

KOMMUNIKATIONSFORSKRIFT FOR GASMARKEDET

Bilag 1 – Syntaks og struktur i EDI-meddelelser

Version: 5
Release: 1
Dato: 1. juni 2022
Ikrafttræden: 1. april 2023

Indholdsfortegnelse

Ændringshistorik.....	3
1. EDIFACT-syntaks og -struktur.....	4
1.1. EDIFACT-struktur og begreber	4
1.2. EDIFACT-syntaks.....	5
1.2.1. Tegnsæt og opbygning	5
1.2.2. Brug af reserverede tegn.....	5
1.2.3. Undlad overflødige tegn.....	5
1.3. Ikke beskrevne segmenter og elementer i meddelelsesguiderne.....	5
1.4. Forsendelsesreference, meddelelses-ID og forsendelsesversion	5
1.5. Faste segmenter	5
1.5.1. UNA – Syntakssegment	6
1.5.2. UNB – EDIFACT envelope for all documents except CONTRL	7
1.5.3. UNH EDIFACT document header	9
1.5.4. UNZ.....	10
2. XML-syntaks og -struktur	11
2.1. Hvad er XML?	11
2.2. XML-syntaks	11
2.2.1. Well-formness	12
2.2.2. Brug af attributter og tomme tags	12
2.2.3. DTD kontra XML-skema.....	12
2.2.4. Anvendelse af tegnsæt og karakterer	13
2.3. Forsendelsesreference, meddelelses-ID og forsendelsesversion	13
2.4. Core Components.....	13
2.5. Anvendelse og opbygning af XML-skemaer	13
2.6. Versionering for XML-skema	14

Ændringshistorik

Ændringshistorikken indeholder ændringer siden sidste version af forskriften. En ændring til versionen indikerer store ændringer til struktur eller syntaks, mens ændringer til release kun indeholder små justeringer.

Afsnit	Tidspunkt	Ændring
Alle	April 2007	Forrige version
Alle	Juni 2022	Flyttet til Evida dokument i forbindelse med overtagelse af ansvaret for forskriften og ordlyd ændret fra el til gas. Døde links fjernet.

1. EDIFACT-syntaks og -struktur

Denne bilagsrapport beskriver den tekniske syntaks og strukturen i EDIFACTstandard. Indholdet bygger på ebIX "Common rules and recommendations".

1.1. EDIFACT-struktur og begreber

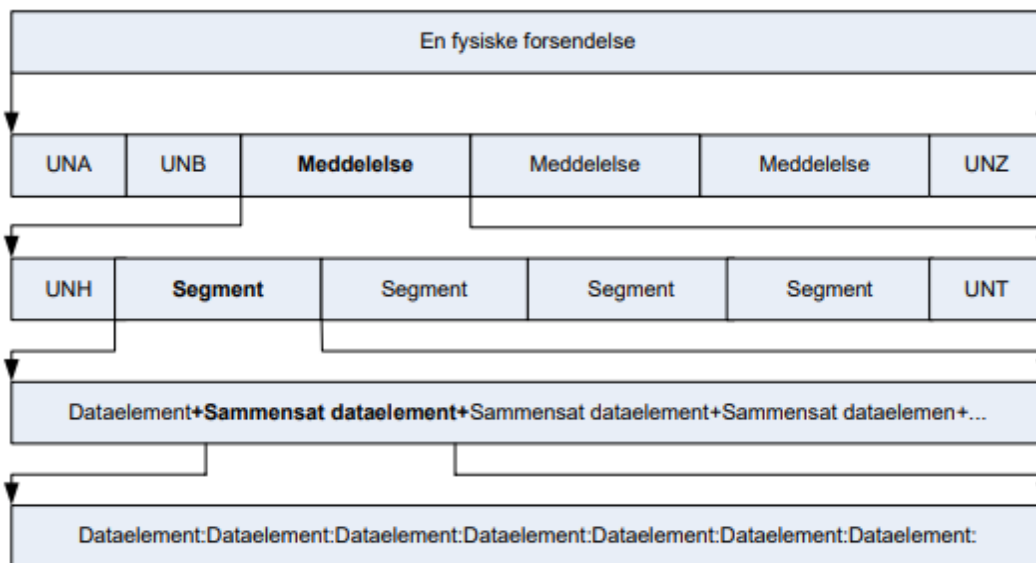
EDIFACT er en FN-standard (UN/CEFACT), der bruges til at flytte data i struktureret form. Det betyder, at EDIFACT er underlagt nogle syntaksregler, der gælder til enhver tid, og som altid skal overholdes.

EDIFACT indeholder følgende begreber:

- En **forsendelse** består af alle oplysninger i en forsendelse. En fysisk forsendelse kan bestå af én eller flere meddelelser. En forsendelse har én afsender og én modtager.
- En meddelelse er eksempelvis:
 - Målinger
 - Planer

I Danmark er det reglen, at der kun må sendes én meddelelse pr. forsendelse.

- Et **segment** er en gruppering af sammenhørende dataelementer/felter. Det kan f.eks. være en adresse. I segmentet udspecificeres adressen i navn, vej, postnummer, by, land osv.
- Et **sammensat dataelement** består af to eller flere dataelementer.
- Et **dataelement/felt** er de enkelte informationer, der sendes i et segment. Det kan f.eks. være en tidsangivelse eller en et antal kilowatt timer.



Ovenstående figur illustrerer de fire begreber anvendt i relation til hinanden. UNA, UNB og UNZ er segmenter, der bruges til at angive begyndelsen på og afslutningen af en forsendelse. UNH og UNT er segmenter, der bruges til at angive begyndelsen og afslutningen på en meddelelse.

Følgende tegn anvendes til at markere adskillelse, afslutning mv. i forbindelse med dataelementer og segmenter:

- + Markerer adskillelse mellem dataelementer og mellem segmenter.
- ' Markerer, at et segment er færdigt.
- : Markerer adskillelse mellem de enkelte datafelter i sammensatte dataelementer; det vil sige flere felter, der beskriver det samme.

1.2. EDIFACT-syntaks

Grundregler for meddelelser anvendt i gasmarkedet.

1.2.1. Tegnsæt og opbygning

- EDIFACT skal sendes som en lang linje (streng) uden linjeskift.
- Forsendelser skal altid sendes i tegnsættet ISO 8859-1, også kaldet UNOC
- I UNB'en skal yymmdd og hhmm udfyldes fuldt ud. Det vil sige, at man ikke må nøjes med at sende f.eks. 645 for at angive klokkeslættet. Det korrekte er 0645.
- Meddelelsesnavne og segmentnavne skrives altid med STORT, når de transmitteres.
- EDIFACT version 3 anvendes i Danmark.

1.2.2. Brug af reserverede tegn

Segmentet UNA beskriver de skilletegn (reserverede tegn), der benyttes i den fremsendte EDIFACT. Standarden er: UNA:+.?'

Hvis man bruger et af de reserverede tegn (+, ;, '), skal man sætte et spørgsmålstegn (?) foran. F.eks. skal addition af to tal (f.eks. 10 + 20) angives som 10?+20. Hvis man skal bruge et spørgsmålstegn, skal man skrive to spørgsmålstegn (??). Det første spørgsmålstegn betyder, at det næste har sin normale betydning. Bruger man en sådan 'release'-karakter, tæller den ikke med i feltlængden. Hvis et felt rummer mulighed for 20 alfanumeriske karakterer, og der er brug for 2 'release'-karakterer, kan feltet fylde op til 22 karakterer.

1.2.3. Undlad overflødige tegn

- Efterstillede plusser (+) i hele segmentet slettes.
- Efterstillede koloner i hvert enkelt data slettes. Foranstillede plusser (+) eller koloner (:) slettes ikke.
- Dataelementer (felter) kan springes over ved at udelade felternes 'værdi'. F.eks. 24500+45++25600, hvor feltværdien mellem 45 og 25600 er udeladt. Hvis der er tale om et sammensat dataelement, kan det f.eks. blive 3650:3575::3400, hvor feltværdien mellem 3575 og 3400 er udeladt.

1.3. Ikke beskrevne segmenter og elementer i meddelelsesguiderne

EDIFACT-meddelelser skal leve op til alle beskrevne regler i forskriften. Såfremt en aktør modtager EDIFACT-meddelelser, der indeholder yderligere segmenter og elementer, der ikke står beskrevet i de gældende guidelines, skal disse ignoreres. Det må således ikke give anledning til en fejlmeddelelse til afsender, så længe meddelelsen overholder den grundlæggende struktur for den pågældende EDIFACT-meddelelse (f.eks. UTILMD eller MSCONS).

1.4. Forsendelsesreference, meddelelses-ID og forsendelsesversion

Forsendelsesreferencen (interchange control reference) findes i UNBsegmentets element S004.0020, som skal være unikt over tid for afsender defineret i S002. Hvis det ikke er tilfældet, skal den sidst modtagne meddelelse automatisk afvises. Meddelelses-ID (message ID) findes i BGM-segmentets element C002.1004 og anvendes til unik identifikation af en meddelelse. Meddelelses-ID skal være unik over tid for hver aktør. Forsendelsesversion benyttes ikke i EDIFACT.

1.5. Faste segmenter

Herunder beskrives udvalgte service-segmenter (UN-segmenter). UNA og UNB indleder en forsendelse og kan betragtes som forsendelsens konvolut. UNB anvendes af det modtagende EDI-system til genkendelse af afsender og modtager.

1.5.1. UNA – Syntakssegment

UNA Service String advice
Function: Til angivelse af notation i EDIFACT-udvekslingen
Classification: Conditional, (1)
Comments: Nedenstående er default. Anvendes i Danmark
 UNA UNA:+.? '

Ref.	Name	Cl.	Form.	Description
	SAMMENSAT DATAELEMENT, SEPARATOR	M	an1	":" (kolon)
	DATAELEMENT, SEPARATOR	M	an1	"+" (Plus)
	DECIMAL NOTATION	M	an1	"." (punktum)
	OPHÆVELSESTEGN	M	an1	"?" (spørgsmålstegn)
	RESERVERET FREMTIDIG BRUG	M	an1	" " (blank)
	SEGMENT TERMINATOR	M	an1	"'" (apostrof)

1.5.2. UNB – EDIFACT envelope for all documents except CONTRL

UNB Interchange Header
Function: Til identifikation og adressering af forsendelsen, angivelse af kvitteringsanmodning, samt testindikation.
Classification: Mandatory (M1).
n:
Comments: Brugen af UNB-segmentet skal aftales i en forsendelsesaftale.
Example: UNB+UNOC:3+5790000395620:14+5790000432752:14+070120:1606+31929++DK-TIS-MET++1+DK

Ref.	Name	CI	Form.	Description
S001	SYNTAX IDENTIFIER	M	.	
0001	Syntax identifier	M	a4	Kode: UNOC (ISO 8859-1)
0002	Syntax version number	M	n1	Kode: 3 Version 3 af EDIFACT-syntax Skal anvendes hvis Syntax identifier er "UNOC"
S002	INTERCHANGE SENDER	M		
0004	Sender identification	M	an..35	GLN- eller EIC-nummer.
0007	Partner identification code qualifier	R	an..4	Kode: 14 GLN (Global Location Number) 305 EIC (European Identification Code)
0008	Address for reverse routing	O	an..14	Kun hvis bilateralt aftalt.
S003	INTERCHANGE RECIPIENT	M		
0010	Recipient identification	M	an..35	GLN eller EIC nummer.
0007	Partner identification code qualifier	R	an..4	Kode: 14 GLN (Global Location Number) 305 EIC (European Identification Code)
0014	Routing address	O	an..14	Kun hvis bilateralt aftalt.
S004	DATE AND TIME OF PREPARATION	M		
0017	Date	M	n6	Dato for dannelse af forsendelsen (YYMMDD)
0019	Time	M	n4	Klokkeslæt for dannelse af forsendelsen (TTMM)

0020	INTERCHANGE CONTROL REFERENCE	M	an..14	Entydig forsendelsesreference. Skal matche med dataelement 0020 i UNZ Skal være unik over tid for afsender defineret i element S002.0004 ellers skal forsen- delsen afvises.
S005	RECIPIENTS REFERENCE, PASSWORD	X		
0022	Recipient's reference/ password	X	an..14	
0025	Recipient's reference/ password qualifier	X	an..2	
0026	APPLICATION REFERENCE	O	an..14	BPI anvendes jf. nationale regler eller "ebIX Business information models".
0029	PROCESSING PRIORITY CODE	X	a1	
0031	ACKNOWLEDGEMENT REQUEST	O	n1	Kode: 1 Kvittering sendes blank Ingen kvittering
0032	COMMUNICATION AGREEMENT ID	O	an..35	Kode: DK Danske kommunikations bestemmelser.
0035	TEST INDICATOR	D	n1	Kode: 1 Anvendes hvis forsendel- sen er en test. Skal bila- teralt aftales. blank Meddelelsen er i pro- duktion.

1.5.3. UNH EDIFACT document header

UNH-segmentet er beskrevet, idet dansk praksis for elementet 0068 afviger fra ebIX.

UNH Message header
Function: UNH sendes en gang pr. dokument i en forsendelse og fortæller, hvilken meddelelse der sendes, versionsnummer og kode for udvekslingssamarbejde.
Classification: Mandatory (M1).
Comments:
Example: UNH+1+UTILMD:D:02B:UN:E5DK02+DK-BT-001-002'

Ref.	Name	CI	Form.	Description
		.		
0062	MESSAGE REFERENCE NUMBER	M	an..14	Angiver en entydig reference for meddelelsen.
S009	MESSAGE IDENTIFIER	M		
0065	Message type identifier	M	an..6	Angivelse af den anvendte meddelelse (f.eks. UTILMD)
0052	Message type version number	M	an..3	
0054	Message type release number	M	an..3	EDIFACT meddelelsesversion.
0051	Controlling agency	M	an..2	Organisation meddelelsen er underlagt.
0057	Association assigned code	C	an..6	MIG versions nummer. Se MIG. Skal anvendes.
0068	COMMON ACCESS REFERENCE	C	an..35	Business Combined id
S010	STATUS OF THE TRANSFER	C		
0070	Sequence message transfer number	M	n..2	
0073	First/last seq. mess. transfer. Indicator.	C	a1	

1.5.4. UNZ

UNZ Interchange Trailer
Function: Til afslutning af forsendelse, samt angivelse af total antal meddelelser i forsendelse
Classification: Mandatory (M1).
Comments: Dataelement 0020 skal være identisk med 0020 i UNB
Example: UNZ+1+358765298'

Ref.	Name	CI	Form.	Description
0036	INTERCHANGE CONTROL COUNT	M	n..6	Antal meddelelser i forsendelse
0020	INTERCHANGE CONTROL REFERENCE	M	an..14	Skal matche med dataelement 0020 i UNB-segmentet

2. XML-syntaks og -struktur

Afsnittet er en hjælp til aktører i det danske gasmarked, der skal anvende XMLmeddelelser. Som følge af, at XML ikke er en standard, gør man i gasmarkedet brug af rammeværk, der består af en række grundkomponenter, som meddelelserne opbygges af. De enkelte meddelelser er ikke beskrevet på samme måde som EDIFACT-meddelelser.

2.1. Hvad er XML?

XML er et sprog, der benyttes til at beskrive data på en struktureret og generel måde. Data beskrives i XML ved brug af ren tekst, der opdeles i en træstruktur. Træstrukturen angives ved brug af tags, der er en opmærkning af data, som vist i det følgende eksempel:

```
<person>
  <navn>Per</navn>
  <postnr>5000</post>
</person>
```

Når data opmærkes, giver det mulighed for struktureret udveksling mellem to eller flere partner. I XML står det enhver frit at definere en vilkårlig opmærkning. Derfor er det i de fleste sammenhænge nødvendigt at definere en standardiseret opmærkning til brug for dataudvekslingen.

XML-forsendelsen i sig selv er ren tekst lagret i en tekstfil. Der skal derfor også tages stilling til hvilken metode, der benyttes til transport af forsendelsen frem til modtageren. Dette kan gøres f.eks. via en webservice eller per e-mail.

2.2. XML-syntaks

XML (eXtensible Markup Language) består af **tags**, der omslutter og derved opmarkerer typen af data. Et postnummer kunne for eksempel være markeret med en start-tag og slut-tag (se eksemplet herunder).

```
<postnr>5000</postnr>
```

I ovenstående eksempel giver værdien 5000 nu mening, idet det er defineret som et postnummer. Alternativt kunne værdien symbolisere hvad som helst.

Start-tag er defineret ved: <tag-navn> Slut-tag er defineret ved: </tag-navn>

Tilsammen definerer start-tag, slut-tag og indholdet mellem disse et element. I ovenstående eksempel definerer tag'en "postnr" således elementet.

En tag kan også indeholde attributter, der giver yderligere information og dermed værdi i forhold til indholdet af tag'en, udover navnet. Attributter erklæres inde i start-tag'en med navn som vist i eksemplet herunder.

```
<postnr landekode="DK">5000</postnr>
```

I ovenstående eksempel giver attributten **landekode="DK"** eksempelvis information om, at indholdet af tag'en (5000) er et dansk postnummer.

Attributter defineres i start-taq'en ved: attributnavn="attributværdi"

Alternativet kunne være en efterfølgende tag for landekoden (se eksemplet herunder)

```
<postnr>5000</postnr>
<landekode>DK</landekode>
```

2.2.1. Well-formness

Hvis et XML-dokument skal være velformet (well-formness), skal dokumentet overholde W3C's XML-specifikation på den måde, at der er nøjagtigt ét rodelement. Alle elementer skal omslutes af en start-tag og en slut-tag. Hvis elementet er tomt, skal der anvendes tags som beskrevet i afsnit 2.2.2. Se et eksempel på anvendelse af tags herunder.

```
<rod-element>
  <navn>Per</navn>
  <postnr>5000</post>
</rod-element>
```

Ovenstående eksempel er well-formed efter den beskrevne specifikation.

2.2.2. Brug af attributter og tomme tags

En tag kan indeholde et vilkårligt antal forskellige attributter.

En attribut defineres ved: `<tag-navn attributnavn="attributværdi">`

En tag behøver ikke nødvendigvis at indeholde data. I så fald er der tale om en tom tag. En tom tag kan defineres ved enten en start- og slut-tag uden noget indhold imellem eller som vist i eksemplet herunder.

```
<postnr/>
```

En tom tag defineres ved enten: `<tag-navn></tag-navn>`
eller ved: `<tag-navn/>`

2.2.3. DTD kontra XML-skema

Document Type Definition (DTD) og XML Schema Definition (XSD) er to mulige måder, hvorpå man kan definere indhold, struktur og type-definitioner for et XML-dokument. DTD har tidligere været ret udbredt i forbindelse med definition af XML-beskedens indhold direkte i XML-dokumentet. I dag går tendensen dog mod separate skema-filer (XSD) til definition af XML-dokumentets indhold. XSD gør det nemt at:

- beskrive muligt indhold
- validere korrektheden af data
- definere datafacetter (restriktioner for dataindhold)
- definere datamønstre (dataformater)
- konvertere data mellem forskellige datatyper (ved brug af redefine)

Desuden er udvikling og validering af XSD understøttet af en række XMLeditorer, parsere og andre værktøjer.

Valideringen af et XML-dokument sker i forhold til datadefinitionen (enten DTD eller XSD).

2.2.4. Anvendelse af tegnsæt og karakterer

I den standardiserede XML-standard anvendes UTF-8 tegnsættet, der er bagudkompatibelt med ASCII-tegnsættet. UTF-8 er et 2 bytes pr. tegn-tegnsæt, der er en komprimeret version af Unicode. Encoding angives i starten af XML-filen på følgende måde:

```
<?xml version="1.0" encoding="UTF-8"?>
```

XML-syntaksen indeholder en række reserverede tegn, der ikke må optræde i indholdet i elementer eller attributter:

Reserveret tegn	Alternativ notation
<	kan skrives som "<";
>	kan skrives som ">";
&	kan skrives som "&";
"	kan skrives som """;
'	kan skrives som "'";

XML-notationen er case-sensitiv (gælder ikke indholdet i et element).

Elementet `<Identifikation></ Identifikation>` er ikke det samme som `<identifikation></ identifikation>` på grund af forskellen på det store og lille I/i.

2.3. Forsendelsesreference, meddelelses-ID og forsendelsesversion

En XML-meddelelse har ikke en forsendelsesreference, men et unikt meddelelses-ID, der opnås ved kombination af elementerne "MessageIdentification" og "MessageVersion". Versionsnummeret er fortløbende startende med 1. Er placeret i elementet "MessageVersion".

2.4. Core Components

Core Components er en række kernekomponenter defineret af UN/CEFACT (United Nations Centre for Trade facilitation and Electronic Business). Core Components er syntaksneutrale og teknologineutale i deres grundform og kan bruges til datamodellering. CCTS (Core Component Technical Specification) sikrer en høj grad af genbrug på tværs af meddelelser og forbedrer interoperabiliteten imellem it-systemer.

Formålet med disse komponenter er, at de er så generiske, at de kan anvendes i en række forskellige meddelelser. Det er komponenter, der eksempelvis definerer rolle-begrebet, dato-tidsstempler og beskedtyper. Der er tale om komponenter, der flere gange anvendes i sammenhæng med forskellige beskedtyper.

Hvor det er muligt, bør disse kernekomponenter anvendes, ligesom der bør tænkes på kandidater til kernekomponenter, når der udvikles nye beskedtyper.

2.5. Anvendelse og opbygning af XML-skemaer

XML-skemaer anvendes til at validere XML-meddelelsen ved afsendelse og modtagelse. Hvis XML-meddelelsen ikke stemmer overens med skemaets definitioner, er meddelelsen ikke valid og bør derfor afvises. Dette sker med en Acknowledgement document.

XML-skemaer skal struktureres således, at der er stærk binding internt i skemaet og mulighed for genbrug. Ved stærk binding forstås, at et element i et skema skal have en relation til de resterende elementer i skemaet. Hvis der er elementer, der kan genbruges i andre skemaer, skal disse dog som hovedregel udtrækkes i separate skemaer (generalisering).

Hvert XML-skema skal struktureres således, at child-notes indlejres i parentnotes, hvorved der kun findes ét hoved-element defineret i hvert skema (se eksemplet herunder).

```
<element name="Adresse" type="AdresseType"/>
<complexType name="AdresseType">
  <xsequence>
    <element name="Vejnavn" type="string"/>
    <element name="Nummer" type="integer"/>
    <element name="Postnummer" type="PostnummerType"/>
  </sequence>
  <attribute name="id" type="string"/>
  <attribute name="version" type="string"/>
</xs:complexType>

<simpleType name="PostnummerType">
  <restriction base="integer">
    <pattern value="\d{4}"/>
  </restriction>
</simpleType>
```

2.6. Versionering for XML-skema

XML-skemaer, der er udviklet til kommunikation mellem Energinet.dk og deres eksterne parter, anvender et target namespace, der er opbygget som følger:

`"http://www.energinet.dk/schemas/" + <subnamespace> + "/" + <document> + "/" + <version> .`

Nedenstående eksempel viser, hvordan navngivning af et namespace kan se ud for XML-skemaet vedrørende balanceansvarlige aktørers udveksling af aktørplaner:

`http://www.energinet.dk/schemas/BalResp/XML/MarketScheduleDocument/v2`

XML-skemaerne versioneres ved brug af filnavnet. Det er en opbygning af navnet på XML-skemaets rodelement kombineret med versionsnummer. Kombinationen af de to dele adskilles af - (bindestreg), som vist herunder:

`<rodelementnavn> + "-" + <version> + ".xsd"`

Nedenstående eksempel viser navngivningen af første version af et XML-skema, hvor rodelementet er navngivet "MarketScheduleDocument":

MarketScheduleDocument-1.xsd

Attributten "version" i schema-elementet skal udover versionsnummer også afspejle udgivelsesnummer adskilt af punktum. Følgende eksempel gælder for version 2, udgivelse 4:

version="2.4"

Ændring i versionsnummeret vil skyldes strukturændringer i skemaet. Strukturændringer kan være tilføjelse eller fjernelse af elementer, navneændringer af elementer eller attributter eller ændringer i strukturen for elementerne. Ændring i udgivelsesnummeret vil skyldes mindre ændringer. Mindre ændringer kan være tilføjelse af valgfri elementer, ændringer i regler for attributindhold (så længe det ikke indskrænker) og lignende. Det vil sige, at der ikke fjernes elementer fra strukturen.

Det skal således være muligt at anvende flere forskellige udgivelser af en version af et XML-skema, uden at de kommer i konflikt med hinanden (det vil sige, at de skal være bagudkompatible). En tidligere version af et XML-skema vil dog komme

i konflikt med en nyere. Energinet.dk sørger for løbende at kunne håndtere den seneste frigivne version samt dens forgænger. Der er således altid to forskellige versioner til rådighed, hvor der kan findes flere udgivelser inden for hver version.