Variable data fra fagsystemer

Indhold

1. Overordnet datastruktur 3

1.1 Header 4

1.2 Body 4

2. Eksempler på data i body 5

2.1 Variable 5

2.2 Tabel med et niveau 6

2.3 Tabel i tabel 6

2.4 Binære data 6

3. Anvendelse af data til bygning af skabeloner 7

3.1 Fagsystemets data 7

3.2 Data set i PrintNet T Designer XML Input 8

3.3 Data set i PrintNet T Designer Layout 9

3.4 Data set i PrintNet T Designer Proof 10

3.5 PrintNet Interactive 11

4. Anvendelse af data til kald af MeddelelseMultiSend 12

4.1 Kald til MeddelelseMultiSend 12

Appendix A 14

|  |  |  |  |
| --- | --- | --- | --- |
| Historik |  |  |  |
| Dokumentid. | Dato | Udarb. af | Ændringer |
| <dok.id.> | 22.12.2010 | Helle Kristensen og Henrik Nørby | Oprettet |

# Overordnet datastruktur

Alle skabelonsæt til A&D opbygges i PrintNet, hvor hvert skabelonsæt består af to adskilte dele:

Et layout og en datadel. Nærværende notat har fokus på datadelen.

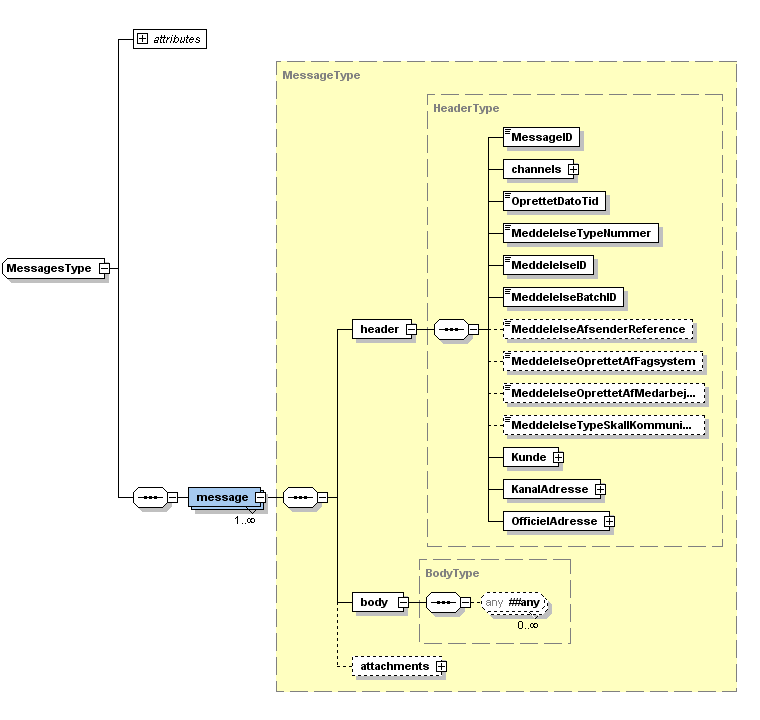
Det enkelte Fagsystem kalder servicen MeddelelseMultiSend eller MeddelelseMultiSendEkspres med det dataformat, der er aftalt via xsd’en for disse services.

I A&D beriges Meddelelserne så med yderligere data, og de berigede data sendes til PrintNet

Interfacet mellem A&D og PrintNet beskriver den samlede datastruktur for en skabelon, idet alle datastrukturer der indgår i denne kan anvendes som strukturer, dvs. variable, tabeller, tekster etc. i det endelige brev/mail.

Dette interface behøver man ikke kende i detaljer for at kunne bygge nye data-strukturer med variable data til skabelonsæt, men man skal naturligvis vide, hvilke data der indgår, så man ikke opbygger variable datasæt, der indeholder oplysninger, der allerede findes.

Kort beskrevet består datasættet af en header del og en body del, hvor header delen er fast og derfor er til rådighed i alle skabeloner, mens bodydelen er helt fri og kan opbygges præcis som det ønskes.



Figur PrintNet interface

## Header

Header delen består af berigede data og indeholder:

* Kundenummer (CPR-nummer, SE-nummer eller UKD-nummer)
* Den adresse meddelelsen skal sendes til (e-mailadresse, post adresse, telefonnummer)
* Kundens officielle adresse (post adresse)
* Diverse informationer om Meddelelsen: Meddelelsestype, MeddelelseID, Fagsystem, Afsenderreference etc.
* Diverse styringsparametre til PrintNet (Om meddelelsen skal sendes som e-mail, brev, sms, ”kun i kommunikationmappe”, hvilken skabelon der skal benyttes, sprog etc.)

Se evt. Appendix A for yderligere detaljer.

## Body

Body delen består af de variable data, som denne skabelon skal anvende.

De variable data fastlægges frit af fagsystemet ud fra de data skabelon-byggerne har brug for til opbygning af brev eller e-mail. Her kan medsendes enkeltvariable såsom journalnummer, betalingsdato etc. eller lister af data.

Datasættene kan opbygges helt frit, men der er selvfølgelig en række overvejelser man bør gøre sig:

Man kan f.eks. vælge at datasæt og skabelon skal stemme 100 % overens. I det tilfælde skal man opbygge en data-fil for hver eneste skabelon.

Hvis den ene skabelon skal bruge 5 variable og den anden skabelon skal bruge de samme 5 variable + 2 mere, så skal der laves to datasæt, og det korrekte datasæt skal sendes med MeddelelseMultiSend.

Fordelen ved dette er, at der er 100 % kontrol over datasæt og skabeloner, mens ulempen er, at hvis der skal tilføjes en ny variabel til 10 skabeloner, der ligner hinanden, så skal der laves tilføjelser til 10 datasæt.

Man kan også vælge, at hver skabelon kun skal anvende en delmængde af datasættet. Hermed kan man opbygge et stort datasæt, og så lade hver skabelon anvende de data, der er behov for. Det komplette datasæt skal så sendes med til alle kald af MeddelelseMultiSend for alle de skabeloner, der anvender datasættet, men der vil naturligvis være tomme variable/tabeller for nogle af skabelonerne.

Fordelen ved dette er, at det er nemt at lave en rettelse til data-strukturen for alle skabeloner, der anvender samme datasæt, mens ulempen er at der ikke er 100 % kontrol over hvilke data der rent faktisk er udfyldt i den enkelte situation.

De to yderste ekstremer er således én datafil pr skabelon eller ét stort samlet datasæt til alle skabeloner. Det fornuftigste er formentlig at finde en balance, hvor en række skabeloner, der ligner hinanden, er bygget på samme datasæt.

# Eksempler på data i body

Stort set alle slags data kan sendes i body, så længe skabelonbyggeren har værktøjer til at fortolke dem.

I nedenstående er der nogle helt basale eksempler.

* Dato i formatet ddmmåå.
  + <<dato>>031110<</dato>>
  + Skabelonbyggeren kan anvende PrintNet standardfunktioner til at præsentere datoen som ”Mandag den tredje november”.
* Tekst
  + <<minTekst>> Her er en tekst der kan være så lang som ønsket. Den kan være på flere 100 ord, en enkelt værdi eller bare et enkelt bogstav.<</minTekst>>.
  + Skabelonbyggeren kan anvende PrintNet standardfunktioner til f.eks. orddeling, font-størrelser etc.
* Binære data (base64 encodede)
  + <<minTiff>>TWFuIGlzIGRpc3Rpbmd1aXNoZWQsIG<</minTiff>>
  + Skabelonbyggeren kan anvende PrintNet standardfunktioner til at decode, tælle sider i en multipage tiff og modificere og placere billedet som ønsket.

Når man opbygger datamodellen er den vigtigste overvejelse man bør gøre sig, hvorvidt en variabel kan forekomme 0, 1 eller flere gange i den samme meddelelse. I virkeligheden de samme overvejelser man vil gøre sig, når man laver en DTD, et XML schema eller laver tabeller.

Når skabelonbyggeren skaber layoutet, der skal præsentere data vil han være tvunget til at forholde sig til, hvordan data skal præsenteres og hermed også tage højde for antallet af gentagelser. Hvis skabelonbyggeren ikke har designet layoutet efter dette fra starten, kan det betyde meget ekstra arbejde.

Et simpelt eksempel kunne være, at der i meddelelsen skal sendes et telefonnummer. Skabelonbyggeren vil umiddelbart placere variablen med telefonnummeret i dokumentet, men hvis der er flere forekomster af telefonnummeret fra tid til anden, skal han beslutte, om de skal skrives kommasepareret i rækkefølge, nedenunder hinanden, i en tabel, om tomme værdier skal fjernes eller andet.

Når man som databygger giver et eksempel på data til skabelondesigneren er det derfor vigtigt at beskrive dette samtidigt.

## Variable

Både elementer og attributter kan anvendes som variable i Layout.

<customer>

<name>Gabriel Okara</name>

<occupation>Poet</occupation>

</customer>

Giver samme muligheder som

<customer name="Gabriel Okara">

<occupation>Poet</occupation>

</customer>

## Tabel med et niveau

Et typisk eksempel kunne være følgende:

<payments>

<payment>

<description>Cheese</description>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>Gum</description>

<dueDate>2010-05-03</dueDate>

</payment>

</payments>

## Tabel i tabel

Et typisk eksempel kunne være følgende:

<payments>

<payment>

<description>

<name>Cheese</name>

<origin>Cow</origin>

</description>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>

<name>Wood</name>

<origin>Tree</origin>

</description>

<dueDate>2010-05-03</dueDate>

</payment>

</payments>

## Binære data

Et (ikke så typisk) eksempel på en enkelt rød prik der er base64 kodet.

<<minPixel>>SUkqABwAAACAP+BACCQJ/wQAO0OCSEQqGQSAgA8A/gAEAAEAAAAAAAAAAAEEAAEAAAACAAAAAQEE

AAEAAAAEAAAAAgEDAAMAAADWAAAAAwEDAAEAAAAFAAAABgEDAAEAAAACAAAAEQEEAAEAAAAIAAAA

FQEDAAEAAAADAAAAFgEEAAEAAAAEAAAAFwEEAAEAAAAUAAAAGgEFAAEAAADcAAAAGwEFAAEAAADk

AAAAHAEDAAEAAAABAAAAKAEDAAEAAAACAAAAPQEDAAEAAAACAAAAAAAAAAgACAAIAAB3AQDoAwAA

AHcBAOgDAAA=<</minPixel>>

# Anvendelse af data til bygning af skabeloner

I PrintNet kan der opsættes skabeloner på baggrund af data-strukturerne i det xml, printNet modtager.

Som tidligere beskrevet består dette xml af en header og en body del. Header delen er fastlagt og indeholder de berigede data fra A&D, mens body delen fastlægges frit af det enkelte fagsystem.

I nedenstående gennemgås et eksempel på, hvordan xml-filen kan benyttes til at opbygge en skabelon med en tabel som variabel data

PrintNets værktøjer og brugen af disse vil blive gennemgået på de fastlagte kurser, og detaljer kan findes i diverse bruger manualer.

Nærværende eksempel skal alene illustrere hvordan fagsystemets datamodel kan anvendes i opbygningen.

## Fagsystemets data

Fagsystemet fastlægger f.eks. en data-struktur bestående af en tabel Payments med tre kolonner: description, amount og dueDate:

<payments>

<payment>

<description>Cheese</description>

<amount>12</amount>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>Gum</description>

<amount>0.10</amount>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>Car</description>

<amount>1200234</amount>

<dueDate>2010-05-03</dueDate>

</payment>

</payments>

Denne data-struktur kan hentes ind i PrintNet T designer på flere måder, f.eks. kan man danne den fulde xml-struktur ud fra xsd-filen til PrintNet-interfacet og substituere body-delen med sin egen data-struktur.

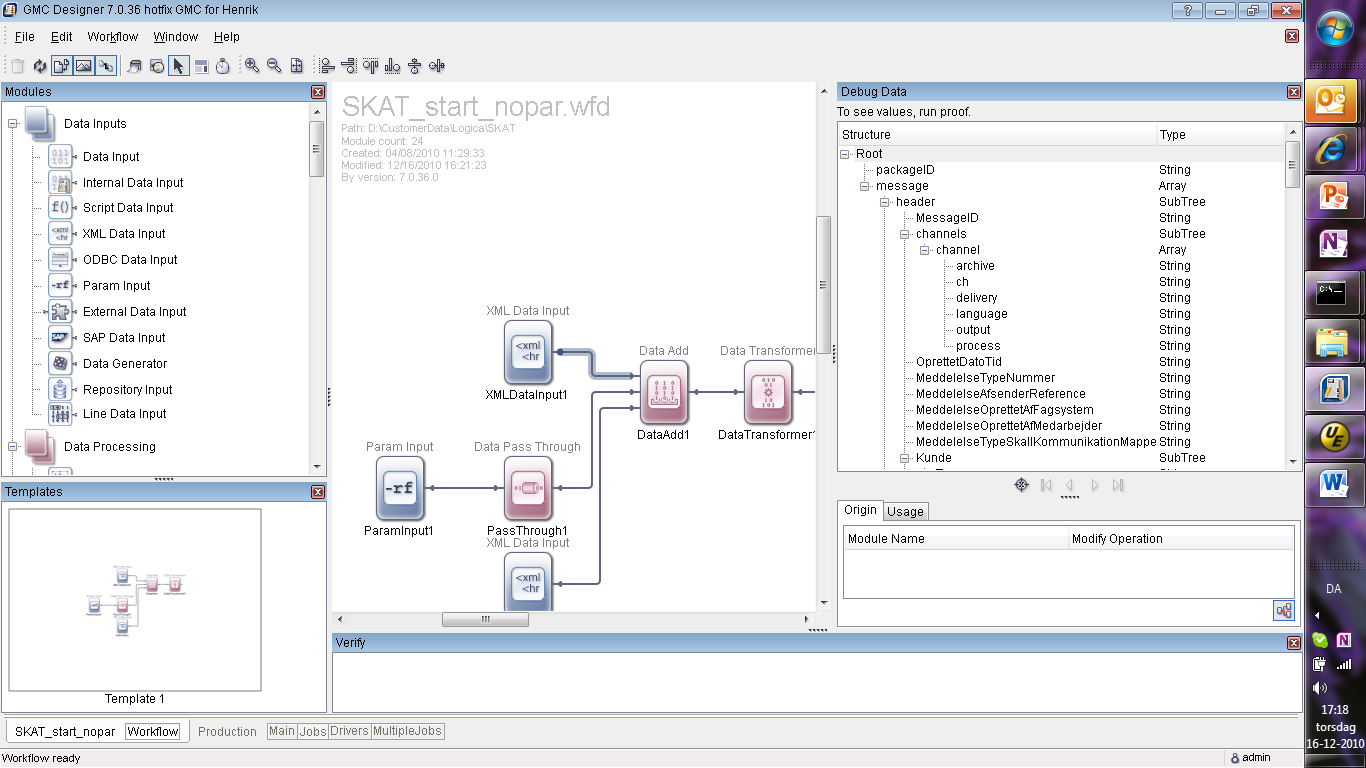
Eller man kan vælge at tage en kopi af en af de eksisterende skabeloner i PrintNet T Designer og så her substituere den eksisterende body-del med sin egen body-del.

Det vigtigste er:

1. Den variable del af datastrukturen (body-delen), som man anvender til opbygning af skabelon-sættet skal være identisk med den datastruktur, man har tænkt sig at sende med MeddelelseMultiSend (i indholds delen), se eksempel i afsnit 4.
2. Den variable del af datastrukturen (body-delen), som man anvender til opbygning af skabelonsættet bør have et repræsentativt indhold, dvs. hvis data i en tabel normalt vil være 2-5 linjer, så bør dette afspejles i xml’en, så skabelonbyggeren kan få et indtryk af, om data skal fylde mange sider, en enkelt linje etc.

## Data set i PrintNet T Designer XML Input

I PrintNet T designer kan datastrukturen hentes ind udefra via en xml-fil, eller opbygges på baggrund af en kopi af en eksisterende:



Figur PrintNet T Designer med xml-fil

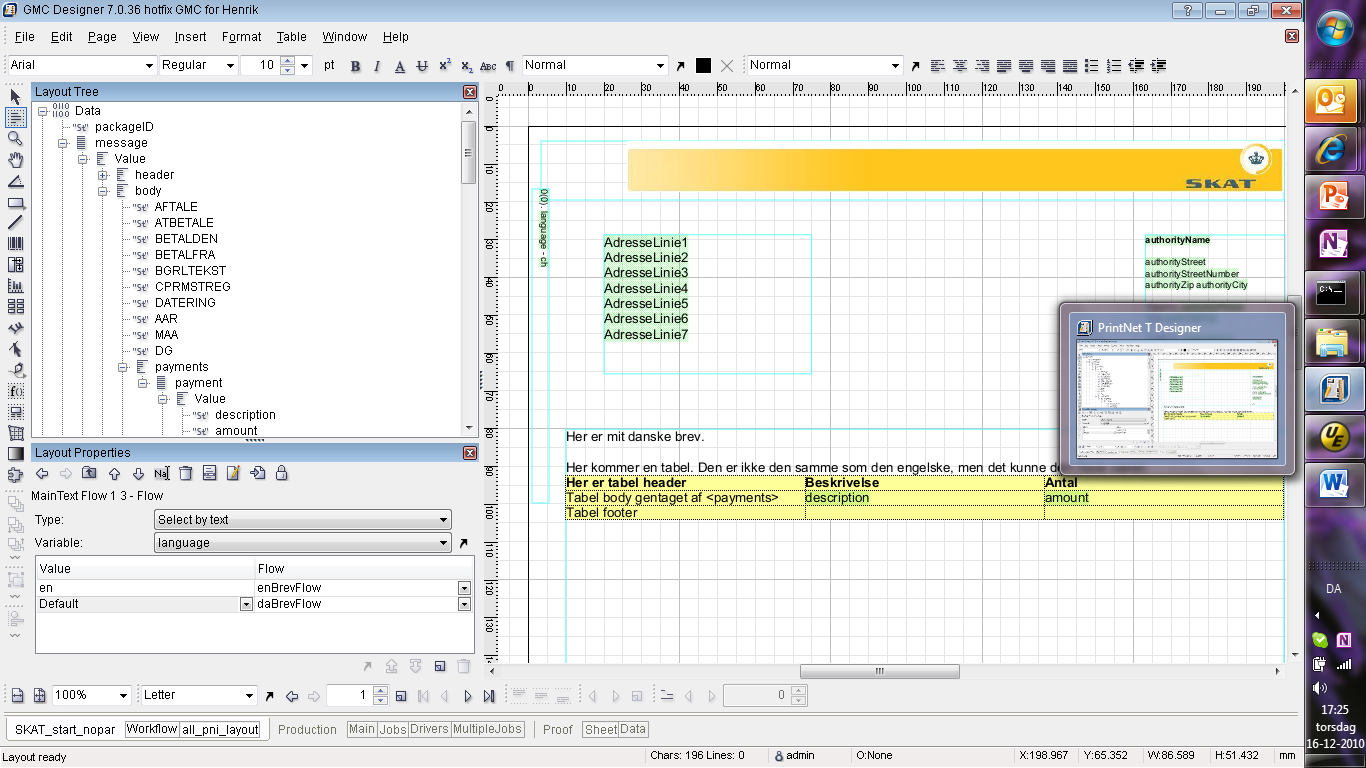
Data kan ses i højre vindue på ovenstående skærmbilledet. Her kan der tilføjes yderligere strukturer, elementer etc.

I ovenstående eksempel vises en del af headerstrukturen. Her må der naturligvis aldrig ændres i strukturen, da header delen skal overholde schemaet for PrintNet interfacet. Strukturer i Headeren må alene anvendes til at opbygge skabelonsættet.

Der må kun foretages rettelser i strukturen i body-delen i denne del af PrintNet T Designer. Samtidigt er der her mulighed for at lægge transformationer og regler ind, såsom at visse dataelementer ikke må være tomme e.l.

## Data set i PrintNet T Designer Layout

Når data-strukturen for body delen er fastlagt, så kan PrintNet T Designer anvendes til at opbygge selve skabelonen:



Figur PrintNet T Designer med layout af tabel

I venstre side af skærmbilledet ses data-strukturen, som anvendes til at opbygge layout’et.

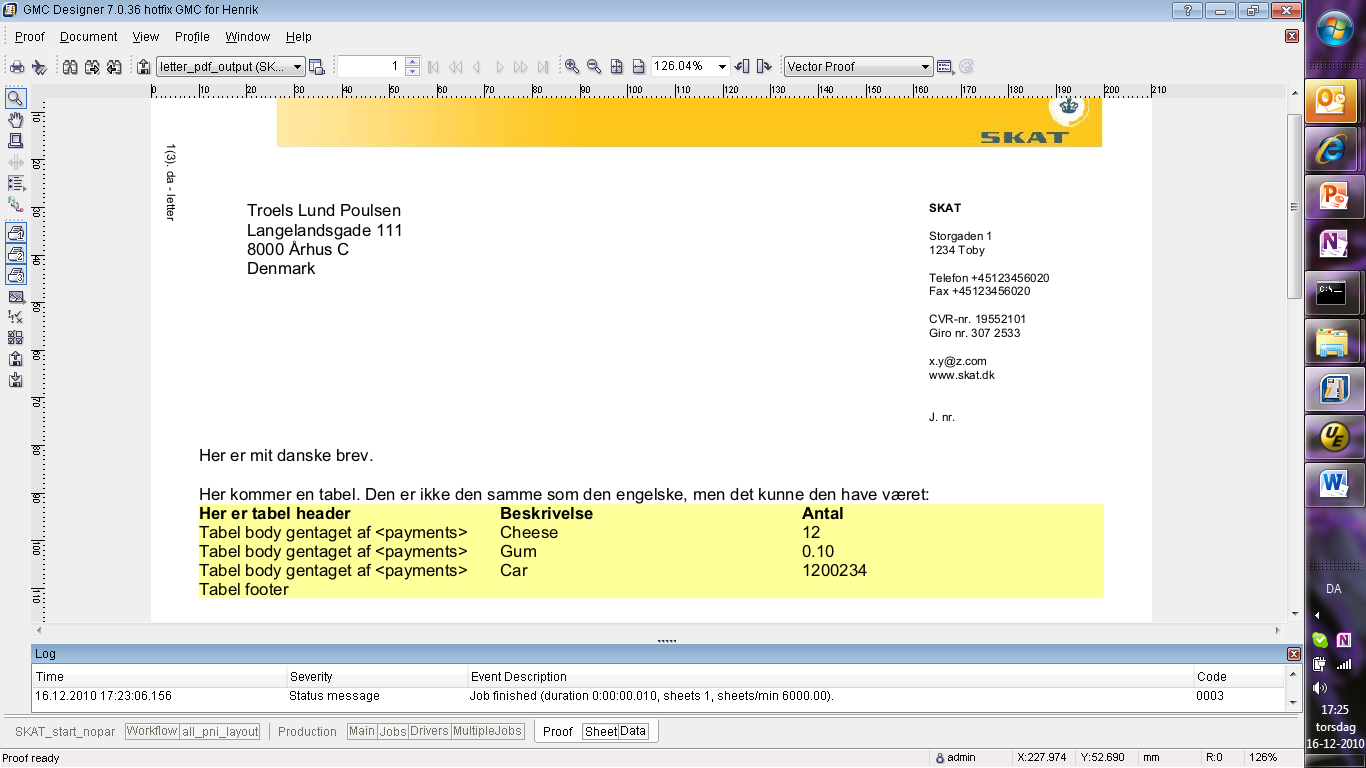
I ovenstående eksempel består data-strukturen af en body del med en række variable – samt eksemplets tabel-struktur: Payments.

Højre side viser anvendelsen af data-strukturerne:

* Øverste venstre del består af en række felter fra header-strukturen (kundens alternative adresse).
* Øverste højre side består af en række skabelon-variable, dvs. at det ikke er variable data for det enkelte brev, men fælles variable for alle breve for denne skabelon.
* Midt på siden ses layout for tabellen baseret på data-strukturen Payments.

## Data set i PrintNet T Designer Proof

Layout kan opbygges trinvis i PrintNet T Designer, og der kan genereres et proof, hvor det færdige brev kan ses. Nedenstående figur viser proof for det definerede layout i forrige afsnit.



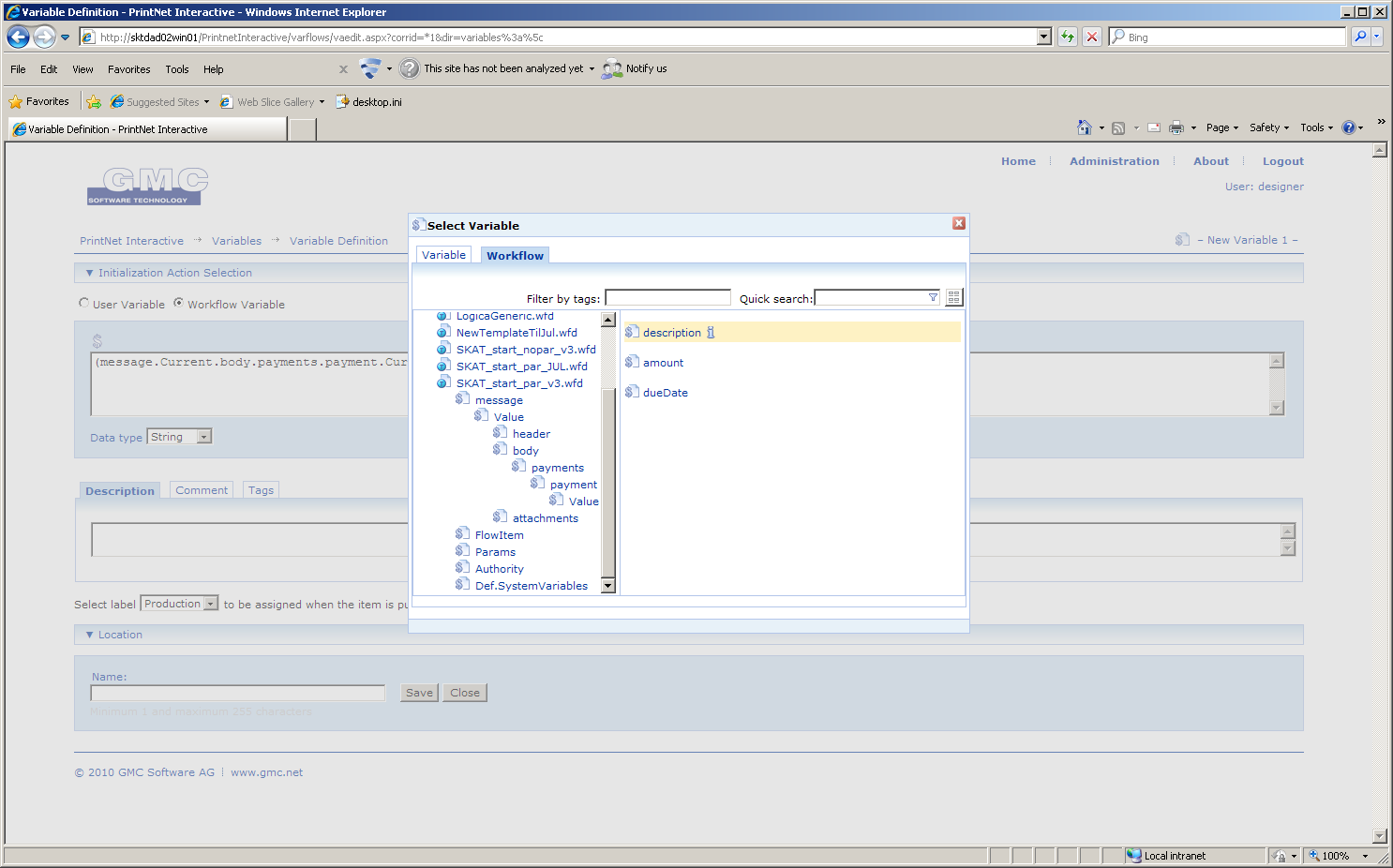
Figur 4 PrintNet T Designer Proof

Samme data set som Proof af et dansk brev. Nu ser man at alle linjer i tabellen er med.

## PrintNet Interactive

Den samme datastruktur er til rådighed i PrintNet Interactive. PrintNet Interactive er et lettere tilgængeligt værktøj til design af simple skabeloner, eller til mindre rettelser i en eksisterende skabelon.

Her kan alle tekster vedligeholdes og datastrukturen kan anvendes ved at oprette variable baseret på de elementer af datastrukturen, der skal anvendes.



Figur Datastruktur set i PrintNet Interactive

I skærmbilledet ses datastrukturen hvor Payment delen er fremsøgt.

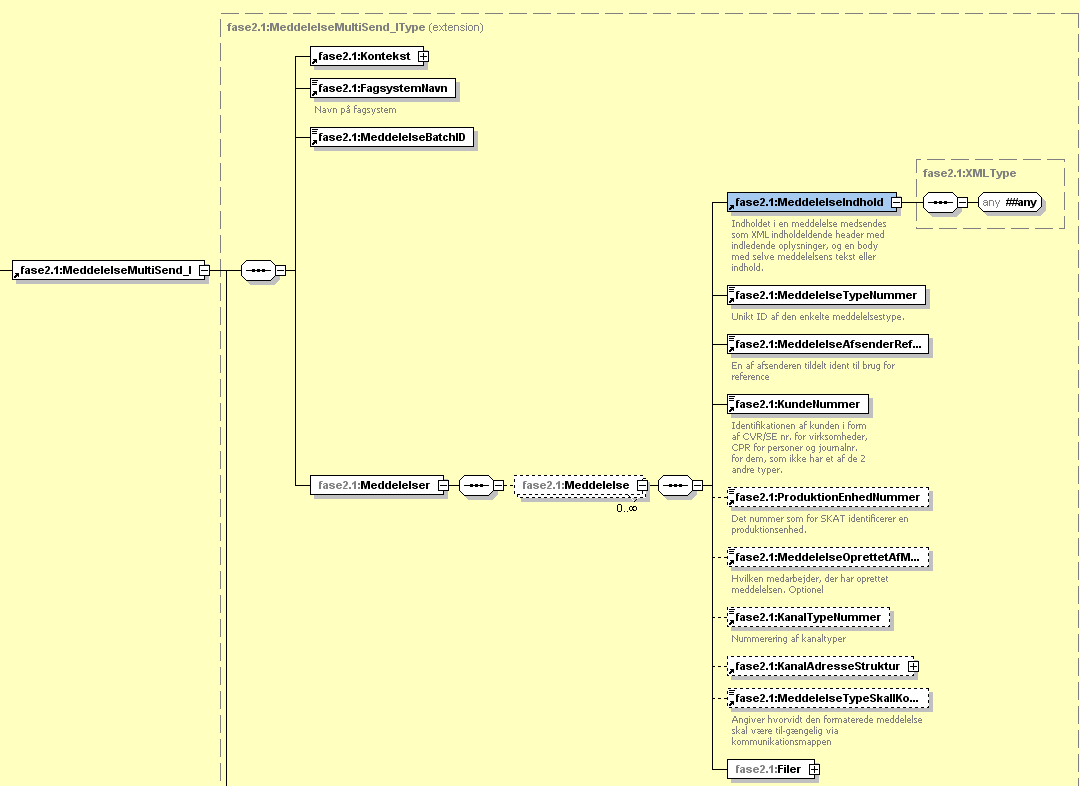
Herefter kan der oprettes en variabel, der udpeger denne del af datastrukturen – og denne variabel kan så anvendes i opbygning af nye paragraffer til skabeloner.

Dvs. at navngivningen i de datastrukturer fagsystemerne opbygger ikke er specielt vigtig i forhold til eventuelle senere lejlighedsvise brugere, der disse formentlig kun skal anvende PrintNet Interactive og datastrukturen her kan ”oversættes til” variable med navne der er egnede til disse skabelonbyggere.

# Anvendelse af data til kald af MeddelelseMultiSend

Når skabelonen er færdigopbygget og der er kørt proof, så man sikrer sig at alle dele af skabelonen og skabelonsættet, hvis der skal opbygges både e-mail, brev og sms, fungerer, så kan man anvende skabelonen til kald af MeddelelseMultiSend

Her skal den generelle xsd for kald af MeddelelseMultiSend naturligvis overholdes, og MeddelelseIndhold udfyldes med den variable datastruktur, skabelonen er opbygget på.



Figur MeddelelseMultiSend xsd

## Kald til MeddelelseMultiSend

I nedenstående eksempel sendes en Meddelelse med MeddelelseIndhold bestående af eksemplets datastruktur i form af en tabel med tre datasæt:

<ns:MeddelelseMultiSend\_I>

<ns:Kontekst>

</ns:Kontekst>

<ns:FagsystemNavn>DMR</ns:FagsystemNavn>

<ns:MeddelelseBatchID>105</ns:MeddelelseBatchID>

<ns:Meddelelser>

<ns:Meddelelse>

<ns:MeddelelseIndhold>

<Content>

<payments>

<payment>

<description>Cheese</description>

<amount>12</amount>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>Gum</description>

<amount>0.10</amount>

<dueDate>2010-05-03</dueDate>

</payment>

<payment>

<description>Car</description>

<amount>1200234</amount>

<dueDate>2010-05-03</dueDate>

</payment>

</payments>

</Content>

</ns:MeddelelseIndhold>

<ns:MeddelelseTypeNummer>80</ns:MeddelelseTypeNummer>

<ns:MeddelelseAfsenderReference>10211</ns:MeddelelseAfsenderReference>

<ns:KundeNummer>2008181818</ns:KundeNummer>

<ns:MeddelelseOprettetAfMedarbejder>user65</ns:MeddelelseOprettetAfMedarbejder> <ns:MeddelelseTypeSkalIKommunikationMappe>true</ns:MeddelelseTypeSkalIKommunikationMappe>

<ns:Filer>

</ns:Filer>

</ns:Meddelelse>

</ns:Meddelelser>

</ns:MeddelelseMultiSend\_I>

# Appendix A

Detaljeret indhold af Header-filen, dvs. data der er til rådighed for skabelon-opbyggeren.

