

## Het opvragen van E-Dairy berichten .

- **Qlip** levert E-Dairy data alléén indien er een machtiging aanwezig is in JoinData op de **combinatie KVK-UBN**. Iets anders werkt niet.
- **RFC** levert E-Dairy data indien daarvoor een machtiging aanwezig is in JoinData op KVK. Eventuele restricties die aangebracht worden in JoinData (op FTN of UBN) zijn louter “cosmetisch” omdat JoinData in het geval van RFC alleen naar **KVK** kijkt. Je kan dus bij RFC opvragen op KVK,UBN en op FTN.

## Waarom is dit onderscheid er tussen RFC en Qlip.

De zuivelfabrieken van Partico willen het principieel technisch onmogelijk maken dat er data wordt opgevraagd van een UBN , waar een veehouder géén toestemming voor heeft gegeven. RFC is het hier mee eens, maar gaat minder ver in het technisch blokkeren van deze mogelijkheid. Dit leidt tot het volgende:

Bij een **KVK** met **2 UBN's**, waarbij alleen een machtiging is bij UBN A, zal het bij een Qlip veehouder onmogelijk zijn om data op te vragen van UBN B, en dat is technisch gezien wel mogelijk bij RFC.

## Wat houdt dit nu in voor data-opvragers.

**Data-opvragers hoeven** in de wereld van JoinData **niet te weten** of een veehouder bij Qlip of bij RFC zit. Indien een data opvragers zeker weet dat een persoon bij RFC zit, zou men de machtiging op KVK niveau kunnen inschieten, en RFC levert dan ook alle data van dat KVK.

Indien de veehouder zou overstappen naar een andere zuivelfabriek, en het data verzoek zou via JoinData dan naar Qlip worden gestuurd, dan geeft Qlip een foutmelding “ Unknown Location Scheme”: Qlip ondersteunt alleen verzoeken op UBN niveau (accepteert dus niet dat er een machtiging op alleen KVK wordt aangevraagd)

Om dit probleem te voorkomen is het voor data-opvragers het meest robuust, om **altijd** **machtigingsverzoeken in te dienen op de combinatie van KVK en UBN** en dan generiek **altijd op UBN op te vragen**: RFC kan daar namelijk ook mee overweg (maar kijkt dan technisch gezien alleen naar het bovenliggende KVK voor toestemming).

## Machtigingen inschieten

Zie onderstaande beschrijving van JoinData

<https://integration.join-data.net/api/docs#purpose-api>

/api/v{version}/purposes/{purpose\_id}/participations Create participations for the given companies.

Het is hier dus de bedoeling om een **restrictie aan te leggen op UBN niveau**. De werking van deze restrictie werkt bij RFC dus anders dan bij Qlip (zoals hierboven aangegeven).

Indien Data-opvragers dit **generiek op KVK-UBN** doen, dan zal er in principe dus **altijd een antwoord** terug komen vanuit JoinData.

## Hoe ziet een Query er uiteindelijk er uit?

Zie ook <https://integration.join-data.net/api/docs#purpose-api>

Met daarin de beschrijving:

/locations/{location-scheme}/{location-id}/milk-deliveries  
/locations/{location-scheme}/{location-id}/milk-qualities  
/locations/{location-scheme}/{location-id}/milk-invoices  
/locations/{location-scheme}/{location-id}/ubl

De vier berichten kunnen als volgt worden opgevraagd. (url verwijst nu nog naar integration):

<https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-deliveries>  
<https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-qualities>  
<https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-invoices>  
<https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-ubl-invoices>

Met daarachter de diverse mogelijke parameters die in de documentatie staan.

Een voorbeeld hiervan is bijvoorbeeld

[https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-deliveries?  
start-date-time=2021-11-07T00:00:01&end-date-time=2022-02-07T23:59:59](https://integration.join-data.net/api/broker/icar-ade/v1/locations/nl.ubn/12345/milk-deliveries?start-date-time=2021-11-07T00:00:01&end-date-time=2022-02-07T23:59:59)

## Parameters

Hierbij zijn de volgende aanvullingen van belang:

- **modified-since**

Deze parameter wordt **door Qlip niet ondersteund**, omdat dit technisch niet mogelijk is zonder vertraging in de datalevering. Het is dus aan de opvragers zelf, om de data actueel te houden in hun eigen database door slim te synchroniseren (vergelijkbaar met de EDI-Zuivel werkwijze).

In het verlengde hiervan: In sommige berichten is er ook sprake van een “delete indicator” deze is bij Qlip ook altijd “N” wat inhoudt dat de data-opvragers zelf moeten kijken of ze een eerder ingelezen data element moeten verwijderen uit hun eigen administratie of niet. Qlip levert te allen tijde de “as is” situatie van het moment van opvragen. (vergelijkbaar met de EDI-Zuivel werkwijze).

- **start-date-time & end-date-time**

Voor de berichten milk-deliveries en milk-qualities kan er **maximaal een periode van 3 maanden** worden opgevraagd. Voor het **invoice** bericht kan er maximaal voor **1 maand** worden opgevraagd. Deze restricties komen voort uit ons voornemen om te voorkomen dat onze systemen “plat” worden gelegd door zeer uitgebreide vragen.

- **location-scheme**

Zoals al eerder aangegeven data provider Qlip geeft alleen antwoord op data verzoeken voorzien van **nl.ubn**.

## Aandachtspunt Milk-Invoice

- In de **Milk-invoice** (zowel JSON als UBL) heeft **RFC** er voor gekozen om **ook de innames** mee te sturen, omdat deze naar mening van RFC, onderdeel van de melknota zijn. **Qlip doet dit niet** omdat de Particoleden van mening zijn dat hier juist het delivery bericht voor bedoeld is (boer aan het roer). Data-opvragers die dus naast Invoice data, ook delivery data willen hebben, doen er verstandig aan om op beide berichten een machtiging aan te vragen. Hiermee voorkom je dus dat je data mist die je wel nodig hebt, als je antwoord uit de data-bron Qlip krijgt.
- **Facturen die worden aangepast worden opnieuw aangeboden** (in de periode waarin ze zijn aangepast) Stel: in augustus wordt een factuur gecorrigeerd die afkomstig is van maand mei. Dan wordt indien de data van augustus wordt opgevraagd de nieuwe factuur van mei ook meegestuurd. Deze werking is eigenlijk gelijk aan de manier waarop dit gedaan werd in EDI-Zuivel. Het is aan de factuur-opvragers zelf om deze correctie op te merken en een correctie boeking te genereren in hun eigen boekhoud systeem.
- Er zit nog **1 bug** in de invoice: Het gaat hierbij om **naam en adres van de veehouder** en ook het **kvk van de veehouder**. (deze geven we nu onterecht niet door: dit gaan we z.s.m. recht zetten)