-versioner merges vil dette også resultere i et event.

```
<Event>
  <EventId>1314369871200010005</EventId>
  <Source>
    <MedicineCard>
      <VersionId>131436572700010001</VersionId>
    </MedicineCard>
    <MedicineCard>
      <VersionId>131436583300020002</VersionId>
    </MedicineCard>
  </Source>
  <Action>
    <Type>MergeMedicineCard</event20120101:Type>
    <Timestamp>2001-12-17T09:30:47.0Z</event20120101:Timestamp>
  </Action>
  <Result>
    <MedicineCard>
      <VersionId>131436599100010001</VersionId>
    </MedicineCard>
  </Result>
</Event>
```

Eksempel 11: Merge af to medicinkort-versioner, her er der to elementer under Source. Resultatet er en ny medicinkort-version.