

EDI-ZUIVEL NW, syntax, protocol en procesflow

Auteur	:	Specificatie datauitwisseling EDI-Zuivel nw
Datum	:	1 maart 2019
Status	:	versie v19p04

1. ACHTERGROND

Dit document omvat de besluiten met betrekking tot de technische inrichting voor EDI-Zuivel nw. Eerder is in dit kader al opgeleverd het datamodel EDI-Zuivel nw dat als basis is gebruikt voor het ontwerp van de berichtenset EDI-Zuivel nw (Delivery, Quality, Invoice).

2. M.B.T. DE ARCHITECTUUR

- De leveranciers van zuivel-data (Qlip, Elda, FC) richten ieder een eigen webservice in waarop zij de zuivel-data aanbieden (zie de architectuurplaat, bijlage 1).
- Ieder van de leveranciers bepaalt voor zichzelf of deze webservice rechtstreeks door de data-vrager aangesproken kan worden of dat dit via JoinData dient te gebeuren.
 - Voor het definiëren van de webservice calls (requests, responses) mag het geen verschil uitmaken of JoinData het endpoint is of dat de webservice van de data-leverancier (Qlip, Elda, FC) direct het endpoint is; de EDI-Zuivel-nw webservice dient voor beide situaties hetzelfde te zijn en dezelfde calls te ondersteunen.
 - Iedere data-leverancier bepaalt voor zichzelf of zij voor het vastleggen van de autorisaties/machtigingen gebruik maakt van AgriTrust van JoinData of hiervoor een eigen autorisatie/machtigingsomgeving gebruikt.
 - Dit betekent in de praktijk dat JoinData door data-leveranciers gebruikt kan worden als 'transparant platform' maar dat het niet verplicht is om van JoinData gebruik te maken.
- Qlip en Elda zullen voornamelijk de eigen machtigingen in een eigen register blijven beheren. Het EDI-Zuivel nw protocol (de methode) moet zowel kunnen omgaan met machtigingen die in een centraal JoinData register worden beheerd als met machtigen die in aparte registers van data-leveranciers worden beheerd
- **Aandachtspunt:** In het geval er voor EDI-Zuivel nw gebruik gemaakt wordt van verschillende autorisatie/machtigingsomgevingen is het zaak dat deze omgevingen autorisaties/machtigingen onderling kunnen delen; daar moet nog een oplossing voor gevonden worden.

3. M.B.T. SYNTAX

- Voor de datauitwisseling wordt voornamelijk gebruik gemaakt van het REST-Json-protocol.
 - Mocht op basis van voortschrijdend inzicht blijken dat REST-Json geen goede keuze is, dan kunnen de gedefinieerde calls (requests, responses) relatief eenvoudig geconverteerd worden voor een andere syntax-protocol combinatie.
- REST-Json lijkt ook bruikbaar voor M2M bulk-datauitwisseling.
- Hetzelfde REST-Json interface wordt zowel gebruikt voor M2M datauitwisseling als ook voor het aansluiten van bijvoorbeeld smartphone apps.
- De metadata over een data-element (bijvoorbeeld de aanduiding van de codelijst die gebruikt is, of de aanduiding van het type bedrijfsID) wordt doorgaans in Json meegegeven door deze te markeren met een '@' teken.
 - Voor definitieve besluitvorming: Tijdens de pilot implementatie wordt definitief besloten over hoe om te gaan met de metadata (@-constructie gebruiken of niet).
- Herhalen van het FTN (FabrieksTankNummer) als unieke sleutels in de REST-Json-calls op de verschillende nesting niveaus is om praktische redenen toegestaan (maar is wel redundant).

4. M.B.T. PRÉPROCESSING EN BULK UITVRAGEN

- Voor EDI-Zuivel nw wordt de data, conform het class diagram, in aparte datasetjes voor Delivery, Quality en Invoice ontsloten. Delivery en Quality data zullen gedurende een afrekeningsperiode (een maand) meerdere keren beschikbaar komen, Invoice komt aan het eind van de maand beschikbaar en heeft betrekking op meerdere leveringen. Ieder datasetje heeft een timestamp van het moment waarop het is aangemaakt.

- Het protocol moet naast het uitleveren van real time gegenereerde berichten ook het uitleveren van préprocessed berichten ondersteunen. Voor accounts die bij Qlip voor meer dan 100 verschillende veehouders data ophalen worden de berichten préprocessed klaargezet.
 - o Voor een bulkafnemer zet Qlip maar één keer eenzelfde bestand klaar, zodra de data-afnemer dit bestand heeft opgehaald is het verdwenen. En dat blijft voorlopig zo, ook voor EDI-Zuivel-nw.
 - o **Opmerking:** Dit betekent dat de EDI-Zuivel-gegevens via EDI-Zuivel in de huidige situatie maar één keer opvraagbaar zijn (met uitzondering van herstel- en historische data). Dit leent zich minder voor het realtime raadplegen via een app op een mobile device.
- Voor Qlip en FC geldt dat de nieuwe EDI-Zuivel data op gezette tijden (bijv. eens per dag) wordt klaargezet:
 - o Qlip heeft in de huidige situatie de uitgifte per data-afnemer georganiseerd. Dit betekent dat wanneer verschillende afnemers (KLW, accountant, etc.) dezelfde data wensen te ontvangen dezelfde berichten in meerdere 'postbussen' worden klaargezet. De data wordt in de huidige situatie naar alle data-afnemers, ongeacht bulk of enkelvoudig, op vergelijkbare wijze uitgeleverd en dat zal in de nieuwe situatie niet anders zijn.
- De cloud-omgevingen van de BMS-leveranciers vragen op gezette tijden nieuwe data uit bij de data-leveranciers:
 - o BMS-leveranciers (VSM, Agrovision, UniformAgri, e.a.) en grootgebruikers (accountancy, diervoederleveranciers, KLW) worden door de data-aanbieders gezien als trusted parties en hoeven niet steeds apart per veehouder aan te loggen; in de autorisatie/machtigingsomgeving is vastgelegd welke dataset voor een combinatie 'data-opvrager - veehouder' mag worden uitgeleverd.
 - o Voor grootgebruikers lijkt het voldoende dat zij eens per dag de data ophalen.
 - o **Requirement:** De architectuur zo opzetten dat het werkt in het geval een datagebruiker eens per dag data ophaalt (komt overeen met de bestaande situatie) en dat het ook werkt in het geval een individuele veehouder meerdere keren per dag actuele data wenst op te halen.
- EDI-Zuivel nw moet zowel ondersteunen in het uitleveren van préprocessed berichten aan grootgebruikers als in het realtime (on the fly) uitleveren van enkelvoudige berichten.
 - o Het heeft de voorkeur om datasets on-the-fly te genereren, maar vanuit performance overwegingen kan het interessant zijn om de Json-berichten préprocessed klaar te zetten.
 - o Het mag het uitvragen geen verschil uitmaken of berichten préprocessed zijn klaargezet of 'on the fly' gegenereerd worden.
- **Requirement:** Er rekening mee houden dat de data m.b.t. RMO-melkophalen (Delivery), kwaliteitbepaling (Quality) en uitbetaling (Invoice), met verschillende frequentie dient te kunnen worden aangeboden (intervallen verschillen per type data); vanaf het moment dat de data ontstaat zou deze zo snel mogelijk beschikbaar moeten zijn voor de data-gebruikers.

5. M.B.T. DE DIALOOG EN DE JSON CALLS

Voor EDI-Zuivel nw wordt een 'meertrapsraket' voorgesteld, bestaande uit de volgende standaard calls:

- o Call 1: Sign_In: Authenticatie voor het opvragen van het token.
- o Call 2: Capabilities: Rapportage van wat de webservice van de data-aanbieder kan bieden.
- o Call 3: Memberlist: Lijst van voor welke veehouders data kan worden opgevraagd.
- o Call 4: Recent_Data: De 'payload'. In eerste instantie worden 4 typen datasets aangeboden: Delivery, Quality, Invoice en All-in.
- o Call 5: Historic_Data: Speciale call voor het opvragen van historische gegevens.
- Call 1 (Sign_In), de aanmeld-call levert uitsluitend een token terug en is verplicht. Voor het aanmelden kunnen verschillende authenticatietechnieken worden gebruikt. Call 1 moet altijd gedaan worden omdat daarmee het token wordt verkregen dat gebruikt moet worden voor calls 2 t/m 4.
- Call 2 (Capabilities) biedt de mogelijkheid om de variëteit in de wijze waarop Qlip, Elda en RFC de data aanbiedt op een uniforme wijze duidelijk te maken aan data-afnemers.
- Call 5, Historic_Data.
EDI-Zuivel nw kent een aparte call voor het uitvragen van de lopende periode (Recent data) en een aparte call voor het uitvragen van historische data (Historic_Data).
- De 5 calls hoeven niet voor alle use cases verplicht geïmplementeerd te worden: in de situatie waarbij een individuele veehouder vanuit zijn BMS op dagelijkse basis de data ophaalt dan zal deze bijvoorbeeld alleen call 1 (aanmelden) en call 4 doen (uitvragen op FTN). Met de Capabilities-call kan dan worden aangegeven welke calls (Recent_Data, Historic_Data) door de data-aanbieders worden ondersteund.
 - o RFC zal bijvoorbeeld zowel Recent_Data, Historic_Data ondersteunen.
 - o Qlip zal voornamelijk uitsluitend de call voor het uitvragen van de lopende periode (Recent_Data) ondersteunen. Door te spelen met Begindatum en Einddatum kan dan over een specifieke periode worden uitgevraagd.

- Qlip negeert vooralsnog elke datum-variabele in Recent_Data en levert als default-periode de lopende periode (melkgeldregels: vanaf laatste afgesloten periode, stel X). Als periode X wordt afgesloten, wordt in het eerstvolgende bericht die periode X nog eens volledig meegeven in de respons. Plus X+1 zover als gevorderd. Navolgende verzoeken bevatten dan weer default de gegevens van periode X+1. De veehouder kan historische data via Z-Net opvragen. De veehouder kan daar per account, telkens eenmalig, aangeven of de historie (dat is dan ook een voor-gedefinieerde periode) moet worden aangeleverd. Daarmee reguleert Qlip deels de workload omdat ze de aanvragers de mogelijkheid ontnemen om standaard de gegevens van jaren geleden op te vragen.
- **Aandachtspunt:** t.z.t. verkennen of de vijf calls meer generiek gedefinieerd kunnen worden zodat ze ook bruikbaar zijn voor het uitvragen van ander soortige data-bronnen.
- De calls zijn in een aparte bijlage verder uitgewerkt en ter illustratie ingevoerd in Swagger.

6. M.B.T. PAGINERING

- In het geval van bulkuitvraag bepaalt de uitleverende partij of de berichten één voor één moeten worden uitgevraagd of dat na één uitvraag uitgeleverd wordt in een lange string met paginering.
 - **Default:** Default wordt er bij bulkuitvraag in één response met paginering uitgeleverd.

7. M.B.T. PULL VERSUS PUSH BERICHTEN

- Eerst wordt uitsluitend pull-functionaliteit geboden; data-gebruikers pullen bij de data-leveranciers. In een volgende fase kan deze eventueel uitgebreid worden met push-functionaliteit voor het doorgeven van bijvoorbeeld notificaties.

8. M.B.T. DOORGEVEN GEMUTEERDE EN VERVALLEN RECORDS

- Aan het eind van de periode (doorgaans een maand) wordt de complete dataset opnieuw aangeboden en mag de eerder opgehaalde data voor deze periode volledig worden gedelete om te worden vervangen door de nieuwe dataset.

9. M.B.T. SFTP

- Het heeft de voorkeur om de data on-the-fly via REST-Json aan te bieden. Mocht de performance van de REST-Json toepassing onvoldoende zijn voor het uitleveren van bulkdata, dan wordt overwogen om de préprocessed berichten voor bulk-afnemers in een sFTP omgeving aan te bieden. Dit zijn dan dezelfde Json-berichten als die voor uitvraag via REST-Json worden aangeboden.

10. M.B.T. FOUTAFHANDELING

- Voorkeur van uitvragen individuele berichten (records) per boer; dan kan de foutafhandeling ook per individueel bericht worden afgehandeld (bij het uitvragen van een batch aan records wordt het veel complexer om de foutmeldingen goed af te handelen).
- Voor foutmeldingen op transportlaag niveau gebruik maken van standaard REST errors: <https://restfulapi.net/http-status-codes/>
- Validatie van de inhoud van het bericht dient door de leverende partij geborgd te zijn.

11. M.B.T. BEDIENEN ACCOUNTANCY

- De accountancy wenst de melkafrekening in de vorm van een standaard xml-UBL-Invoice bericht te ontvangen. Dat gebeurt nu ook voor EDI-Zuivel oud; op het EDI-Circle platform (JoinData) worden de ADIS-berichten geconverteerd naar xml-UBL-berichten en aangeboden aan de accountancy.
- Bij het maken van de impactanalyse (volgt op de ontwerpfasen waarvan dit het rapport het resultaat is) wordt er ervan uitgegaan dat de EDI-Zuivel nu datasets in Json formaat worden opgeleverd. Dat geldt ook voor de Invoice; deze wordt in eerste instantie nog niet in UBL-xml opgeleverd. T.z.t. wordt een aparte kostenberekening gemaakt voor opleveren van de UBL-xml invoice (zou eventueel ook middels één centrale conversiemodule kunnen worden gegenereerd uit de aangeleverde Json data). Zodra er inzicht

is in de extra kosten voor het genereren van een UBL-xml invoice wordt de accountancy gevraagd naar de mogelijkheden om deze te financieren.

12. M.B.T. NIEUWE DATASETJES

- Er is vraag naar nieuwe datasets die door de zuivelondernemingen moet worden uitgeleverd; data die niet past in de reeds gedefinieerde Delivery, Quality en Invoice berichten. Bijvoorbeeld m.b.t. weidegang, maar zo zullen er nog meer eigenschappen op bedrijfsniveau gevraagd worden.
 - In de informatieanalyse is hier al rekening mee gehouden; zie de structuur van MilkProductionLocation -> MilkProductionLocationCharacteristic -> MilkProductionLocationCharacteristicLine.
 - Voor het rapporteren van deze eigenschappen wordt t.z.t. een aparte dataset (bijv. ProductionUnitCharacteristics) en een nieuwe call gedefinieerd voor het uitvragen van de dataset.

13. M.B.T. MEEGEVEN PDF'S

- Vooralsnog in EDI-Zuivel nw geen rekening houden met het meegeven van pdf-bestanden.

14. M.B.T. SWAGGER EN OPEN API

- Zijn diverse tools voor, Swagger is er één van. Vanuit Swagger kunnen de specificaties van de calls geëxporteerd worden in het standaard Open-API file-formaat en overgedragen worden naar een andere tool of een andere Swagger instance. Dus het is geen probleem om een eerste aanzet te maken in de VSM-Swagger-omgeving van Johan Vloet.

Bijlage 1: Architectuurplaat EDI-Zuivel nw

