

## Issuelog UBL-invoice eDAIRY

---

**Betreft** : Issuelog UBL-invoice eDAIRY  
**Auteur** : Conny Graumans, AgroConnect  
**Datum** : 19jul22  
**Status** : concept

---

### 1. Simpler Invoice, PEPPOL

25jan21, Van: Hoogenberg, J. (Jan) [mailto:Jan.Hoogenberg@frieslandcampina.com]  
Aan: Conny Graumans; 'Hendricus Wind'; 'Rene Rankenberg' CC: Smeenk, G.J. (Gerrit)  
Onderwerp: Nieuwe versie UBL factuur

Allen, Bijgaand de nieuwe versie van de UBL, zoals deze op dit moment op onze integratieomgeving staat. Vanwege wat systeemtechnische problemen is de doorlooptijd wat langer geweest, gelukkig draaien onze systemen op dit moment alweer een tijd stabiel. In de nieuwe versie is het probleem mbt de leverancier en de klant opgelost. Ik hoor graag of de huidige versie correct is en goed verwerkt kan worden.

25jan21, Van: Hendricus Wind [mailto:hendricus.wind@unifiedpost.com]  
Aan: 'Hoogenberg, J. (Jan)'; 'Conny Graumans'; 'Rene Rankenberg' CC: 'Smeenk, G.J. (Gerrit)'  
Onderwerp: RE: Nieuwe versie UBL factuur  
Wij lopen tegen 2 zaken aan:

- Wij kunnen de XML niet valideren omdat we de CustomizationID niet herkennen.
- De gebruikte schemeID's zijn niet valide, bv schemeID="nl.plnt" schemeID="nl.ubn".

25jan21, Van: Hoogenberg, J. (Jan) [mailto:Jan.Hoogenberg@frieslandcampina.com] Aan: Hendricus Wind; 'Conny Graumans' CC: Smeenk, G.J. (Gerrit); 'Rene Rankenberg'  
Hendricus, Dank voor de snelle reactie, wellicht goed om een call te doen met Conny om te kijken wat er fout gaat. CustomizationId en SchemeId komen uit de specificatie. Zal ik een invite uitdoen om dit te beslechten?

25jan21, Van: Hendricus Wind [mailto:hendricus.wind@unifiedpost.com] Aan: 'Hoogenberg, J. (Jan)'; 'Conny Graumans' CC: 'Smeenk, G.J. (Gerrit)'; 'Rene Rankenberg'  
Halo Jan, Lijkt mij goed dat Conny eerste een blik werpt, wij volgen de SI UBL waarden. Zoals die te vinden zijn op [https://stpe.semantic-treehouse.nl/#/TreeView/Property\\_1468923843\\_785779](https://stpe.semantic-treehouse.nl/#/TreeView/Property_1468923843_785779)

25jan21, Van: Conny Graumans [mailto:c.graumans@chello.nl] Aan: 'Hendricus Wind'; 'Hoogenberg, J. (Jan)' CC: 'Smeenk, G.J. (Gerrit)'; 'Rene Rankenberg'  
Hendricus, Het bericht is valide met de UBL v2.1. SI UBL is daar een inperking op; binnen SI UBL lukt het ook niet om bijvoorbeeld de samenstelling van de melk mee te geven. Dat is dezelfde discussie die we met de invoice voor Van Iperen hebben. Ik ben van mening dat UnifiedPost ook de uitwisseling van standard UBL v2.1 berichten zou moeten ondersteunen.

20jan21, Van Iperen discussie: Van: Jelte Jansen [mailto:jelte@headon.nl] Aan: Veer, R.J.J. (Robin) de CC: Conny Graumans; 'Hendricus Wind'

Onderwerp: Re: FW: Harmonisatie UBL v1.2 en SI-UBL v2.0

Robin, in het bericht aan ons vragen ze het wel wat het pad is om dit ondersteund te krijgen in SI-UBL 2.0, dus ik denk dat dit juist iets is om binnen het STPE te bespreken. Dat zou oorspronkelijk dan ook mijn antwoord zijn geweest, van SI-UBL 1.2 gaat geen updates meer krijgen.

Dus ik ga maar even uit van SI-UBL 2.0. Heel kort door de bocht is het pad dan denk ik als volgt:

- De use-cases vaststellen binnen het STPE.
- Kijken of de oplossingsrichting toepasbaar is binnen een CIUS van de Europese Norm 16931.
- Zo niet, dan besluiten of de use-cases sterk genoeg zijn voor een extensie.
- De semantische regels voor NLCIUS of de extensie vaststellen met de werkgroep STPE
- Deze regels implementeren in SI-UBL 2.X.

25jan21, Van: Hendricus Wind [mailto:hendricus.wind@unifiedpost.com] Aan: 'Conny Graumans'; 'Hoogenberg, J. (Jan)'  
CC: 'Smeenk, G.J. (Gerrit)'; 'Rene Rankenberg'

Hallo Conny, Begrijp je argument, maar zou ook verder kijken dan alleen Unifiedpost. Sluit je bericht wel aan bij SI dan is het bereik waar je kunt afleveren veel groter dan alleen de partijen die dit specifieke formaat hebben geïmplementeerd

27jan21, Van: Conny Graumans [mailto:c.graumans@chello.nl] Aan: 'Hendricus Wind'; 'Hoogenberg, J. (Jan)'; 'Rene Rankenberg' CC: 'Smeenk, G.J. (Gerrit)'

Onderwerp: RE: Nieuwe versie UBL factuur

Jan, René, Hendricus, Gisteren een en ander doorgesproken met Jan Hoogenberg (RFC), Robin de Veer (TNO) en René Rankenberg (Countus).

Bij de ontwikkeling van de UBL-melkgeldafrekening is van begin af aan besloten om deze conform UBL-invoice v2.1 op te zetten; inclusief de mogelijkheid om middels AdditionalItemProperty de samenstelling van de melk mee te geven.

```
<cac:Item>
  <cac:AdditionalItemProperty>
    <cbc:Name>Vet</cbc:Name>
    <cbc:NameCode listAgencyID="ZZZ" listID="CL761">021</cbc:NameCode>
    <cbc:ValueQuantity unitCode="P1" unitCodeListID="UN/ECE 6411">3.93</cbc:ValueQuantity>
  </cac:AdditionalItemProperty>
</cac:Item>
<cac:Item>
```

Het AdditionalItemProperty en ook de type identifiers die we gebruiken (n.ubn, nl.kvk etc.) passen niet in de NLCIUS SI-UBL v2.0 versie. Ik heb dit doorgesproken met Robin de Veer (TNO, NL-CIUS).

Robin heeft toegezegd om de change om dit wel mogelijk te maken in SI-UBL v2.0 voor te leggen aan de onderhoudscommissie van voornoemde standaard (de STPE-werkgroep). De verwachting is dat er op korte termijn geen aanpassing komt op de SI UBL 2.0.

Het alternatief is om de technische gegevens weer uit de UBL-invoice te knippen en middels een DespatchAdvice-bericht uit te wisselen; of om de extra gegevens in een extensie op SI-UBL v2.0 te definiëren.

Ik heb met René overlegd hoe belangrijk het voor de accountancy is om de invoice berichten conform SI-UBL v2.0 te ontvangen. Voor Countus is dat niet belangrijk; zij kunnen prima uit de voeten met het UBL-invoice v2.1 bericht zoals dat nu is gedefinieerd.

Vraag aan Hendricus is of er andere klanten dan de accountancy zijn, die van de UBL v2.1 eDAIRY invoice niet zouden willen / kunnen ontvangen en verwerken?

Mochten andere partijen ook met UBL-invoice v2.1 berichten uit de voeten kunnen, dan zou ik willen voorstellen om het te laten zoals het nu is (UBL-invoice v2.1). De vraag blijft dan wel of onze eDAIRY UBL-facturen via een platform als Z-factuur / Z-dossier verwerkt kunnen worden?

29jan21, Van: Hendricus Wind [mailto:hendricus.wind@unifiedpost.com] Aan: 'Conny Graumans'; 'Hoogenberg, J. (Jan)'; 'Rene Rankenberg' CC: 'Smeenk, G.J. (Gerrit)'

Lastig dilemma. Alle landelijke pakketten volgen de NLCISU / NPA lijn en ondersteunen de SI / NLCIUS / en extensie varianten, waarbij ze valideren tegen de CustomizationID. Voldoet de factuur niet hieraan zullen ze veelal de factuur niet kunnen inlezen. Voor ons platform gaat een eDAIRY versie betekenen dat wij een extra versie moeten implementeren, onderhouden en daarop ook specifieke conversies moeten maken en onderhouden. UBL eDAIRY - EAV, UBL eDAIRY - UBL 2.1 en UBL eDAIRY - 2.0 om alle klanten op ons platform te kunnen bedienen. Zo hebben wij bv diverse klanten en ook accountantskantoren die werken met Snelstart maar ook die werken met Agro pakketten die alleen EAV/Agroconnect ontvangen.

Bij veel van deze pakketten en ook bij ons zal het een business case afweging zijn of een dergelijke versie geïmplementeerd gaat worden. Hoop dat dit iets meer inzicht geeft.

29jan21, Van: Conny Graumans [mailto:c.graumans@chello.nl] Aan: 'Hendricus Wind'; 'Hoogenberg, J. (Jan)'; 'Rene Rankenberg' CC: 'Smeenk, G.J. (Gerrit)'

Mmm, moeilijk. Laten we het eerst maar even houden bij hoe het nu is geïmplementeerd, UBL v2.1 dus.

Status 29jan21: In behandeling.

## 2. AdditionalItemProperty, lege segmenten weglaten

25jan21, From: Conny Graumans <[c.graumans@chello.nl](mailto:c.graumans@chello.nl)> To: Hoogenberg, J. (Jan) <[Jan.Hoogenberg@frieslandcampina.com](mailto:Jan.Hoogenberg@frieslandcampina.com)>  
Cc: Smeenk, G.J. (Gerrit) <[Gerrit.Smeenk@frieslandcampina.com](mailto:Gerrit.Smeenk@frieslandcampina.com)>; 'Hendricus Wind' <[hendricus.wind@unifiedpost.com](mailto:hendricus.wind@unifiedpost.com)>; 'Rene Rankenberg' <[r.rankenberg@countus.nl](mailto:r.rankenberg@countus.nl)>  
Subject: RE: Nieuwe versie UBL factuur  
I used [UBL-Invoice-2.1\\_eDAIRY\\_12jan21.xsd](#) to validate the eDAIRY UBL test message.  
I noticed that:

```
<cac:Delivery>
  <cbc:ActualDeliveryDate>2018-07-01</cbc:ActualDeliveryDate>
  <cac:Shipment>
    <cbc:ID>0028403774</cbc:ID>
    <cac:GoodsItem>
      <cac:Item>
        <cac:AdditionalItemProperty>
          <cbc:Name/>
          <cbc:NameCode listAgencyID="ZZZ" listID="CL761"/>
          <cbc:ValueQuantity unitCode="" unitCodeListID="UN/ECE 6411"/>
        </cac:AdditionalItemProperty>
      </cac:Item>
    </cac:GoodsItem>
  </cac:Shipment>
</cac:Delivery>
```

Should be:

- This instance of an **AdditionalItemProperty** does not refer to a specific property. It should be something like:

```
<cac:Item>
  <cac:AdditionalItemProperty>
    <cbc:Name>Vet</cbc:Name>
    <cbc:NameCode listAgencyID="ZZZ" listID="CL761">021</cbc:NameCode>
    <cbc:ValueQuantity unitCode="P1" unitCodeListID="UN/ECE 6411">3.93</cbc:ValueQuantity>
  </cac:AdditionalItemProperty>
</cac:Item>
<cac:Item>
```

- So I removed it from the enclosed version of the test message.

The same goes for:

```
<cac:Shipment>
  <cbc:ID>0028430699</cbc:ID>
  <cac:GoodsItem>
    <cac:Item>
      <cac:AdditionalItemProperty>
        <cbc:Name/>
        <cbc:NameCode listAgencyID="ZZZ" listID="CL761"/>
        <cbc:ValueQuantity unitCode="" unitCodeListID="UN/ECE 6411"/>
      </cac:AdditionalItemProperty>
    </cac:Item>
  </cac:GoodsItem>
</cac:Shipment>
```

- I also removed this one from the enclosed version of the test message.

Now the modified (enclosed) test message is valid.

19feb21, Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: 'Conny Graumans' <c.graumans@chello.nl>  
Onderwerp: FW: Nieuwe versie UBL factuur  
N.a.v. telefoontje van Jan over de status van b openstaande issue. Afgesproken dat Jan en ik dit ma 22feb21 doorspreken; Jan belt mij.

Status 22feb21: In behandeling.

### 3. Testbericht Abab

14okt21, Van: c.graumans@chello.nl [mailto:c.graumans@chello.nl]  
Aan: Affourtit, Frido <frido.affourtit@abab.nl>  
Onderwerp: RE: E-dairy UBL  
Bijgevoegde twee testvoorbeelden van de eDAIRY ubl-invoice:  
- 210112.UBL\_v2.1\_Dairy\_Invoice\_icl\_deliveries.xml (met fake waarden)  
- 210125.UBL invoice test 25012021\_cg.xml (met realistische waarden)  
Zou je die eens door de mangel willen halen en kijken of jullie die kunnen verwerken?  
De invoerinstructie voor de eDAIRY ubl-invoice zit ook in het zip-bestand.  
Bij voorbaat dank en met vriendelijke groet.

14okt21, Van: Affourtit, Frido <frido.affourtit@abab.nl> Aan: 'c.graumans@chello.nl' <c.graumans@chello.nl>  
Dag Conny, Zie onderstaande reactie van Ton.

Van: Ton Langens (Abab) <ton.langens@abab.nl>  
Goedemiddag Frido, Dit bestand heb ik zoals we UBL berichten verwerken kunnen verwerken.  
Echter heb mijn twijfels over het feit dat het nieuwe formaat is. Copie factuur zat hier niet bij omdat de link naar de factuur een lokaal pad was op de pc van Rene Rankenberg volgens mij ook klant gegevens van Rene  
Het bestand "210125 ubl invoice test 25012021\_cg.xml heb ik ook ingelezen hierbij kwam wel de debiteur campina maar geen boekingsregels. Graag zou ik van enkele facturen en verschillende klanten het nieuwe bericht rechtstreeks van de campina willen ontvangen om een goede test te kunnen uitvoeren.

18okt21, Van: c.graumans@chello.nl <c.graumans@chello.nl> Aan: 'Affourtit, Frido' <frido.affourtit@abab.nl>; 'ton.langens@abab.nl' <ton.langens@abab.nl>  
Ton, Frido, Bedankt voor de feedback. Ik vraag Jan Hoogenberg van RFC of hij een aantal realistische facturen kan aanleveren. Hebben jullie voor Jan en mij een paar Abab-klanten waarvoor je de ubl-invoices zou willen ophalen dan kan ik Jan vragen om voor die klanten een paar realistische facturen aan te leveren?

18okt21: In behandeling.

#### 4. Visma en credit factuur

18okt21, From: Rene Rankenberg <[r.rankenberg@countus.nl](mailto:r.rankenberg@countus.nl)> To: Smeenk, G.J. (Gerrit)

<[Gerrit.Smeenk@frieslandcampina.com](mailto:Gerrit.Smeenk@frieslandcampina.com)>

Subject: Overleg AgroConnect WG Zuivel

Hallo Gerrit, a.s donderdagochtend kan ik tot uiterlijk 10:30 u aanhaken bij het overleg, daarna zal ik afhaken. Voor het geval dat deel ik hier alvast input van mijn kant.

##### Aansluiting op Visma

Inmiddels blijkt Visma voor haar eigen applicatie (Visma.net) een test eDairy-UBLfactuur in te richten waarbij VAA is intermediary is/wordt ingeschakeld om via restJson het bericht op te halen bij JoinData en uit te leveren aan Visma.net. Wij (Countus) worden hierbij betrokken. Vervolgens zal hun dochteronderneming AccountView ook meegenomen worden om hetzelfde bericht in te kunnen lezen.

##### DelInd versus credit factuur

Vervolgens een vraag/opmerking over het volgende in de agenda.

---

Ter kennisgeving:

- **Issue 8**. Verwijderde regels (**bijlage 4**). <ter kennisgeving>

Om aan te geven dat regels verwijderd zijn is op verschillende niveaus in de eDAIRY-belichten het boolean type ...DelInd toegevoegd; voor het milkDelivery segment is dit milkDeliveryDelInd, voor het invoice segment is dit invoiceDelInd.

---

Als een factuur wordt aangepast dan dient er niet alleen een gewijzigd bericht te worden klaargezet maar een credit-factuur en een nieuwe factuur (beide met notadatum van uitgifte van nieuwe factuur, te worden aangeleverd. In het verleden hebben wij (aanmerkelijke) problemen ervaren als dit niet op éénduidige en juiste manier werd uitgevoerd. Bijvoorbeeld toen de btw op de heffing voor fosfaatreductieplan werd gecorrigeerd. Wij (accountants) kunnen er van uitgaan dat de nieuwe systematiek dit op een correcte wijze uitvoert?

18okt21, Conny aan René Rankenberg

Vraag aan René, Haalt VAA bij RFC dan de ubl-invoice op of haalt VAA bij RFC de Json eDAIRY invoice op en maakt VAA daar dan een ubl-invoice van (dat zou wel een beetje vreemd zijn(?))?

18okt21: René aan Conny:

Ik heb tot nu toe alle info over betrokkenheid VAA (sinds afgelopen vrijdagmiddag) via consultant successievelijk projectleider Visma en nog geen info van VAA zelf. Gaat mij meer om het signaal dat er vanuit zijde accountants en specifiek de ondersteunende applicaties, er beweging komt om aan te sluiten op JoinData via de voor accountants "nieuwe route" (tot nu toe ontsloten wij via de bestaande EDI-Circle-route).

18okt21: In behandeling.

## 5. Multiple ubl-invoices in Json call response, array

4dec21, From: Sehgal, G. (Gunjan) <Gunjan.Sehgal@frieslandcampina.com> To: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>

Subject: Regarding UBL!

Hi Jan, I made the correction and there are few things to be highlighted:

- 1) I have used below URL:  
<https://a4o.frieslandcampina.com:1443/RESTAdapter/EDairy/locations/nl.ftn/001200012/UBLInvoices?start-date-time=2021-02-01T00:00:00Z&end-date-time=2021-04-30T23:59:59Z>
- 2) For this PFA the output just before it goes to JDH.
- 3) It has 2 invoices, but in the output you see just one as in the structure they provided 1..1 occurrence for invoice:

Structure	Occurrences	Type
ActualDeliveryDate	0..1	xsd:string
Shipment	0..1	xsd:string
ID	0..1	xsd:string
GrossWeightMeasure	0..1	xsd:string

Structure	Occurrences	Type
Invoice	1..1	p7:InvoiceType
UBLVersionID	0..1	p8:UBLVersionIDType
CustomizationID	0..1	p8:CustomizationIDType
ProfileID	0..1	p8:ProfileIDType

Which ideally should not be the case as in total 2 invoice should be passed to JDH in this case.

- 4) Please check the attached payload, if all the values looks ok to you, we need to connect with Conny to modify the structure again, then shall modify the final mapping.

9dec21: Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: c.graumanns@chello.nl  
Onderwerp: RE: Regarding UBL!

Conny, Mbt de UBL factuur, ik liep er tegenaan dat bij meerdere facturen in dezelfde periode dit fout ging. Nu heb ik Gunjan gevraagd dit aan te passen alleen geeft zij aan dat dit niet conform de spec is. Wellicht even overleggen hoe hier mee verder te gaan.

10dec21: Analyse door Conny Graumanns an Jan Hoogenberg

Zie niet waarom de UBL-invoice specs hierop aangepast zouden moeten worden? Iedere invoice heeft zijn eigen invoice-nummer en is uniek. Als response op een call om de invoices voor een periode aan te leveren kunnen dus meerdere invoices worden geretourneerd. Als dit niet goed werkt dan zit het probleem wellicht in de response call waarin is vastgelegd dat maar één invoice mag worden geretourneerd maar dat ligt buiten de specs van de UBL-invoice, dat is vastgelegd in de specificatie van de REST Json call. Of mis ik iets?

23dec21: From: Sehgal, G. (Gunjan) <Gunjan.Sehgal@frieslandcampina.com>

Sent: donderdag 23 december 2021 18:52 To: Hoogenberg, J. (Jan)

<Jan.Hoogenberg@frieslandcampina.com> Subject: UBL Invoice Issue with mutiple invoice!

Hello Jan, I checked and this has to be fixed at Conny's side:

Name	Category	Type	Occurrence	Default	Details	Description
Invoice	Message	InvoiceType	1			This element MUST b
p1:UBLVersionID	Element	p1:UBLVersionIDType	1			
p1:CustomizationID	Element	p1:CustomizationIDType	1			
p1:ProfileID	Element	p1:ProfileIDType	1			
p1:ID	Element	p1:IDType	1			
p1:IssueDate	Element	p1:IssueDateType	1			
p1:IssueTime	Element	p1:IssueTimeType	1			

This xsd was provided to us, which has occurrence of Invoice at root level 1. Hence always 1 invoice is passed to JDH. If this occurrence can be changed to 0. Unbounded. Then multiple invoices can be passed to third party as well.

Attaching the payload just before it goes to JDH. It has two invoices for 0092529397 and 0092540200



While the one passing to JDH is only for : 0092529397 (PFA the payload to JDH). Hope this helps.  
I am available for a discussion with Conny to explain him better. Kindly do let me know.

24dec21:

Good morning, Jan, Gunjan, Gijs,

Concerning the multiple eDAIRY xml ubl-invoices in the response to the JoinData REST Json request.

**@Gunjan:** The xsd (UBL-Invoice-2.1\_eDAIRY\_12jan21.xsd) you refer to, indeed specifies the structure of one xml ubl-invoice. This xsd is based on the global standard ubl-invoice xsd that also provides the specification of a single ubl-invoice. That is what it is and cannot be altered.

The way you incorporate multiple invoices in one response to a requests should be specified in the API protocol that is implemented; the way this array of ubl-invoices is incorporated in de response, depends on the API protocol used and will differ for SOAP, FTPs, REST Json.

So, there is a clear distinction between the specification of the single xml ubl-invoice (UBL-Invoice-2.1\_eDAIRY\_12jan21.xsd specification by AgroConnect) and the specifications of the protocol used to exchange these invoices (**there specifications should be made available by JoinData**).

Since in the use case with FrieslandCampina, the API calls are routed through the JoinData platform, I suppose a REST Json API protocol is used to exchange the xml ubl-invoices.

**@Gijs:** It is up to Join Data to deliver the specifications of how the multiple xml ubl-invoices should be incorporated in the JoinData REST Jason post or get calls.

In case RFC needs to post the xml ubl-invoices to the JoinData platform, the multiple invoices will probably need to be reported as an Json array, using square brackets, etc. The same goes for a response to a REST Json get call.

**@Gijs:** how would you like to see multiple xml ubl-invoices reported in the post or get REST Json call via JoinData; could you provide a specification and an example?

Please let me know if I'm wrong about this and if you don't agree.

Kind regards, Conny Graumans, AgroConnect, +31 6 43038612

24dec21: Van: Sehgal, G. (Gunjan) <Gunjan.Sehgal@frieslandcampina.com>

Aan: c.graumans@chello.nl; Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>

CC: Gijs Nijman (JoinData) <Gijs@join-data.nl>

Onderwerp: RE: UBL Invoice Issue with mutiple invoice!

Hello Conny, So if I understand the mail correctly the xsd you provided is only for single ubl invoice, for multiple we need specification from JDH as what they are expecting and in json or xml format?

Ok then I wait for response from Gijs. Thanks so far and Merry Christmas and great 2022

26dec21: Van: c.graumans@chello.nl <c.graumans@chello.nl>

Aan: 'Sehgal, G. (Gunjan)' <Gunjan.Sehgal@frieslandcampina.com>; 'Hoogenberg, J. (Jan)'

<Jan.Hoogenberg@frieslandcampina.com>

Indeed, that is how I see it Gunjan.

6jan22: Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>

Aan: Jan Herrygers <JHerrygers@vaa.com>; c.graumans@chello.nl

CC: 'Johan Fiddelaar' <johan@join-data.nl>; 'Renze van de Kuilen' <Renze@join-data.nl>; Jack Scheepens <JScheepens@vaa.com>; 'Gijs Nijman' <Gijs@join-data.nl>

Voor het aanleveren van meerdere facturen in één bericht wacht ik even op JoinData mbt het format dat we moeten aanleveren, voor jou informatie wij leveren een XML aan bij JoinData en zij zetten er een json schil omheen. Ik denk dat de json schil straks bij ons er om heen moet en er een array met facturen in de json komt. Maar ik wacht hier nog op input van JoinData, zij zullen dit met prio oppakken.

3feb22: uit de eDAIRY onderhoudscommissie:

- **Status:** Heeft de aandacht van JoinData (**Renze**) en RFC (**Jan**). Deze issue opnemen in het eerdergenoemde change register voor het monitoren van de status.

21feb22: Uit telefonisch overleg met Jan Hoogenberg n.a.v. vraag Floris Ruitkamp:

Ik heb daarnet Jan (Hoogenberg) gesproken over het uitleveren van de ubl-invoice. RFC levert voor de melkgeldafrekening de ubl-invoice uit. Wanneer deze via JoinData wordt uitgevraagd dan worden de ubl-invoices verpakt in een Json schil. Het probleem was dat in één Json response op een uitvraag eigenlijk 1 of meerdere ubl-invoices in de Json-schil meegeleverd moeten kunnen worden en dat er in de huidige implementatie maar steeds 1 ubl-bericht in de Json schil geleverd kon worden. Om meerdere ubl-berichten binnen 1 Json schil te kunnen uitleveren moet in de schil een array worden ingevoegd; Jan en JoinData zijn hard aan het werk om deze aanpassing door te voeren.

Status 21feb22: In behandeling.



## 6. Issue-date, due-date, VAA

24dec21, Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: Gijs Nijman <Gijs@join-data.nl> CC: Johan Fiddelaar <johan@join-data.nl>; Renze van de Kuilen <Renze@join-data.nl>; c.graumans@chello.nl

Onderwerp: RE: eDairy UBL en overige endpoints. Gijs, Zie onder. Jan [red.: antwoorden van Jan Hoogenberg in blauw weergegeven]

23dec21: From: Gijs Nijman <Gijs@join-data.nl> To: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Cc: Johan Fiddelaar <johan@join-data.nl>; Renze van de Kuilen <Renze@join-data.nl> Subject: FW: eDairy UBL en overige endpoints  
Hoi Jan, Ivm het verlof van Renze, zet ik deze punten even naar je door. Jan Herrygers [red.: Van Aaken Data Works] heeft een aantal bevindingen en vragen welke jullie endpoint(s) raken.

- Verder merk ik dat er bij langere periodes soms gateway time-outs optreden. Is het mogelijk om enkel nieuwe en/of gewijzigde data op te vragen voor een bepaalde periode?  
<24dec21,jhoo: Op dit moment niet, we hebben hier nog geen "modified since" voorzien. In principe wordt een factuur nooit gewijzigd. Wellicht dat we deze dan "creation date" oid moeten noemen.  
mbt time outs moeten we even kijken, bij de andere berichten dwingen we een maximale periode af van 3 maanden, om te voorkomen dat men in één keer 2 jaar aan data gaat halen. De hoeveelheid data bij de UBL is aanmerkelijk groter dan bij de andere berichten.>  
<27dec21,jher: Persoonlijk heb ik geen probleem om de parameter "modified since" te noemen. Bij de andere stromen komen nieuwe berichten die (nog) niet zijn gewijzigd immers ook door zolang ze maar zijn gecreëerd na de opgegeven peildatum. Dan heb je het voordeel dat de naam van de parameter herkenbaar is over de stromen heen. Los van de uiteindelijke naam, zou het voor ons zeer wenselijk zijn dat de parameter er komt.>
- Ik merk dat er bij elke aanroep maximaal 1 UBL terugkomt. Als ik opvraag van 01-01-2021 tot 31-12-2021, dan krijg ik 1 "Afrekening melkgeld januari 2021". Als ik bij dezelfde klant opvraag van 01-02-2021 tot 31-12-2021, dan krijg ik 1 "Afrekening melkgeld februari 2021". Is dat hoe het hoort te werken? Zo ja: hoe moeten we dan omgaan met nabetalingsfacturen, want dan zijn er 2 facturen in 1 maand?  
<24dec21,jhoo: Hier is over gesproken met Conny, als het goed is heb je hier een mail van gekregen, we moeten even afstemmen hoe we dit in een correct format gaan zetten.>  
<27dec21,cg: @Jan Herrygers (VAA): op één request moeten 0, 1 of meerdere ubl-invoices teruggekeerd kunnen worden. Zie bijgevoegde correspondentie m.b.t. het opleveren van meerdere ubl-invoices>  
<27dec21,jher: Ik zie ook dat de UBL terug komt in een JSON array, dus er is technisch gezien plaats voor een reeks van invoices. De aanleiding van mijn vraag is dat er altijd maar 1 invoice in de array zit, ook al zijn de parameters ruim genoeg gekozen om meer invoices terug te krijgen.>
- Ik zie dat de inhoud van het EmbeddedDocumentBinaryObject niet goed doorkomt (er zitten extra karakters "\n" tussen, als ik alle voorkomens van "\" gevolgd door "n" weghaal, dan werkt het wel.  
<24dec21,jhoo: Moet ik naar kijken, wij hebben de documenten zo opgeslagen en dit is voor zo ver ik weet een standaard manier ( base64 encoded ).>  
<27dec21,jher: Dit heb ik zojuist getackled. De \n komt voort uit het encoden van de XML als een JSON string. Dit zou dus opgelost moeten zijn als we de XML automatisch gaan uitlezen uit de JSON envelop. Nu vraag ik de documenten nog op met de hand voor de eerste controles.>
- Ook zie ik dat de IssueDate en DueDate anders gevuld zijn dan ik had verwacht.  
<24dec21,jhoo: Graag iets specifieker in wat je bedoeld.>  
<27dec21,jher: Bijvoorbeeld factuur 0092518401. Deze heeft volgens de UBL IssueDate (factuurdatum) 2021-03-31 en DueDate (betaaldatum) 2021-04-07. Zie ook <https://www.agroconnect.nl/LinkClick.aspx?fileticket=AVJmJnXfy6c%3d&portalid=10>:

Invoice. Issue Date. Date	Occurrence	1 .. 1
	Type	Date. Type
	Definition	The date, assigned by the sender, on which this document was issued.
	Remark	Dairy: 2018-12-06. ADIS: 201134 Factuurdatum.
Invoice. Due Date. Date	Occurrence	0 .. 1
	Type	Date. Type
	Definition	The date on which Invoice is due.
	Remark	Dairy: The ultimate date the invoice has to be paid. In Dutch: Betaaldatum.

Volgens de embedded PDF is de factuurdatum 07 april 2021 en de betaaldatum 12 april 2021. Ik had verwacht dat de UBL de datums geeft zoals ze in de PDF staan. Zou jij hier naar kunnen kijken en antwoord geven / een aanpassing doen indien nodig? >

27dec21: Van: Jan Herrygers <JHerrygers@vaa.com> Aan: 'Hoogenberg, J. (Jan)' <Jan.Hoogenberg@frieslandcampina.com>; 'Gijs Nijman' <Gijs@join-data.nl>  
CC: 'Johan Fiddelaar' <johan@join-data.nl>; 'Renze van de Kuilen' <Renze@join-data.nl>; Jack Scheepens <JScheepens@vaa.com>; c.graumans@chello.nl  
Gijs, Jan, Mijn antwoorden hieronder [red. hierboven] in het groen.

27dec21, Conny aan Jan Herrygers:

M.b.t. .... Volgens de embedded PDF is de factuurdatum 07 april 2021 en de betaaldatum 12 april 2021. Ik had verwacht dat de UBL de datums geeft zoals ze in de PDF staan. ....  
Hoe staan ze nu in de UBL en waar wijkt dat af van de pdf volgens jou?

27dec21: From: Jan Herrygers <JHerrygers@vaa.com> To: c.graumans@chello.nl; Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>; 'Gijs Nijman' <Gijs@join-data.nl>  
Cc: 'Johan Fiddelaar' <johan@join-data.nl>; 'Renze van de Kuilen' <Renze@join-data.nl>; Jack Scheepens <JScheepens@vaa.com>  
Zoals ik al schreef: heeft volgens de UBL IssueDate (factuurdatum) 2021-03-31 en DueDate (betaaldatum) 2021-04-07.

Letterlijk:

<cbc:IssueDate>2021-03-31</cbc:IssueDate>

<cbc:DueDate>2021-04-07</cbc:DueDate>

Mijn vermoeden is dat de verkeerde datum (31 maart i.p.v. 7 april) wordt ingevuld in de UBL.

Ter volledigheid: ik verwacht dat in de UBL komt te staan, op basis van de PDF, en op basis van mijn ervaring met EDI-Zuivel berichten van FrieslandCampina:

<cbc:IssueDate>2021-04-07</cbc:IssueDate>

<cbc:DueDate>2021-04-12</cbc:DueDate>

6jan22: Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>

Aan: Jan Herrygers <JHerrygers@vaa.com>; c.graumans@chello.nl

CC: 'Johan Fiddelaar' <johan@join-data.nl>; 'Renze van de Kuilen' <Renze@join-data.nl>; Jack Scheepens <JScheepens@vaa.com>; 'Gijs Nijman' <Gijs@join-data.nl>

Je opmerking mbt de Issue date / Due date ga ik naar kijken, of en waar dit fout gaat. Ook de modified since creation date ) ga ik bekijken met name wat hier de impact van is en wanneer dit gerealiseerd kan worden aan onze kant.

Voor het aanleveren van meerdere facturen in één bericht wacht ik even op JoinData mbt het format dat we moeten aanleveren, voor jou informatie wij leveren een XML aan bij JoinData en zij zetten er een json schil omheen. Ik denk dat de json schil straks bij ons er om heen moet en er een array met facturen in de json komt. Maar ik wacht hier nog op input van JoinData, zij zullen dit met prio oppakken.

3feb22: uit de eDAIRY onderhoudscommissie:

- **Status:** Heeft de aandacht van RFC (**Jan**). Deze issue opnemen in het eerdergenoemde change register voor het monitoren van de status.

**Aandachtspunt:** Besluiten over welke changes backwards compatible moeten zijn en welke niet (datum vanaf); dit duidelijk in het change register aangeven.

**Status 9feb22:** In behandeling.

## 7. eDAIRY Invoice issues Qlip

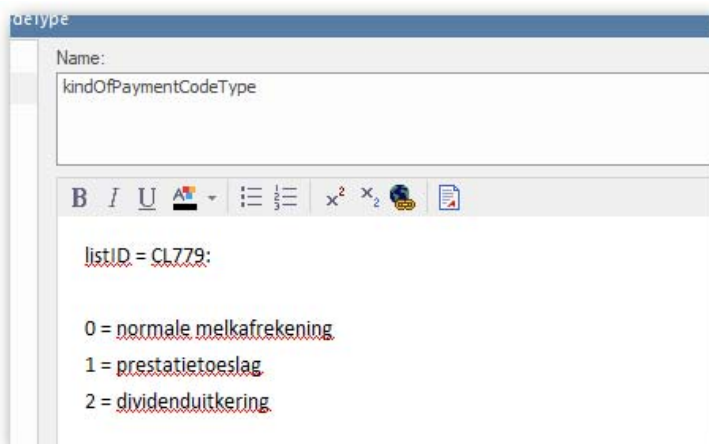
7mrt22, Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl> Aan: 'c.graumans@chello.nl' <c.graumans@chello.nl>

**Vraag 1:** In het invoice bericht komt het volgende element voor  
"kindOfPaymentcode" Indicates whether it concerns a regular payment or some kind of special payment.  
Dutch reference: ADED 204640 Afrekeningsvolgnummer:  
0 = in geval van een normale melkafrekening  
1 = in geval van een prestatietoeslag  
2 = in het geval van dividenduitkering.

→ Conny waar is de codelijst die hier bij hoort?

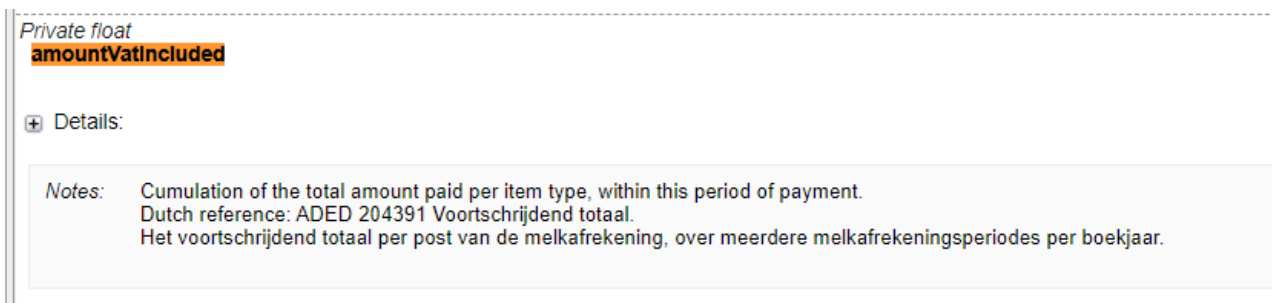
**Antwoord 1, 8mrt22\_cg:**

Staat in het eDairy class model:



Dit zijn alle enumeraties die hierop van toepassing zijn.

**Vraag 2:** amountVatIncluded



→ Conny : de omschrijvingen die hier bovenstaan lijken elkaar tegen te spreken: in de eerste zin wordt er gesproken over *Cumulation of the total amount paid per item type, within this period of payment* à dit zit momenteel ook in het EDI bericht  
In de laatste omschrijving (naar Aded) wordt er gesproken over totalen over meerdere melkafrekeningsperiodes per boekjaar (een soort totaal betaalde vat in het lopende jaar oid?) à dit zit momenteel NIET in het EDI bericht






Onze voorkeur gaat dus uit naar de eerstgenoemde omschrijving: kan jij hier ietst slims over zeggen?

**Antwoord 1, 8mrt22\_cg:**

De definitie van amountVatIncluded was fout. Het moet zijn:

Attribute

amountVatIncluded	float
-------------------	-------

**B** *I* U     $x^2$   $x_2$   

Total amount of money (VAT included) for this invoice line.

Ik heb dit in het model aangepast.

Status 8mrt22: Afgehandeld.

## 8. Correctie invoices in de Nederlandse zuivel

9mrt22: Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl> Aan: 'c.graumans@chello.nl' [c.graumans@chello.nl](mailto:c.graumans@chello.nl)  
CC: Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com>; 'Rene Rankenberg'  
[r.rankenberg@countus.nl](mailto:r.rankenberg@countus.nl)

Ik wil graag nog een onderwerp aandragen. Ik ben hier opgekomen doordat er in het UBL gevraagd wordt om aan te geven of het een normale factuur is, of een correctie factuur.

Nu heb ik in het verleden (ik geloof van Rene) gehoord dat de enige "correcte" manier om, om-te-gaan met een **correctie** als volgt is:

- Start: factuur wordt aangemaakt. Maar hier zit dus een fout in, dus wordt de oude factuur tegen geboekt (zodat alles weer op 0 loopt)
- Daarna wordt er een nieuwe factuur aangemaakt met de juiste gegevens.

In de zuivel wereld werkt dit iets anders (zowel voor RFC als alle zuivelfabrieken bij Qlip) :

- Als er een fout wordt gemaakt, wordt de oude factuur (uit de vorige periode) **vervangen** door de goede (zie ook beschrijving hieronder)
- Het financiële verschil wordt in de volgende periode als een **correctie regel** op de factuur van de volgende periode gezet. Deze werkwijze is al 30? 50? jaar de manier van werken in de Nederlandse zuivel en is niet zo maar even om te bouwen.

Dan nu mijn vraag: In de UBL wordt er gevraagd of een factuur dus een reguliere factuur is, of een correctie factuur, waarschijnlijk omdat er een andere workflow bij hoort: Ik verwacht dat de workflow in UBL gebaseerd is op de enig "correcte" manier zoals ik hierboven heb beschreven, maar dat gaat dan waarschijnlijk niet goed? @Rene (Rankenberg): kan jij hier iets over zeggen?

(Detail beschrijving hieronder weergegeven hoe wij gaan vaststellen of er sprake is van een correctie)

Correctie Invoices bij Partico:

- Stel het is nu maand 3, maand 2 is afgesloten en daar zat een fout in: de veehouder had meer vet geleverd dan eerst berekend, en daarom krijgt de veehouder een nabetaling van 1000 Euro in maand 3.
- Het gewijzigde vetpercentage, wordt meteen verwerkt in Z-Net (in periode 2)
- De factuur die er was in periode 2 wordt vervangen door de nieuwe factuur (met +1000 euro tov de vorige factuur als gevolg van het gewijzigde vetpercentage): In de database blijft er maar 1 normale factuur per periode (de geüpdate), de oude pdf blijft wel bestaan: er komt een nieuwe PDF bij die gebaseerd is op de nieuwe waarheid in de database.
- De 1000Euro extra, wordt toegevoegd als een factuur regel voor de factuur in periode 3.

Het is het plan om het aan de hand van de periode waarin de factuur gecorrigeerd is vast te stellen dat er een correctie is van een voorgaande periode. Stel: iemand vraagt uiteindelijk in maand 5 de factuurgegevens op van periode 3 & 4, dan zal de gecorrigeerde factuur van periode 2 ook meekomen, omdat de datum van de gecorrigeerde factuur in periode 3 valt. Een dergelijke factuur krijgt in de UBL opmaak in `<cbc:InvoiceTypeCode listID="UNCL1001">384</cbc:InvoiceTypeCode>` (384 = Corrective invoice) In geval van een normale factuur (niet gecorrigeerd) zou dit 389: Selfbilling invoice zijn.

Voor de opvragende partij zou dus het voorkomen van (384 = Corrective invoice) dus de trigger moeten zijn om in hun eigen systeem de oude eerst weg te gooien en te vervangen met de nieuwe versie.

En er moet logica zijn om in de factuur van periode 3, de correctie invoice line van +1000Euro ook goed af te handelen.

10mrt22: Van: Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com> Aan: Floris Ruiterkamp [ruiterkamp@qlip.nl](mailto:ruiterkamp@qlip.nl) CC: Steenbergen, R.M.G. van (Remco) <Remco.vanSteenbergen@FrieslandCampina.com>; Rene Rankenberg <r.rankenberg@countus.nl>; Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>; Conny Graumans (c.graumans@chello.nl) <c.graumans@chello.nl>; Steenbergen, M.K. (Mark) <mark.steenbergen@frieslandcampina.com>

Hi Floris, Wij hebben een ander beeld van het correctieproces voor melkgeldafrekeningen dan Qlip/ Partico. Zie bijgaand.

**Factuurgegevens**

Omschrijving	Hoeveelheid	Prijs	Per	BTW	Bedrag	Toelichting
Garantieprijs		45,00	100 KG			
Elwit garantieprijs	-1,47	721,96	100 KG	L	-10,61	Correctie periode 202201
Elwit garantieprijs	3.439,46	721,96	100 KG	L	24.831,53	
Vet garantieprijs	37,32	360,98	100 KG	L	134,72	Correctie periode 202201
Vet garantieprijs	4.225,83	360,98	100 KG	L	15.254,40	
Lactose garantieprijs	-2,92	72,20	100 KG	L	-2,11	Correctie periode 202201
Lactose garantieprijs	4.327,93	72,20	100 KG	L	3.124,77	
Vaste kosten	1,00	-131,00	1 ST	L	-131,00	
<b>Melkgeld totaal excl. BTW</b>					<b>43.201,70</b>	
Bijdrage ZuivelNL	94.722,00	-0,05	100 KG	G	-47,36	

**10mrt22: Analyse door Conny Graumans**

Volgens mij zouden we niet met "correctie" facturen gaan werken maar met "credit" facturen. Dus in het geval een factuur niet correct is dan dient deze volledig gecrediteerd te worden en dient een nieuwe factuur met een nieuw factuurnummer aangemaakt en verstuurd te worden. Maar als dit niet werkbaar is dan moeten we kijken hoe we hier een mouw aan passen.

De wijze waarop RFC het doet, met correctieregels in de eerstvolgende factuur, is denk ik een correcte oplossing.

**10mrt22:** Van: Rene Rankenberg <r.rankenberg@countus.nl> Aan: Smeenk, G.J. (Gerrit)

<Gerrit.Smeenk@frieslandcampina.com>; Floris Ruiterkamp <ruiterkamp@qlip.nl>

CC: Steenbergen, R.M.G. van (Remco) <Remco.vanSteenbergen@FrieslandCampina.com>; Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>; Conny Graumans (c.graumans@chello.nl)

<c.graumans@chello.nl>; Steenbergen, M.K. (Mark) <mark.steenbergen@frieslandcampina.com>

Onderwerp: RE: correctie invoices in de nederlandse zuivel.

Goedemorgen, Even een uitleg van werkwijze en gevolgen. Bij het inboeken wordt op basis van notadatum de factuur toegekend aan het tijdvak (maand). Bij het verzorgen van de btw-aangifte wordt voor het van toepassing zijnde tijdvak met bijbehorende facturen die in dat tijdvak vallen de btw-aangifte opgesteld.

Als nu na afloop van het tijdvak (x) EN na het verzorgen van de btw-aangifte (x) een nieuwe factuur binnenkomt MET factuurdatum die in dat tijdvak (x) valt, dan is de verzorgde aangifte BTW met terugwerkende kracht onjuist! En moet de procedure voor correctie worden uitgevoerd. Dat is handwerk, tijdrovend en foutgevoelig. Wij hebben dat op grote schaal meegemaakt bijvoorbeeld bij de correctie van de in 1e instantie rekening gebrachte btw op de fosfaatreductieheffing. Maar ook incidenteel hierna, vaak gerelateerd aan een specifieke zuivelonderneming.

Sterker nog: Wettelijk is vastgelegd dat de factuurdatum de datum is waarop de factuur wordt uitgereikt. Bij het corrigeren van een uitgereikte factuur dient dan ook een correctie op de oude factuur te worden opgesteld waarbij de systeemdatum de factuurdatum is voor de correctiefactuur en een nieuwe factuur, hiervoor geldt ook systeemdatum is factuurdatum, wordt uitgereikt.

Overigens stellen wij dus meteen wederom vast dat er feitelijk er dus een aantal zuivelondernemingen zijn die op dit moment een onjuiste systematiek factuurdatum hanteren. Immers laatste dag van de maand van levering is soms notadatum, terwijl de nota gemiddeld 7 tot ca 21 dagen later wordt opgesteld/uitgereikt.

[https://www.belastingdienst.nl/wps/wcm/connect/bldcontentnl/belastingdienst/zakelijk/btw/administratie\\_bijhouden/facturen\\_maken/facturen\\_ureisen/](https://www.belastingdienst.nl/wps/wcm/connect/bldcontentnl/belastingdienst/zakelijk/btw/administratie_bijhouden/facturen_maken/facturen_ureisen/)

**10mrt22:** Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl> Aan: 'Smeenk, G.J. (Gerrit)'

[Gerrit.Smeenk@frieslandcampina.com](mailto:Gerrit.Smeenk@frieslandcampina.com) CC: Steenbergen, R.M.G. van (Remco)

<Remco.vanSteenbergen@FrieslandCampina.com>; Rene Rankenberg <r.rankenberg@countus.nl>;

Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>; Conny Graumans

(c.graumans@chello.nl) <c.graumans@chello.nl>; Steenbergen, M.K. (Mark)

<mark.steenbergen@frieslandcampina.com>

Als eerste een reactie op Rene's mail: we zijn nu bezig met de overgang van EDI naar E-Dairy. We zullen het moeten doen met de werkwijze die we nu hebben. En ook al komt deze niet 100% overeen met hoe het zou moeten volgens het boekje, de huidige methode van werken is denk ik al 40 jaar (?) zo : Dit aanpassen is echt een enorme operatie waarvan ik me sterk afvraag of we dat wel moeten willen doen met zijn allen.

Dank voor de aanvulling Gerrit : Onze processen verschillen inderdaad toch een beetje. Wat overeen blijft is wel dat we beiden een correctie proces hebben dat niet overeenkomt met de gangbare boekhoudprocessen.



Echter jullie proces zorgt er voor dat er nooit een InvoiceTypeCode = 384 (384 = Corrective invoice) ontstaat, omdat er gewoon negatieve invoicelines ontstaan. Dus binnen UBL afhandeling geen probleem gok ik (klopt Rene?)

Bij ons hebben we dus wel een correctie invoice en daarom blijft de vraag relevant: Kan er in UBL worden omgegaan met een InvoiceTypeCode = 384 zonder dat deze volgens de gangbare manier is tegen geboekt?

10mrt22, resultaat overleg onderhoudscommissie:

- Floris (Ruiterkamp) overlegt dit verder met René Rankenberg (Countus). Komt in het volgende overleg terug op de agenda.

Status 10mrt22: In behandeling.

## 9. Invoice, uitsplitsen leveringen

10mrt22: Van: Floris Ruiterkamp (Qlip)

Ter vergadering van de Onderhoudscommissie ingebracht door Floris.

Geconstateerd is dat RFC op haar invoices ook informatie op het niveau van de afzonderlijke leveringen mee geeft; conform de RFC papieren factuur. Daar waar Qlip op haar facturen uitsluiten de geaggregeerde data over alle leveringen in de betreffende periode mee geeft. Ontvangers zullen dit verschil zien. De vraag is of dit aan de ontvangende kant tot problemen gaat lijden.

Status 10mrt22: Deze issue in het volgende overleg aan de orde stellen.

To: Hoogenberg, J. (Jan) <[Jan.Hoogenberg@frieslandcampina.com](mailto:Jan.Hoogenberg@frieslandcampina.com)>

Subject: RE: eDAIRY ubl-invoice met json tags eromheen

Waar komen de backslashes `\` en `/` vandaan? Die horen niet thuis in een xml-bericht.

```
<n2:Invoice xmlns:n2=\"urn:oasis:names:specification:ubl:schema:xsd:Invoice-2\" xmlns:cac=\"urn:
\urn:oasis:names:specification:ubl:schema:xsd:CommonBasicComponents-2\" xmlns:ssi=\"http://ww
https://docs.oasis-open.org/ubl/os-UBL-2.1/xsd/maindoc/UBL-Invoice-2.1.xsd\"><cbc:UBLVersionID>
\cbc:ProfileID><cbc:ID>0092002494<\cbc:ID><cbc:IssueDate>2018-01-31<\cbc:IssueDate><cbc:Due
Afrekening melkgeld januari 2018<\cbc>Note><cbc:DocumentCurrencyCode listID=\"ISO 4217 Alpha
cbc:DescriptionCode listID=\"CL???\">2018010<\cbc:DescriptionCode<\cac:InvoicePeriod><cac:i
cac:Attachment><EmbeddedDocumentBinaryObject mimeType=\"application/pdf\" filename=\"Melkgeld
JVBERI0xLjMNCiXi48/TDQoLUlNUWFBERJmUGFyYnNCi9TnbmZvIDI0IDAgUg0KPj4nCnN0YXk30eHJlZG0KMzZNDQNC
AgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIC
cbc:DocumentHash>6AD7EF57154A6633BBB42ACEBACC42933<\cbc:DocumentHash><cbc:HashAlgorithmMethod>
cac:AccountingSupplierParty><cbc:CustomerAssignedAccountID schemeID=\"nl.plnt\">001780250<\cb
cac:PartyIdentification><ID schemeID=\"nl.kvk\">32164040<\cbc:ID><\cac:PartyIdentification>
/cbc:StreetName><cbc:CityName>LEUSDEN<\cbc:CityName><cbc:PostalZone>3831 SB<\cbc:PostalZone>
cac:PartyTaxScheme><cbc:RegistrationName>HG van Burgsteden<\cbc:RegistrationName><cbc:Company
/cac:PartyTaxScheme><cbc:PartyLegalEntity><cbc:RegistrationName>HG van Burgsteden<\cbc:Regis
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- eDAIRY test message RFC, modified by Conny graumans 26jan21 -->
<n2:Invoice xsi:schemaLocation="urn:oasis:names:specification:ubl:schema:xsd:Invoice-2 https://docs.oasis-
urn:oasis:names:specification:ubl:schema:xsd:Invoice-2" xmlns:cac="urn:oasis:names:specification:ubl:schem
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <cbc:UBLVersionID>UBL v2.1</cbc:UBLVersionID>
  <cbc:CustomizationID>UBL-invoice-v2.1-eDAIRY</cbc:CustomizationID>
  <cbc:ProfileID>UBL-invoice-v2.1-eDAIRY</cbc:ProfileID>
  <cbc:ID>0092093766</cbc:ID>
  <cbc:IssueDate>2018-07-31</cbc:IssueDate>
  <cbc:DueDate>2018-08-08</cbc:DueDate>
  <cbc:InvoiceTypeCode listID="UNCL1001">389</cbc:InvoiceTypeCode>
  <cbc:Note languageID="NL">Afrekening melkgeld juli 2018</cbc:Note>
  <cbc:DocumentCurrencyCode listID="ISO 4217 Alpha">EUR</cbc:DocumentCurrencyCode>
  <cac:InvoicePeriod>
    <cbc:StartDate>2018-07-01</cbc:StartDate>
    <cbc:EndDate>2018-07-31</cbc:EndDate>
    <cbc:DescriptionCode listID="CL???">2018070</cbc:DescriptionCode>
  </cac:InvoicePeriod>
  <cac:AdditionalDocumentReference>
    <cbc:ID>AB_MGA-0092093766</cbc:ID>
```

Verder kwam ik in het ubl-invoice xml voorbeeld (onderdeel van de aangeleverde Json array) de volgende fouten tegen die hersteld moeten worden:

<ID schemeID="nl.kvk"> moet zijn <cbc:ID schemeID="nl.kvk">

```
<cbc:EmbeddedDocumentBinaryObject mimeType="application/pdf"
```

<|D scheme|D="nl.plnt">001</cbc:|D> moet zijn <cbc:|D scheme|D="nl.plnt">001</cbc:|D>

<cac:GoodsItem><cac:Item> moet zijn <cac:GoodsItem><cac:Item>

filename="Melkgeldnota\_201801\_0000780250\_20180208\_135652.pdf">JVBERi0xLjMNCiXi48TDQolUINUWF  
etc. etc... Zie bijgevoegd voorbeeld.

`mimeCode="application/pdf" filename="Melkgeldnota_201801_0000780250_20180208_135652.pdf">` moet zijn  
(volgorde omdraaien) `filename="Melkgeldnota_201801_0000780250_20180208_135652.pdf"`  
`mimeCode="application/pdf">`

`<cbc:ActualDeliveryDate/>` datum invullen plus eindtag `<cbc:ActualDeliveryDate>2018-07-01</cbc:ActualDeliveryDate>`

`<TaxSubtotal>` moet zijn `<cac:TaxSubtotal>`

Graag bereid tot nader overleg, Groet, Conny Graumans, AgroConnect

29mrt22: Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: c.graumans@chello.nl  
Conny, De backslashes ( \ " en \ " ) zijn nodig om dit in json in te bedden (escape characters ), zonder deze krijg je syntax fouten in de json. Ik heb ook de versie van JoinData opgevraagd en deze is hetzelfde als die wij nu hebben. Misschien goed als ik eens schakel met VanAaken om te kijken hoe zij dit zien, is tenslotte belangrijkste developer van onze klanten.

29mrt22: Conny aan Jan:  
...en hoe krijgt de ontvanger die slashes er dan weer uit; inclusief de slashes is de xml niet valide.

29mrt22: Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: c.graumans@chello.nl  
Ik heb zojuist met Jan Herrygers gesproken, volgens hem kan hij dat handmatig met XMLSpy doen door de JSON in te lezen en dan wordt deze gede-escaped. Klonk als niet ingewikkeld, maar ik heb er geen verstand van. Tevens geeft hij aan dat als het handmatig met XMLSpy kan dan moet het met andere tooling ook automatisch kunnen.

Hij stuurde de volgende link: <https://www.altova.com/json-tools>

Verder vond ik nog het volgende op internet:

Escape XML in JSON - JSON String Escape / Unescape

- Backspace is replaced with \b.
- Form feed is replaced with \f.
- Newline is replaced with \n.
- Carriage return is replaced with \r.
- Tab is replaced with \t.
- Double quote is replaced with \"
- Backslash is replaced with \\

29mrt22, Van: c.graumans@chello.nl <c.graumans@chello.nl> Aan: 'Hoogenberg, J. (Jan)' <Jan.Hoogenberg@frieslandcampina.com>

...maar Jan, dat moet je toch niet willen dat vanwege JoinData iedereen zijn xml ubl-invoice moet opschonen; of wordt dat dan weer een extra service die JoinData aanbiedt?

30mrt22: uit telefonisch overleg met Jan Herrygers (VAA).

Is een keuze van JoinData om alles in Json te doen. Betekent inderdaad dat alle xml-berichten die via JoinData uitgevraagd worden, worden vertaald naar Json varianten waarbij de noodzakelijke escape-characters (backslash combinaties) in het bericht worden ingevoerd. De ontvangende partij ontvangt dan geen xml maar een Jason variant daarop. Mocht de ontvangende partij dat weer terug willen vertalen naar Json dan dient deze de escape-characters weer te verwijderen.

**Conclusie** Conny Graumans: Het is vreemd dat een intermediaire partij (JoinData) de berichten inhoudelijk aanpast en xml-berichten vertaalt naar een Json variant daar waar de ontvangende partij eigenlijk xml dacht te ontvangen. De vraag is of je de xml ubl-invoices wel via JoinData zou moeten willen uitwisselen; er zijn alternatieve wegen die de bericht syntax ongemoeid laten en netjes xml doorleveren.

31mrt22: Van: c.graumans@chello.nl <c.graumans@chello.nl> Aan: 'Gijs@join-data.nl' <Gijs@join-data.nl>  
CC: Gerrit Smeenk (FrieslandCampina) <Gerrit.Smeenk@frieslandcampina.com>

Beste Gijs, Zie bijgevoegde issuelog. Kun je aangeven wat hier precies gebeurt?

Is het inderdaad zou dat wanneer partijen via JoinData de xml ubl-edairy-invoice opvragen een Json variant krijgen aangeleverd?

6apr22: Van: Gijs Nijman <Gijs@join-data.nl> Aan: c.graumans@chello.nl  
CC: Gerrit Smeenk (FrieslandCampina) <Gerrit.Smeenk@frieslandcampina.com>

Ja, we gooien er een json wrapper omheen (de joindata envelop). Dat doen we omdat er meta data toegevoegd kan worden en meerdere bronnen benaderd kunnen worden. Om een XML bericht in een JSON bericht te doen dient deze volgens, in overeenstemming met de JSON standaard geescaped te worden.

Elke tool die JSON inleest zou dit automatisch moeten de-escapen. Dit is niet anders dan al onze huidige api's. Ook de XML-leverEAB en XML-FaktuurEAB worden op dezelfde manier aangeboden. Hopende hiermee voldoende antwoord te hebben gegeven.

7apr22: Van: c.graumans@chello.nl <c.graumans@chello.nl> Aan: 'Gijs Nijman' <Gijs@join-data.nl>  
CC: 'Gerrit Smeenk (FrieslandCampina)' <Gerrit.Smeenk@frieslandcampina.com>;  
'jan.hoogenberg@frieslandcampina.com' <jan.hoogenberg@frieslandcampina.com>  
OK Gijs, Kun je garanderen dat met deze conversie van xml -> json -> xml er geen risico is dat data in het bericht verandert?

7apr22: Gijs: Ja, dat kunnen we garanderen.

8apr22, Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl> Aan: 'c.graumans@chello.nl' [c.graumans@chello.nl](mailto:c.graumans@chello.nl)  
CC: Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com>  
Onderwerp: FW: Agenda AgroConnect Onderhoudscommissie eDAIRY, do 7apr22, 10:00 uur  
Nav van de discussie van gisteren heb ik Wesley gevraagd en hij ziet weinig tot geen beren op de weg.  
Groeten Floris:

7apr22, Van: Wesley van de Kamp <kamp@qlip.nl>

Ik denk persoonlijk dat het probleem niet zo heel groot is. Dit is het gevolg als je een XML bericht in een JSON gaat wrappen. Het is dan heel normaal om escape characters te gebruiken. Sterker nog, dit is zelfs verplicht omdat je anders geen valide JSON hebt. Daarnaast kan vrijwel elke library hier mee omgaan en zal deze automatisch "gede-escaped" worden. Het wordt misschien een probleem als je zelf iets gebouwd hebt. Gevolg is dan inderdaad dat je zelf deze escape characters er uit zal moeten filteren. Maar ook dat stelt eerlijk gezegd niet zoveel voor...

Status 8apr22: Afgesloten.



## 11. PaymentMeansCode

21mrt22: Van: Floris Ruiterkamp (Qlip), uit telefoongesprek met Conny

Vraag van Floris: Wat moet er voor PaymentMeansCode worden ingevuld? Qlip is voornemens om default code 58 in te vullen.

<b>Invoice. Payment Means</b>	<b>Definition</b>	The value of this item property, expressed as a quantity
	<b>Occurrence</b>	0 .. unbounded
	<b>Type</b>	Payment Means. Details
	<b>Definition</b>	Expected means of payment.
Payment Means. Payment Means Code. Code	<b>Occurrence</b>	1 .. 1
	<b>Type</b>	Payment Means_ Code. Type
	<b>Definition</b>	A code signifying the type of this means of payment.
	<b>Remark</b>	Dairy: "UNCL4461". E.g.: 18= a debit transaction.
Payment Means. Payment Due Date. Date	<b>Occurrence</b>	0 .. 1
	<b>Type</b>	Date. Type
	<b>Definition</b>	The date on which payment is due for this means of payment.

22mrt22, analyse door Conny Graumans:

In de ubl-invoice is PaymentMeansCode een verplicht data-element.

De toegestane codes zijn gepubliceerd middels: UNECE\_PaymentMeansCode\_D20A.xsd

Code 58 staat voor: "Credit transfer inside the Single Euro Payment Area (SEPA) system."

Het geeft aan door middel van welke techniek (credit card, automatische afschrijving, etc.) de betaling wordt uitgevoerd.

value	ns2:Name	ns2:Description
A 51	FR, norme 6 97-Telereglement CFONB (French Organisation for Banking Standards) - Option	A French standard procedure that allows a debtor to p
A 52	Urgent commercial payment	Payment order which requires guaranteed processing
A 53	Urgent Treasury Payment	Payment order or transfer which must be executed, b
A 54	Credit card	Payment made by means of credit card.
A 55	Debit card	Payment made by means of debit card.
A 56	Bankgiro	Payment will be, or has been, made by bankgiro.
A 57	Standing agreement	The payment means have been previously agreed bet
A 58	SEPA credit transfer	Credit transfer inside the Single Euro Payment Area (S
A 59	SEPA direct debit	Direct debit inside the Single Euro Payment Area (SEP
A 60	Promissory note	Payment by an unconditional promise in writing made
A 61	Promissory note signed by the debtor	Payment by an unconditional promise in writing made
A 62	Promissory note signed by the debtor and endorsed by a bank	Payment by an unconditional promise in writing made
A 63	Promissory note signed by the debtor and endorsed by a third party	Payment by an unconditional promise in writing made
A 64	Promissory note signed by a bank	Payment by an unconditional promise in writing made
A 65	Promissory note signed by a bank and endorsed by another bank	Payment by an unconditional promise in writing made
A 66	Promissory note signed by a third party	Payment by an unconditional promise in writing made
A 67	Promissory note signed by a third party and endorsed by a bank	Payment by an unconditional promise in writing made
A 68	Online payment service	Payment will be made or has been made by an online
A 69	Transfer Advice	Transfer of an amount of money in the books of the ac
A 70	Bill drawn by the creditor on the debtor	Bill drawn by the creditor on the debtor.

In de Qlip self-billing ubl-invoice vult Qlip voor PaymentMeansCode standard code 58 in.

In de eDAIRY Json invoice wordt PaymentMeansCode niet gebruikt.

Voorstel aan Qlip en RFC: Voor de self-billing ubl-invoice standaard code 58 "SEPA credit transfer" invullen.

Status 22mrt22: In het eerstvolgende overleg van de eDAIRY onderhoudscommissie aan de orde stellen.



## 12. GRAS codes

25mrt22: Van: Floris Ruiterkamp <Ruiterkamp@qlip.nl> Aan: 'c.graumans@chello.nl' <c.graumans@chello.nl>

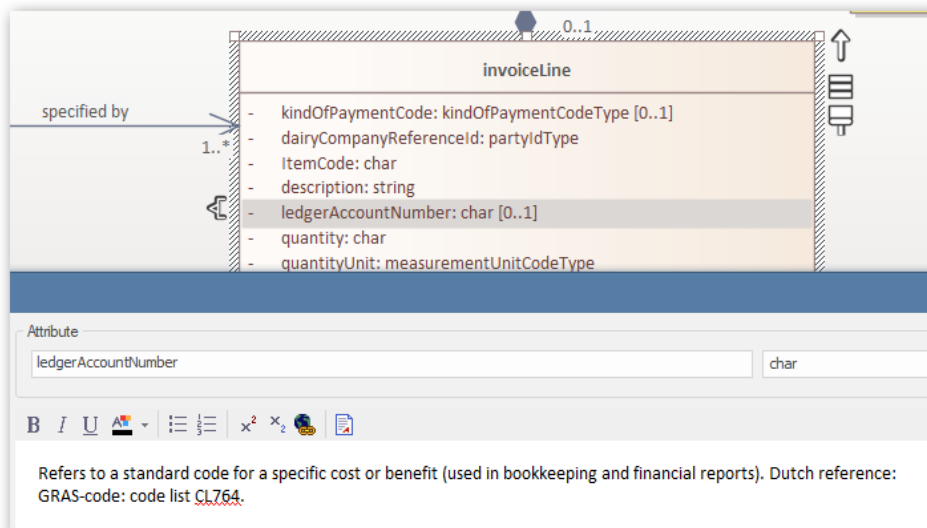
Ik krijg van meerdere mensen feedback (bij ons intern, maar ook van accountants) dat de GrasCode in de praktijk helemaal niet meer gebruikt wordt. In het kader van “een schone start” zijn wij dan ook van plan om deze niet meer op te nemen in het invoice bericht. Zou jij dit kunnen agenderen in de komend Agroconnect overleg.

27mrt22: Van: c.graumans@chello.nl <c.graumans@chello.nl> Aan: 'Floris Ruiterkamp' [Ruiterkamp@qlip.nl](mailto:Ruiterkamp@qlip.nl)  
CC: Gerrit Smeenk (FrieslandCampina) <Gerrit.Smeenk@frieslandcampina.com>  
OK Floris, Neem ik mee naar de Onderhoudscommissie van 7apr22.

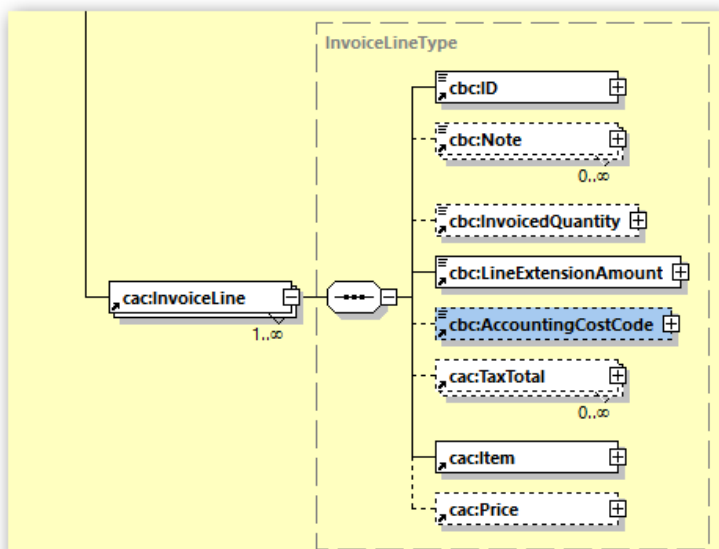
27mrt22, analyse door Conny Graumans:

GRAS staat voor Geüniformeerd Rekeningschema Agrarische Sector en is een standaard grootboek codering die indertijd door LEI en VLB-accountantskantoren is vastgesteld.

In de eDAIRY Json-invoice is deze als volgt als optioneel data-element gedefinieerd:



In de eDAIRY ubl-invoice is deze als volgt als optioneel data-element gedefinieerd:



Invoice Line. Accounting Cost Code. Code	Occurence	0 .. 1
	Type	Code. Type
	Definition	The buyer's accounting cost centre for this invoice line, expressed as a code.
	Remark	Dairy: ADIS: 201088 Rekeningnummer (e.g. the GRAS-code).

**Voorstel AgroConnect:** Gewoon als optioneel data-element laten staan, hoeft niet gevuld te worden.

**Status 25mrt22:** In het eerstvolgende overleg van de eDAIRY onderhoudscommissie aan de orde stellen.

### 13. InvoiceLine - identifier

11mei22: Van: Floris Ruitkamp <Ruitkamp@qlip.nl>

Aan: 'c.graumans@chello.nl' <c.graumans@chello.nl>

Onderwerp: foutje in de UBL documentatie?

De drie Jasons (deliveries, kwaliteit en invoice) hebben we aan de praat en getest en JoinData is nu bezig om ons aan te sluiten op hun integratie omgeving.

We zijn nu met de UBL invoice bezig en ik kwam (denk ik) een foutje tegen in de documentatie. Zie hieronder:

In de guideline heb ik de volgende fout gevonden. Op pagina 10 staat

Invoice Line. Identifier	Occurrence	1 .. 1
	Type	Identifier. Type
	Definition	An identifier for this invoice line.
	Remark	Dairy: E.g. a sequence number for the invoice lines that are on the same invoice. ADIS: 204640 Afrekeningsvolgnummer.

Maar als je adis 204640 erbij pakt, dan staat er heel iets anders.

(Volgnummer om onderscheid te kunnen maken tussen regulieren melkafrekeningen, nabetalingen en dividend) 0 = regulier 1 tot 9 andersoortige uitbetalingen)

Als ik zie hoe RFC deze heeft opgepakt, dan is het gewoon een identifier van een regel geworden.

12mei22: analyse Conny

Vraag aan Jan Hoogenberg (RFC): Jan, hoe vullen jullie dit veld nu? Gewoon met een volgnummer voor de factuurregels of doe je ook nog iets met '...0 = regulier 1 tot 9 andersoortige uitbetalingen...'. Als het eerste het geval is (een volgnummer), dan pas ik de implementatiegide daarop aan.

Status 12mei22: In behandeling.

## 14. Ubl-invoice voor Countus

28apr22: From: Hoogenberg, J. (Jan) To: Rene Rankenberg <r.rankenberg@countus.nl>; Renze van de Kuilen <Renze@join-data.nl> Cc: David Goolooze <david.goolooze@visma.com>; Jan Herrygers <jherrygers@vaa.com>; Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com>; 'Andre van der Kamp' <avdkamp@alfa.nl>; Ton Langens <ton.langens@abab.nl>; Edwin Oonk <Edwin.Oonk@flynth.nl>  
Subject: RE: Voortgang eDairy auto-invoice

Rene, Op dit moment is de nieuwe uitvoering van de UBL-factuur, ingepakt in het bediscussieerde JSON, gereed op onze ontwikkelomgeving. Na een migratie van deze omgeving zijn er helaas enige nieuwe technische problemen ontstaan. We hebben de meeste interfaces op onze ontwikkel- en testomgevingen weer in de lucht en kunnen nu hier ook weer mee verder.

De volgende stappen moeten gedaan worden:

- laatste check op RFC ontwikkelomgeving à Jan Hoogenberg
- Overzetten naar RFC integratie/test omgeving à RFC PO-team
- Aanpassingen JoinData ( verwijderen wrapper ) à JoinData
- Inhoudelijke checks RFC à Jan Hoogenberg
- en parallel testmogelijkheid/controle inhoud door developers accountants

Hoe zie ik dit in de tijd:

- vrijdag 29/04 t/m 02/05 laatste check en overzetten naar integratie/test omgeving
- informeren JoinData, als eea beschikbaar is op integratieomgeving, 02/05

Hierna is eea afhankelijk van JoinData om de wrapper eraf te halen, RFC kan wel starten met inhoudelijke controles. Ik zal een reminder in mijn agenda zetten voor de komende weken om een voortgangsrapportage uit te sturen. Het blijft een 'bumpy road' de UBL.

6mei22, From: Hoogenberg, J. (Jan) To: 'Rene Rankenberg' <r.rankenberg@countus.nl>; 'Renze van de Kuilen' <Renze@join-data.nl> Cc: 'David Goolooze' <david.goolooze@visma.com>; 'Jan Herrygers' <jherrygers@vaa.com>; Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com>; 'Andre van der Kamp' <avdkamp@alfa.nl>; 'Ton Langens' <ton.langens@abab.nl>; 'Edwin Oonk' <Edwin.Oonk@flynth.nl>  
Subject: RE: Voortgang eDairy auto-invoice

Rene, Renze, Tijdens de controle op 2 Mei bleek dat de aanpassing niet goed was doorgekomen op de testomgeving. Dit is nu hersteld en de JSON schil zit er nu omheen. Inhoudelijk is er nog een issue met de factuur ( in het XML-deel ) bij meerdere facturen in de JSON en we werken aan de oplossing hiervan. Ik hoop dat dit vandaag of maandag is opgelost. @Renze van de Kuilen: JoinData moet nu het bericht een op een gaan doorgeven en niet meer de embedding in JSON doen, kun je dit in gang zetten?

13mei22, Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com> Aan: Rene Rankenberg <r.rankenberg@countus.nl>; Renze van de Kuilen [Renze@join-data.nl](mailto:Renze@join-data.nl) CC: David Goolooze <david.goolooze@visma.com>; Jan Herrygers <jherrygers@vaa.com>; Smeenk, G.J. (Gerrit) <Gerrit.Smeenk@frieslandcampina.com>; 'Andre van der Kamp' <avdkamp@alfa.nl>; Ton Langens <ton.langens@abab.nl>; Edwin Oonk <Edwin.Oonk@flynth.nl>; c.graumans@chello.nl  
Onderwerp: RE: Voortgang eDairy auto-invoice

Rene, Renze, De UBL vanuit FrieslandCampina staat op onze testomgeving ook de inhoudelijke fout is opgelost. Vanuit Renze kreeg ik door dat de aanpassing bij JoinData bovenaan de backlog staat en in de komende sprint wordt opgepakt, ik heb hier geen exacte datum voor gekregen. Volgens mij is er niemand die op dit moment op de productieve omgeving de UBL ophaalt en ik ben voornemens om de wijzigingen door te zetten naar de productieomgeving om het "Work In Progress" op te schonen.

Status 15mei22: In behandeling.

## 15. Fouten in XML eDairy UBL Invoice FrieslandCampina

6jul22: Van: Jan Herrygers <JHerrygers@vaa.com> Aan: Hoogenberg, J. (Jan)  
[Jan.Hoogenberg@frieslandcampina.com](mailto:Jan.Hoogenberg@frieslandcampina.com) CC: Conny Graumans <c.graumans@chello.nl>

Onderwerp: Fouten in XML eDairy UBL Invoice FrieslandCampina

Jan, Zojuist heb ik jou 2 UBL-bestanden gestuurd. Het ene bestand is wat we een tijdje geleden hebben opgevraagd bij de JoinData integration omgeving, het andere bestand is dezelfde UBL die ik heb aangepast.

59761881\_2021-02.xml is het bericht zoals we dat een keer via JoinData hebben opgevraagd.

Dit bestand bevat een aantal fouten waardoor het niet kan worden verwerkt met onze XML tooling:

- op verschillende plaatsen wordt de namespace prefix vergeten bij het openen van een ID-element. Bij het sluiten wordt deze wel meegenomen: cbc:ID.
- het element "GoodsItem" wordt weergegevens als "GoodsItem;". De komma punt aan het einde zou daar niet mogen staan. (zowel bij openende als de afsluitende tag)
- bij het EmbeddedDocumentBinaryObject-element ontbreekt de namespace prefix: cbc:EmbeddedDocumentBinaryObject (zowel bij openende als de afsluitende tag)
- bij het TaxSubtotal-element ontbreekt de namespace prefix: cac:TaxSubtotal. (zowel bij openende als de afsluitende tag)

In het bestand 59761881\_2021-02\_altered.xml heb ik bovenstaande fouten opgelost. Dit bestand wordt als geldig aangegeven bij deze online validator: <https://ecosio.com/en/peppol-and-xml-document-validator/> met ruleset "UBL Invoice 2.1". Kan je deze punten bekijken en oplossen bij het genereren van de UBL?

7jul22: Conny aan Jan en Jan:

Jan en Jan, Dit lijken me van Jan (Herrygers) inderdaad terecht geconstateerde tekortkomingen die moeten worden rechtgetrokken.

19jul22, Van: Hoogenberg, J. (Jan) <Jan.Hoogenberg@frieslandcampina.com>

Aan: c.graumans@chello.nl; 'Jan Herrygers' <JHerrygers@vaa.com>

De geconstateerde gebreken zijn opgelost en de UBL is beschikbaar op de testomgeving. Hij gaat deze week ook door naar productie.

Status 19jul22: Afgehandeld.

-----  
-----