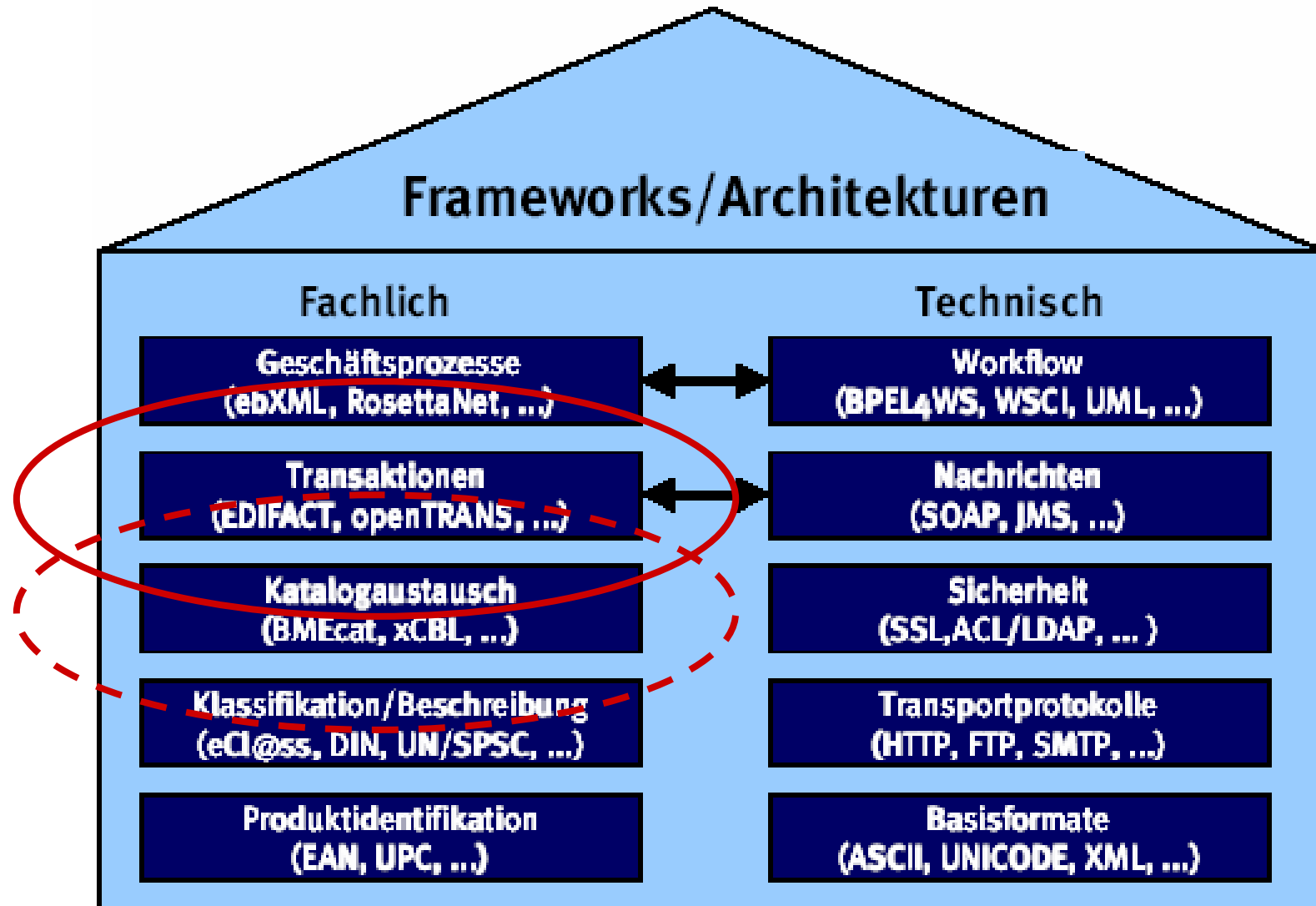




7437 – EDI und E-Business Standards, 4661 – E-Business: Standards und Automatisierung

Einführung in UN/EDIFACT
UN/EDIFACT *bottom-up*
Subsets, EANCOM
IAs und MIGs



Berlecon Research 2003



Einführung in UN/EDIFACT

UN/EDIFACT:

United **N**ations

Electronic **D**ata **I**nterchange **f**or
Administration, **C**ommerce and **T**rade



Vorschau: Unser Ziel!



```
UNB+UNOB:3+2100000000005:14+2900000000001:14+02  
0502:2350+1234'UNH+H123+ORDERS:D:96A:UN:EAN008'  
BGM+220+1234567'DTM+137:20020415:102'DTM+2:20020  
420:102'NAD+BY+2100000000005::9++XY~Handels~AG'N  
AD+SU+2900000000001::9++ABC~GmbH'LIN+1++2900000  
100008:EN'QTY+21:20'LIN+2++2900000200005:EN'QTY+2  
1:10:PCE'UNS+S'CNT+1:30'UNT+13+H123'UNZ+1+1234'
```

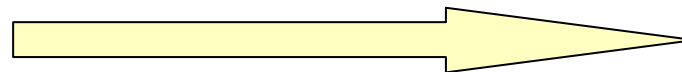
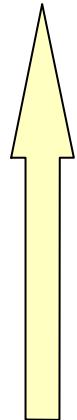
(Blanks durch ~ ersetzt)



Der Weg zu UN/EDIFACT



	Proprietär	Branchenspezifisch	Branchen- übergreifend
International		EANCOM (subset): Handel+ S.W.I.F.T: Banks	UN/EDIFACT EANCOM (subset)
Regional		ODETTE (Auto, EU) RINET (Versicherung, EU)	ASC X.12
National		VDA (Auto, DE) SEDAS (Handel, DE / AT) GENCOD (Handel, FR)	TRADACOMS (UK)
Bilateral	BAV (Siemens) VW Formate		





- Unabhängig von ...
 - ... Sprache
 - Codes statt Klartext
 - ... Branche
 - modularer und generischer Nachrichtenaufbau
 - Viele optionale Segmente und Datenelemente
 - ... Land
 - International standardisiert unter Aufsicht der Vereinten Nationen
 - ... IT-Systemen
 - Spezifikationen sind völlig unabhängig von Plattformen und Anwendungen



Ausgewählte Nachrichtentypen



Name	EANCOM	Description
PARTIN	4	Party information
PRICAT	5	Price/sales catalogue data
PRODAT	1	Product data
ORDERS	7	Purchase order
ORDRSP	4	Purchase order response
ORDCHG	4	Purchase order change request
INVOIC	7	Invoice
DESADV	4	Despatch advice
REMAADV	2	Remittance advice
IFTMIN	1	Transport instruction
IFTSTA	1	Transport status
REQOTE	1	Request for quotation
QUOTES	1	Quotation
INVRPT	2	Inventory report
SLSFCT	2	Sales forecast report
SLSRPT	3	Sales data report



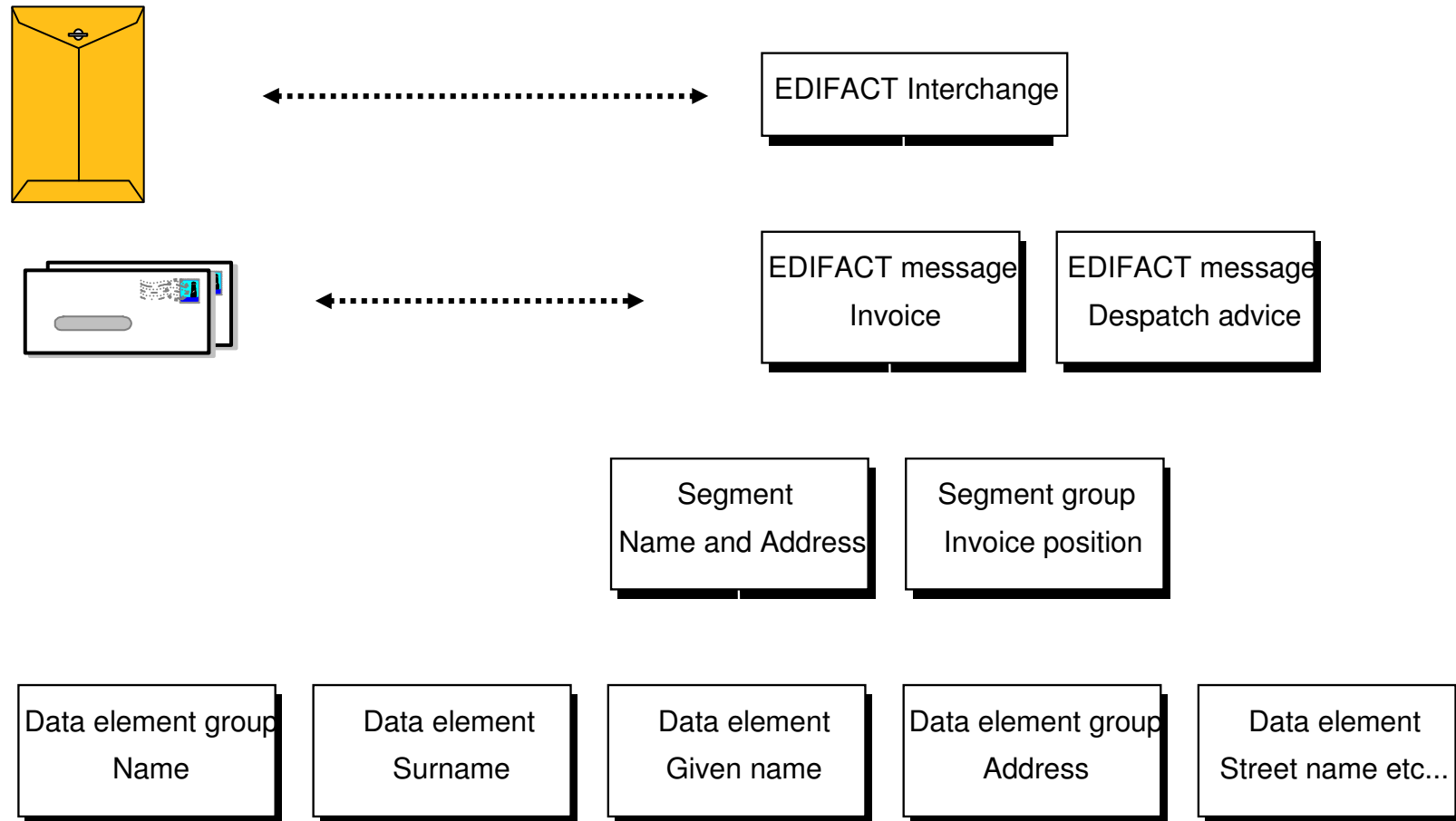
In use



Planned or in progress



UN/EDIFACT data structure





- Literaturangaben

- Thorsten Georg, EDIFACT: Ein Implementierungskonzept für mittelständische Unternehmen. DUV, Wiesbaden, 1993
- Thomas Schmoll, Handelsverkehr, elektronisch, weltweit: Nachrichtenaustausch mit UN/EDIFACT. Markt und Technik, Haar bei München, 1994
- Rainer Scheckenbach, Semantische Geschäftsprozeßintegration, DUV, Wiesbaden, 1997



- *Web links*
 - <http://www.unece.org/trade/untdid/welcome.html>,
 - <http://www.unece.org/etrades/welcome.htm>
 - Verteilung der Directories (UNTDID)
 - <http://www.gefeg.com/jswg/>
 - Joint Syntax Working Group
 - <http://www.gefeg.com/>
 - M. Dill von GEFEG ist offizieller deutscher Vertreter des DAT (UN Directory Audit Team)
 - <http://www.gli.de/>
 - Herstellerseite mit gut aufbereiteten Links zu Original-Doku
 - <http://www.edifactory.de/>
 - Hilfen, Software, Beispiele, lesbare UN/EDIFACT Directories



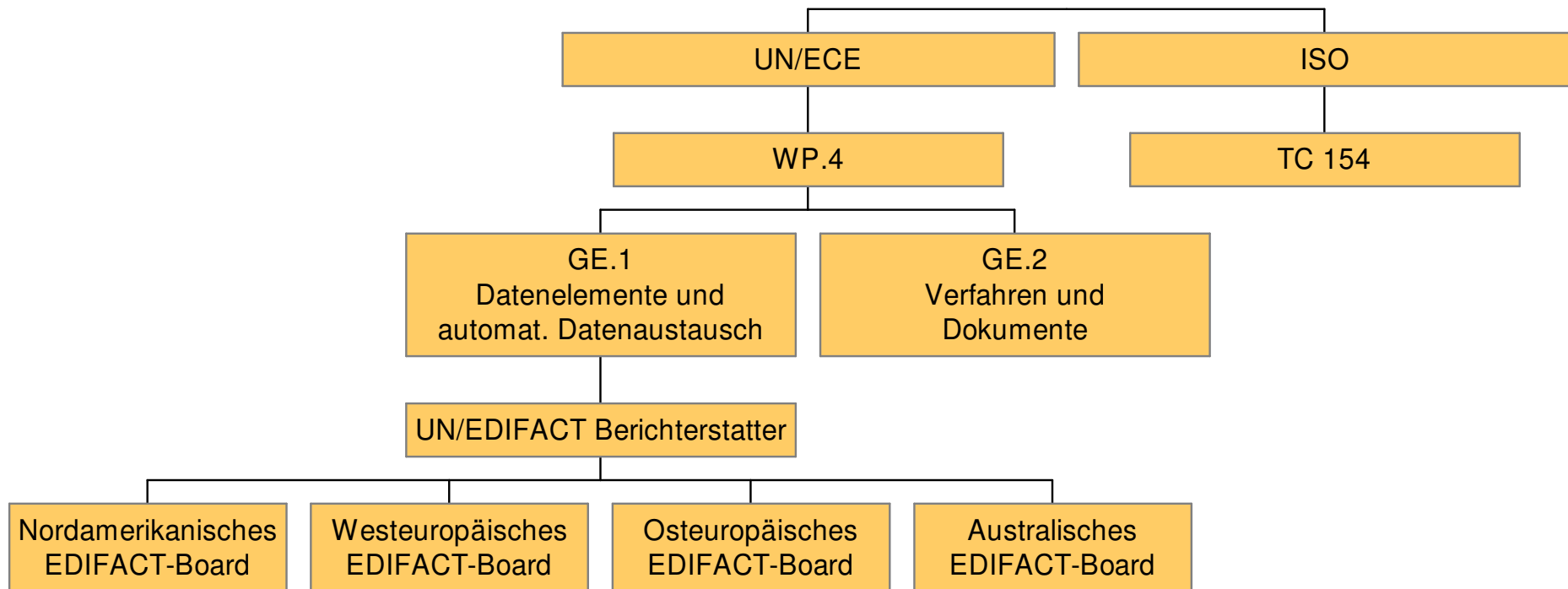
- **1947: Gründung der UN/ECE**
(Wirtschaftskommission für Europa)
- **1956: Beitritt der BR Deutschland zur UN/ECE**
- **1963: UN-Layout Key**
 - Basis für die Angleichung jeglicher Handels-, Zoll- und Transportdokumente
- **197x: Gründung der UN/ECE WP.4 (*working party 4*)**
 - Erleichterungen von Verfahren im internationalen Handel
 - Entwicklung von einheitlichen Datenelementen, Codes, Syntax-Regeln und Nachrichtentypen
 - Ziel: Automatisierung des Handelsdatenaustauschs

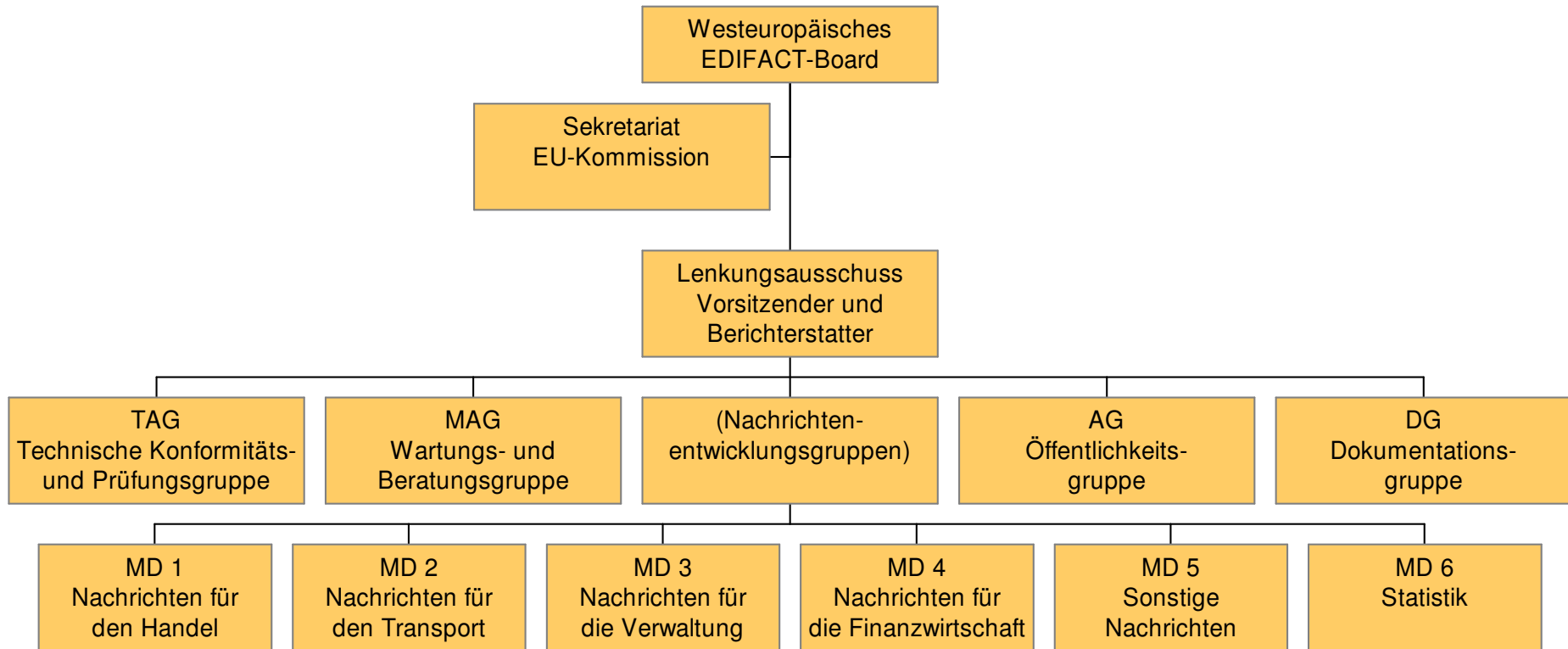


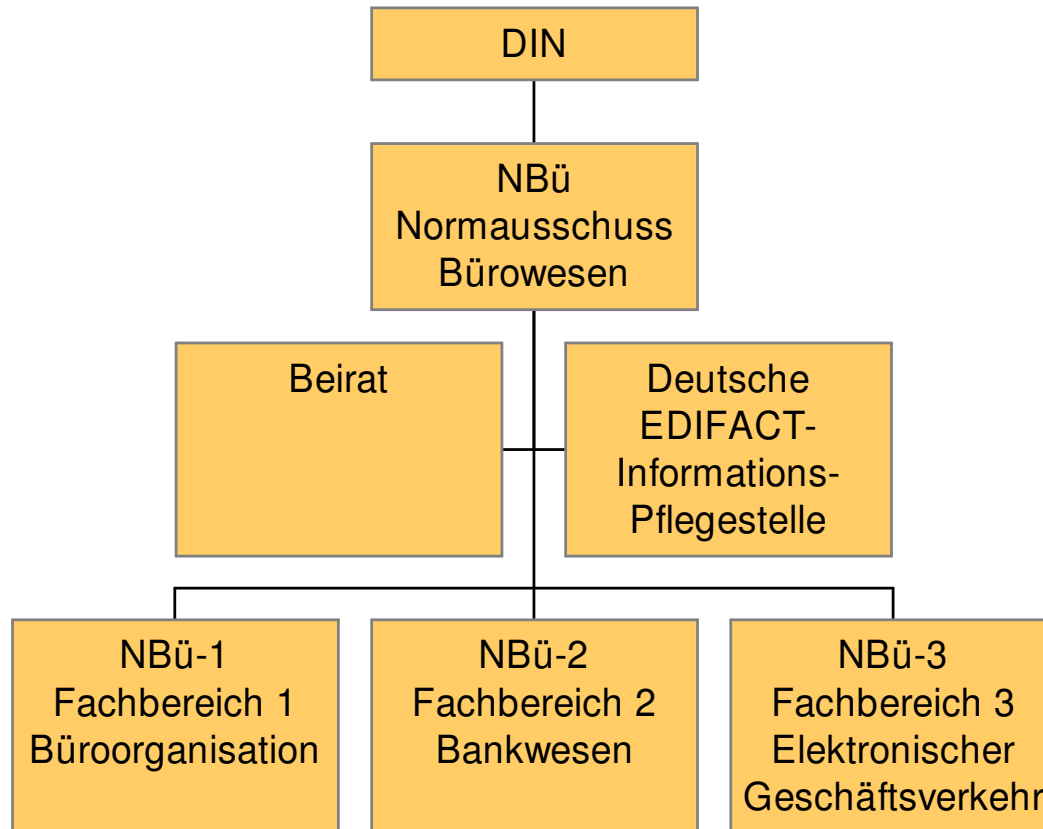
- **1977: SEDAS**
 - “Standardregelungen einheitlicher Datenaustauschsysteme”
 - Industrie und Handel, DE und AT
 - CCG - Centrale für Coorganisation, Köln (heute GS1 Germany)
- **1978: VDA-Norm**
 - VDA: Verband der deutschen Automobilindustrie e.V.
- **1981: GTDI**
 - Als TDID Teil 4 (Syntaxregeln) von UN/ECE veröffentlicht
 - GTDI: *Guidelines for Trade Data Interchange*
 - TDID: *Trade Data Interchange Directory*
- **1982: ANSI ASC-X12**
 - *Accredited Standard Committee X12 for Electronic Business Data Interchange (EBDI) by the Am. Ntl. Standards Institute*



- **1987: EDIFACT Syntaxregeln**
 - Überarbeitung / Aktualisierung durch die UN/JEDI Group
 - Synthese aus GTDI und ANSI X.12
 - Juli: UN/JEDI Group verabschiedet
 - *Message Design Guidelines*
 - Erste Nachricht: INVOIC
 - September: Übernahme der UN/ECE-Empfehlungen der EDIFACT-Syntaxregeln durch ISO, CEN, DIN:
 - International: **ISO 9735** (15. Juli 1988)
 - EU-Ebene: EN 27 372
 - Deutsche Norm: DIN 16559
- **1990: EANCOM** (wichtiger Subset)
- **Mitte 2007: D.07A – 195+17** Nachrichtentypen







- NBü-3.01: Service-Nachrichten und Nachrichtenbausteine
- NBü-3.02: Technische Prüfung und Unterstützung
- NBü-3.03: Datentransfer, NBü-3.04: Öffentlichkeitsarbeit ...
- NBü-3.11 bis 3.52: versch. Branchen-Nachrichten



- Bestandteile des UN/TDID (UN Trade Data Interchange Directory)
 - EDIFACT Syntax Rules (ISO 9735)
 - Syntax implementation guidelines
 - EDED (EDIFACT Data Elements Directory, Teilmenge des UN/TDED)
 - UNCL (UN Code List)
 - EDCD (EDIFACT Composite Data Elements Directory)
 - EDSD (EDIFACT Segments Directory)
 - EDMD (EDIFACT Standard Messages Directory); vgl. UNSM
 - Message Design Guidelines
 - Uniform Rules of Conduct for the Interchange of Trade Data by Teletransmission (UN/CID), sowie
 - erläuterndes Material, Glossar, Lizenzvereinbarung, ...
- Demo am Beispiel D.07A



- UNOA

- A - Z
- Ziffern 0 - 9
- Leerzeichen
- Punkt .
- Komma ,
- Bindestrich -
- Runde Klammern ()
- Schrägstrich /
- Gleichheitszeichen =
- Ausrufungszeichen !
- Prozentzeichen %
- Anführungszeichen “
- Kaufmännisches und &
- Stern *
- Semikolon ;
- kleiner, größer < >
- Apostroph ‘
- Pluszeichen +
- Doppelpunkt :
- Fragezeichen ?

- UNOB

- Wie UNOA, aber zusätzlich:
- Kleinbuchstaben:
 - a - z
- nicht-druckbare Informationstrennzeichen:
 - IS1, IS3, IS4

- Bemerkungen:

- UNOA:
 - Von IA5 abgeleitet, Telex-kompatibel
- UNOB:
 - Von 7-bit-Code ISO 646 abgeleitet (ASCII)



- **SV1, 15.07.1988**
 - Zeichensätze: UNOA, UNOB
 - I.w. die Grundlage bis heute
- **SV2, 1.11.1990**
 - Minimale Dokumentationsänderungen
 - DE 0054: n..3 C nach an..3 M
 - DE 0052: an..2 C nach an..2 M
 - Neue Default-Trennzeichen: IS1, IS3, IS4 (ohne UNA-Verwendung)
- **SV3, 1992**
 - UNOC ISO8859-1 “Latin 1”
 - UNOD ISO8859-2 “Latin 2”
 - UNOE ISO8859-5 “Latin / Cyrillic”
 - UNOF ISO8859-7 “Latin / Greek”
 - Service-Nachrichtentyp CONTRL spezifiziert (nicht in UN/TDID)
 - Formale Regelung für *subsets* und *amendments*



- **SV4 - Umfassende Reformen und Erweiterungen**
 - Part 1 (*Common syntax rules*), 1.10.1998
 - Wiederholungs-Elemente, *service repetition character*
 - Neue Anwendungsbereiche, dafür neue Service-Segmente
 - Alle Zeichensätze von ISO8859-1 bis ISO8859-9 unterstützt
 - *Dependency notes*: Formale Regelungen zur Beschreibung von Abhängigkeiten zwischen Segmenten bzw. (C)DE gleicher Stufe
 - *Segment collision prevention*: Servicesegmentpaar UGH, UGT
 - Erweiterungen in UNB, UNH; Reform von UNG
 - Part 2 (*Batch EDI*), 1.10.1998
 - Part 3 (*Interactive EDI*), 1.10.1998
 - Part 4 (CTRL), 15.12.1998
 - Part 5 (*Security rules*), 1.04.1999
 - (Parts 6 - 9: Siehe Fortsetzung)



- **SV4** (Forts.)
 - Part 6 (AUTACK), 1.04.1999 → Elektron. Unterschrift!
 - Part 7 (*Security rules - confidentiality/encryption*), 1.08.1999
 - Part 8 (*Associated data - packages, attachments*), 1.10.1999
 - Part 9 (KEYMAN - *security key & certificate msg.*), 1.04.1999



- Status 0
 - Arbeitspapier
 - Wird nicht herausgegeben
- Status P
 - Entwurf - erste Arbeitsphase abgeschlossen
 - Wird zur Begutachtung herausgegeben
- Status 1
 - Norm-Entwurf
 - Wird zur Probeanwendung freigegeben
- Status 2
 - Als Norm registriert und als UNSM zur Anwendung freigegeben
- Revisions
 - Weiterentwicklungen sind in Form von *revisions* von Release zu Release möglich. Bsp.: INVOIC erreicht in D.01C *release*-Nr. 13



- (87) Erste Nachricht - noch kein richtiges Release
- 88.1 Messages (nur eine: INVOIC), segments, composites
- 90.1 (2), 90.2 (+15)
 - Nun vollständig:
Messages, segments, composites, elements, codes
- 91.1 (34), 91.2 (= 90.1 +17, *2)
 - Reform - neue Segmentstruktur: Modularität, kleinere Einheiten
 - Neues Verwaltungskonzept:
 - YY.1 enthält auch Status-1 Nachrichtentypen,
 - YY.2 nur Status-2
- 92.1 (=91.1 + 18, *14)
- 93.2 (=91.2 + 9, *18)



- D.93A, S.93A
 - ferner nur geplant: D.93B, S.93B
 - D = Draft - enthält Nachrichten in Status 1 & 2
 - S = Standard - enthält Nachrichten in Status 2
- D.94A, D.94B
 - Erneuter Konzeptwechsel (Vereinfachung):
 - D = Directory - enthält Status-1 wie auch Status-2 Nachrichten
- Ab jetzt i.w. abwärtskompatible Weiterentwicklung:
 - Ergänzung um neue Nachrichtentypen
 - Weiterentwicklung vorhandener N.T.
- D.95A, D.95B
- D.96A, D.96B
- D.97A, D.97B



- Ab 1998 neue Qualitäten hinzutretend durch SV4:
- D.98A, D.98B (batch, SV3 & interactive, SV4)
 - D.98A: Erstmals mit Syntax Version 4
(nur *Interactive* EDI; neu)
- D.99A, D.99B
 - (*batch SV3, batch SV4, interactive SV4*)
 - D.99B: Erstmals auch *Batch-EDI* mit Syntax-Version 4
- D.00A, D.00B
- D.01A, D.01B, D.01C
- D.02A, D.02B
- D.03A, D.03B
- D.04A, D.04B
- D.05A, D.05B



UN/EDIFACT “*bottom-up*”

Aufbau einer EDIFACT-Nachricht,
beginnend mit einzelnen
Datenelementen



Erinnerung: Unser Ziel!



```
UNB+UNOB:3+2100000000005:14+2900000000001:14+02  
0502:2350+1234'UNH+H123+ORDERS:D:96A:UN:EAN008'  
BGM+220+1234567'DTM+137:20020415:102'DTM+2:20020  
420:102'NAD+BY+2100000000005::9++XY~Handels~AG'N  
AD+SU+2900000000001::9++ABC~GmbH'LIN+1++2900000  
100008:EN'QTY+21:20'LIN+2++2900000200005:EN'QTY+2  
1:10:PCE'UNS+S'CNT+1:30'UNT+13+H123'UNZ+1+1234'
```

(Blanks durch ~ ersetzt)



Eine einfache Beispiel-Bestellung



- Kopfinformation
 - Bestellnummer
 - Bestelldatum
 - Gewünschtes Lieferdatum
 - Besteller

 - Lieferant

 - Positionsinformation
 - Position
 - Artikelnummer
 - Menge
 - Position
 - Artikelnummer
 - Menge

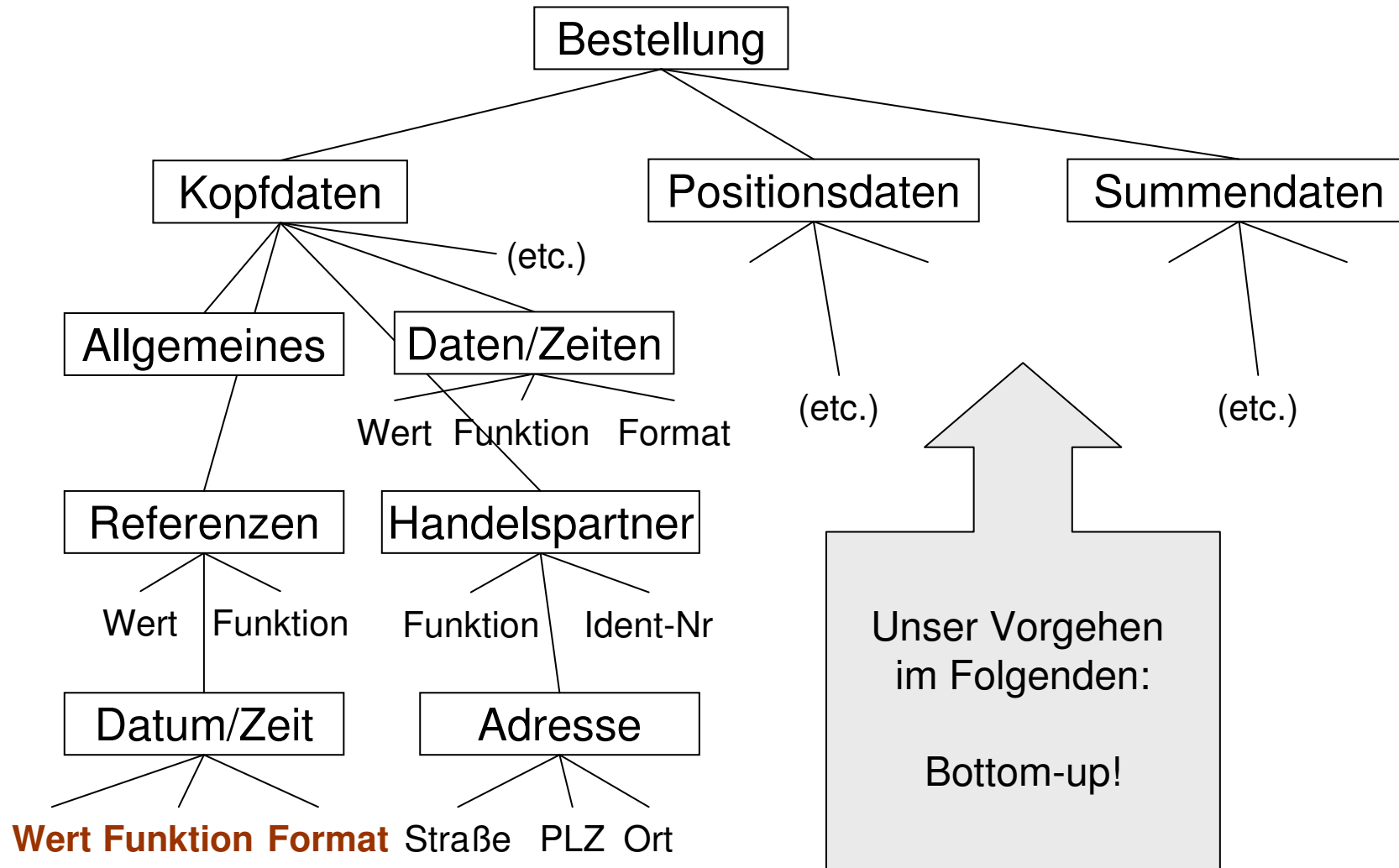
 - Summeninformation
 - Gesamtzahl bestellter Artikel
- Beispieldaten:
 - 1234567
 - 15. 4. 2002
 - 20. 4. 2002
 - XY Handels AG
 - GLN = 2100000000005
 - ABC GmbH
 - GLN = 2900000000001

 - Pos. 1
 - GTIN = 2900000100008
 - 20 Stück
 - Pos. 2
 - GTIN = 2900000200005
 - 10 Stück

 - 30 Stück



Erinnerung: Gruppenbildung





Beispiel-Bestellung: Codierung



- Datum der Bestellung (Belegdatum)
 - Inhalt hier: 15.4.2002
 - Problem 1: Übertragungsformat
 - Als String “15.4.2002” ? Ist das international?
 - Varianten: 15.04.2002, 15.4.02, 15-04-02, 4/15/02, ...
 - Als String “2002-04-15”, wie z.B. in XML-Schema?
 - Oder kompakt als Zahl “20020415” ?
 - Problem 2: Codierung der Bedeutung
Hier: “Belegdatum” oder “Gewünschtes Lieferdatum”?
 - Standardisierte Darstellung?
 - Kompakte Darstellung?
 - Sprachunabhängig?



- Antworten zur Codierungsfrage
 - Semantische Ebene
 - **Basic Semantic Repository (BSR):**
Schaffung einer Sammlung von Fachbegriffen mit klarer Definition + Codenummer und abgestimmter Übersetzung in alle gängigen Sprachen.
 - Codierungsebene
 - Darstellung entsprechender Inhalte in präzise gefassten und eindeutig adressierten **Datenelementen**
 - UN: UN Layout-Key, Trade Data Elements Directory
 - EDIFACT: Data Elements Directory (EDED)
 - ebXML: Core Components



- Antworten zur Codierungsfrage (Forts.)
 - Qualifizierung von *Bedeutungen*
 - Nicht-sprechende, i.d.R. numerische Codes repräsentieren
 - Begriffe aus dem BSR, z.B. 137 = Belegdatum
 - Formatspezifikationen, z.B. 102 = YYYYMMDD
 - Rückgriff auf vorhandene internationale Normen wenn möglich
 - z.B. ISO Ländercodes (2- oder 3-Zeichen-Code, DE bzw. FRA)
 - ISO Währungscodes (z.B. USD, GBP, DEM, EUR)
 - Verwaltung
 - über EDIFACT *Code List* (EDCL), mit EDED als Index
- Zusammenfassung
 - **Bedeutung** Belegdatum: 137
 - **Wert** Datum/Zeit: 20020415
 - **Formatspezifikation**: 102



EDIFACT “*bottom-up*”: EDED



- Ein einfaches Datenelement aus EDED
 - Code: 4-stellig
 - Kurztext Einzeilige Beschreibung
 - Darstellung Codierte Darstellung
 - Beschreibung Ausführliche Beschreibung

- Beispiel:
 - Code: **2860**
 - Kurztext *Date or time or period value*
 - Darstellung an..35
 - Beschreibung *The value of a date, a date and time, a time or of a period in a specified representation.*



- Anmerkungen
 - Zahlenkonventionen bei den Codes
 - Codes sind immer vierstellig, bei Bedarf mit führenden Nullen
 - Codes < 100 sind für EDIFACT-interne Zwecke reserviert
 - DE mit gerader Codenummer enthalten variable Inhalte,
 - DE mit ungerader Codenummer enthalten Qualifier / Codes z.B. gemäß EDCL
 - Formatangaben gemäß ISO 9735 (Beispiele)
 - a3 Alphabetisch, genau drei Zeichen
 - n..8 Numerisch, höchstens 8 Stellen
 - an3 Alphanumerisch, genau drei Zeichen
 - an..35 Alphanumerisch, höchstens 35 Zeichen
 - Bei numerischen Längenangaben NICHT mitgezählte Zeichen:
 - Vorzeichen
 - Dezimalzeichen (, oder .)
 - Exponentialzeichen (E oder e) sowie der Exponent selbst



- Weitere Regeln zu numerischen Angaben:
 - Generelle Ausrichtung nach ISO 6093, mit Einschränkungen:
 - Tausenderblöcke werden nicht gekennzeichnet (kein Triadenkennz.!)
 - Als Dezimalzeichen sind sowohl Komma als auch Punkt zulässig.
 - Die Null vor dem Dezimalzeichen darf entfallen
 - Auf das Dezimalzeichen muß mindestens eine Ziffer folgen
 - Vorzeichen sind nur in Präfixnotation zulässig
 - Pluszeichen und Space sind nicht zulässig
 - Führende Nullen bei Angaben variabler Länge sind zu vermeiden
 - Ausnahme: Eine Null vor dem Dezimalzeichen ist zulässig.
 - Signifikante Nullen sind zulässig, z.B. als Ausdruck der Genauigkeit eines Messergebnisses. Nicht signifikante Nullen sind zu vermeiden.

– Beispiele

- | | | | | |
|--------|---|------|----------|----------------|
| • 2,00 | 2 | 0.5 | .5 | Zulässig |
| • 0, | , | +0.5 | 1.000,50 | NICHT erlaubt! |



- Weitere Beispiele:
 - Code: **2005**
 - Kurztext *Date or time or period function code qualifier*
 - Darstellung an..3
 - Beschreibung *Code qualifying the function of a date, time or period.*

 - Code: **2379**
 - Kurztext *Date or time or period format code*
 - Darstellung an..3
 - Beschreibung *Code specifying the representation of a date, time or period.*

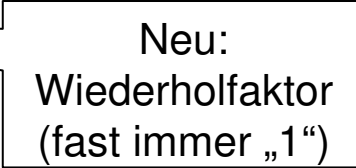


- Erkenntnis:
 - Eine konkrete Datums/Zeitangabe erfordert praktisch immer das Zusammenspiel mehrerer DE
- Schlussfolgerung
 - Bündelung mehrerer DE zu einer höheren logischen Einheit: **Composite Data Element (CDE)**, verwaltet mittels **EDCD**.
 - Bestandteile: **Component elements (CE)**.
- Struktur eines CDE:
 - Code: analog DE
 - Kurztext analog DE
 - Beschreibung analog DE
 - CE-Liste neu:
 - Geordnete Liste der Codes der enthaltenen DE
 - mit Angabe M/C (*mandatory / conditional*)
 - ab SV4 mit Angabe “Wiederholfaktor”



- Beispiel:

- Code: **C507**
- Kurztext: *Date / time / period*
- Beschreibung: *Date and/or time, or period relevant to the specified date/time/period type.*

– DE-Liste	2005	M	1	
	2380	C	1	
	2379	C	1	

- Bemerkungen

- CDE-Codes sind vierstellig (an4): Ein Buchstabe + 3 Ziffern
- Zulässige Anfangsbuchstaben:
 - “C” Normalfall
 - “E” CDE aus Segment für Interactive EDI (nur SV4)
 - “S” bei ISO 9735-internen CDE (*service* CDE)



EDIFACT “*bottom-up*”: EDCD



- EDIFACT-Codierung
 - Trennzeichensyntax, mehrstufig!
 - Standard-Trennzeichen zwischen CE ist der Doppelpunkt “:”
 - Beispiel C507 für Belegdatum: `137:20020415:102`
 - analog: Gewünschtes Lieferdatum: `2:20020420:102`
- Umgang mit fehlenden optionalen CE gemäß Syntaxregeln:
 - fehlende/leere CE am CDE-Ende werden komplett ausgelassen
 - fehlende/leere CE innerhalb eines CDE werden durch entsprechende Wiederholung des Trennzeichens übersprungen
 - Beispiele (das folgende CDE bestehe aus 5 CE, alle optional):
 - a) CE:CE:CE:CE:CE vollständige Bestückung mit 5 CE
 - b) CE:::CE:CE CE 2 und 3 werden übersprungen
 - c) :CE:CE CE 1, 4 und 5 werden ausgelassen
 - d) **CE:CE:::** **falsch!** Korrekt wäre “CE:CE”



- **Segmente**

- Bündelung mehrerer DE/CDE zur Beschreibung komplexerer Bausteine von Geschäftsdokumenten
- Verwaltung über
 - *segment TAG* (an3)
(es bürgerten sich hier sprechende, an Englisch angelehnte Codes ein)
 - und **Segmentverzeichnis EDSD**

- Ein Segment aus EDSD

- TAG: 3-stellig (an3)
- Kurztext Einzeilige Beschreibung
- Funktion Ausführliche Beschreibung
- (C)DE-Liste Geordnete Liste der enthaltenen einfachen oder Composite-Datenelemente
 - mit Angabe M/C (*mandatory / conditional*)
 - ab SV4 mit Angabe “Wiederholfaktor”



- Einfaches Beispiel: DTM

- TAG: **DTM**
- Kurztext: *Date / Time / Period*
- Funktion: *To specify date, and/or time, or period.*
- (C)DE-Liste:
C507 M 1

- Bemerkungen

- Das Segment ist insofern trivial, als es nur aus einem CDE besteht.
- Grund für die Umhüllung des CDE mit einem TAG ist die angestrebte Modularität:
 - Datums/Zeitangaben werden an vielen Stellen benötigt, auch auf der Ebene von Segmenten!



- Komplexeres Beispiel: NAD
 - TAG: **NAD**
 - Kurztext: *NAME AND ADDRESS*
 - Funktion: *To specify the name/address and their related function, either by C082 only and/or unstructured by C058 or structured by C080 thru 3207.*

– (C)DE-Liste:

3035	M	1
C082	C	1
C058	C	1
C080	C	1
C059	C	1
3164	C	1
C819	C	1
3251	C	1
3207	C	1



EDIFACT “*bottom-up*”: EDSD



- Darstellung im EDSD
 - Die reine (C)DE-Liste ist schwer lesbar. Daher enthält das EDSD auch die CE der CDE, sowie die Codes, Kurzbeschreibungen und Darstellungen der CE und DE
- Beispiel NAD:

3035	PARTY FUNCTION CODE QUALIFIER	M	1	an..3
C082	PARTY IDENTIFICATION DETAILS	C	1	
3039	Party identifier	M		an..35
1131	Code list identification code	C		an..17
3055	Code list responsible agency code	C		an..3
C058	NAME AND ADDRESS	C	1	
3124	Name and address description	M		an..35
3124	Name and address description	C		an..35
3124	Name and address description	C		an..35
3124	Name and address description	C		an..35
3124	Name and address description	C		an..35



- Beispiel NAD (Forts.):

C080	PARTY NAME	C	1	
3036	Party name	M		an..35
3036	Party name	C		an..35
3036	Party name	C		an..35
3036	Party name	C		an..35
3036	Party name	C		an..35
3045	Party name format code	C		an..3
C059	STREET	C	1	
3042	Street and number or post office box identifier	M		an..35
3042	Street and number or post office box identifier	C		an..35
3042	Street and number or post office box identifier	C		an..35
3042	Street and number or post office box identifier	C		an..35
3164	CITY NAME	C	1	an..35



EDIFACT “*bottom-up*”: EDSD



- Beispiel NAD (Forts.):

C819 COUNTRY SUB-ENTITY DETAILS	C	1	
3229 Country sub-entity name code	C		an..9
1131 Code list identification code	C		an..17
3055 Code list responsible agency code	C		an..3
3228 Country sub-entity name	C		an..70
3251 POSTAL IDENTIFICATION CODE	C	1	an..17
3207 COUNTRY NAME CODE	C	1	an..3



- EDIFACT-Codierung eines Segments
 - Standard-Trennzeichen zwischen (C)DE eines Segments ist das Pluszeichen “+” (“*data element separator*”)
 - Standard-Trennzeichen zwischen Segmenten ist das Apostroph “ ‘ ” (“*segment terminator*”)
- Beispiel NAD für Lieferant: NAD+SU+2900000000001::9’
- Beispiel NAD für Sekretariat, mit strukturierter Klartext-Anschrift:

NAD+LT+++FACHHOCHSCHULE WIESBADEN:SEKRETARIAT
INFORMATIK:GEB. C+KURT-SCHUMACHER-RING 18+
WIESBADEN++65197+DE’
- Bemerkungen zur Anwendung von NAD:
 - Je strukturierter, desto besser → Anwendung von C058 meiden
 - C082 besser als C080...3207, C080...3207 besser als C058 !



- Umgang mit fehlenden optionalen DE gemäß Syntaxregeln:
 - fehlende/leere DE am Segmentende werden komplett ausgelassen
 - fehlende/leere DE innerhalb eines Segments werden durch entsprechende Wiederholung des Trennzeichens übersprungen
- Beispiele (Segment bestehe aus 4 DE, alle optional):
 - a) TAG+DE+DE+DE+DE’ vollständige Bestückung mit 4 DE
 - b) TAG+DE+++DE’ DE 2 und 3 werden übersprungen
 - c) TAG++DE+DE’ DE 1 und 4 werden ausgelassen
 - d) TAG+DE+DE++’ **falsch!**
Korrekt wäre “ TAG+DE+DE’ ”
 - e) TAG+DE+::CE++CE:CE’ auch in Kombination mit
CDE / CE-Regeln



Die Segmente zur Beispiel-Bestellung



- Kopfinformation
 - Bestellnummer
 - Bestelldatum
 - Gewünschtes Lieferdatum
 - Besteller
 - Lieferant
- Positionsinformation
 - Position
 - Artikelnummer
 - Menge
 - Position
 - Artikelnummer
 - Menge
- Summeninformation
 - Gesamtzahl bestellter Artikel
- Segmente für Beispieldaten:
 - BGM 1234567
 - DTM 15. 4. 2002
 - DTM 20. 4. 2002
 - NAD XY Handels AG
 - GLN = 2100000000005
 - NAD ABC GmbH
 - GLN = 2900000000001
 - LIN Pos. 1
 - GTIN = 2900000100008
 - QTY 20 Stück
 - LIN Pos. 2
 - GTIN = 2900000200005
 - QTY 10 Stück
 - CNT 30 Stück



Weitere Information: BGM



- BGM BEGINNING OF MESSAGE
- *Function: To indicate the type and function of a message and to transmit the identifying number.*

C002 DOCUMENT/MESSAGE NAME	C	1	
1001 Document name code	C		an..3
1131 Code list identification code	C		an..17
3055 Code list responsible agency code	C		an..3
1000 Document name	C		an..35
C106 DOCUMENT/MESSAGE IDENTIFICATION	C	1	
1004 Document identifier	C		an..35
1056 Version identifier	C		an..9
1060 Revision identifier	C		an..6
1225 MESSAGE FUNCTION CODE	C	1	an..3
4343 RESPONSE TYPE CODE	C	1	an..3



Weitere Information: LIN



- LIN LINE ITEM
- *Function: To identify a line item and configuration.*

1082 LINE ITEM IDENTIFIER	C	1	an..6
1229 ACTION REQUEST/NOTIFIC. DESCR. CODE	C	1	an..3
C212 ITEM NUMBER IDENTIFICATION	C	1	
7140 Item identifier	C		an..35
7143 Item type identification code	C		an..3
1131 Code list identification code	C		an..17
3055 Code list responsible agency code	C		an..3
C829 SUB-LINE INFORMATION	C	1	
5495 Sub-line indicator code	C		an..3
1082 Line item identifier	C		an..6
1222 CONFIGURATION LEVEL NUMBER	C	1	n..2
7083 CONFIGURATION OPERATION CODE	C	1	an..3



Weitere Information: QTY



- QTY QUANTITY
- *Function:* *To specify a pertinent quantity.*

C186 QUANTITY DETAILS	M	1	
6063 Quantity type code qualifier	M		an..3
6060 Quantity	M		an..35
6411 Measurement unit code	C		an..3

- CNT CONTROL TOTAL
- *Function:* *To provide control total.*

C270 CONTROL	M	1	
6069 Control total type code qualifier	M		an..3
6066 Control total value	M		n..18
6411 Measurement unit code	C		an..3



Segmentdarstellung der Beispiel-Bestellung



- Kopfinformation
 - Bestellnummer
 - Bestelldatum
 - Gewünschtes Lieferdatum
 - Besteller
 - Lieferant
- Positionsinformation
 - Position
 - Artikelnummer
 - Menge
 - Position
 - Artikelnummer
 - Menge
- Summeninformation
 - Gesamtzahl bestellter Artikel

- Segmente für Beispieldaten:
 - BGM+220+1234567'
 - DTM+137:20020415:102'
 - DTM+2:20020420:102'
 - NAD+BY+2100000000005::9++
XY Handels AG'
 - NAD+SU+2900000000001::9++
ABC GmbH'

LIN+1++2900000100008:EN'
QTY+21:20'

LIN+2++2900000200005:EN'
QTY+21:10:PCE'

CNT+1:30'

Segment-
gruppe



- **Segmentgruppen (SG)**
 - Zusammenfassung mehrerer Segmente zu einer logischen Einheit
 - Regel: Das erste Segment einer SG (“Triggersegment”) ist stets “M”
 - SG tragen wie Segmente
 - eine M/C-Kennung und
 - einen Wiederholfaktor
 - SG unterscheiden sich aber von Segmenten:
 - Sie tragen keinen Namen und auch keine Beschreibung, sondern werden nur fortlaufend durchnummeriert.
 - Sie werden nur innerhalb einer Nachricht gebildet und werden daher nicht in einem eigenen Verzeichnis verwaltet.
 - Sie dienen der logischen Strukturierung einer Nachricht.
 - Dennoch finden sich gleiche Segmentgruppen in vielen Nachrichten
 - **WICHTIG:**
 - **SG sind rein logische Strukturen, ohne Entsprechung auf der Ebene der Segmente. Sie treten im EDIFACT-Datenstrom nicht explizit in Erscheinung!**



- **Hierarchiestufe** (*Level*)

- Aus Segmenten lassen sich Segmentgruppen (SG) bauen
- Aus Segmenten und SG lassen sich übergeordnete SG bauen
- Dadurch entsteht eine Hierarchie (Baumstruktur) von Segmenten und SG.
- Konvention:
 - Die Verschachtelungstiefe eines Segments / einer SG zählt man und nennt sie **Hierarchiestufe** des Segments / der SG
 - Die Aufzählung beginnt mit Stufe 0. Das ist nicht die Stufe des (nur gedachten) Wurzelknotens des Baums, sondern die oberste Stufe der real auftretenden Segmente
- Einschränkende Sonderregeln für Stufe 0:
 - Keine Segmentgruppen
 - Nur Segmente mit genau einem Exemplar
 - Einheiten, die diese Vorgaben nicht erfüllen, aber sonst keine hierarchische Abhängigkeit aufweisen, zählen zu Stufe 1 !



- **Standard Messages - UNSM**
 - Eine Abfolge von Segmenten bzw. SG
 - Einschließlich der Möglichkeit, Blöcke zu wiederholen
 - Zweck: Darstellung kompletter Geschäftsdokumentstypen (“Nachrichtentypen”)
 - Einzelnes Exemplar: “Nachricht”
 - Offiziell im EDIFACT-Release enthaltene Nachrichten heißen *UN Standard Messages (UNSM)*
- **Verwaltung:**
 - Über spezielles Verzeichnis (**EDMD**)
 - Struktur eines Eintrags:
 - Abfolge der Segmente und SG
 - Definition der SG dieses Nachrichtentyps
 - Spezifikationen zu Wiederholungen: M/C, max. Wiederholfaktor
 - Erläuterungen



EDIFACT “bottom-up”: UNSM



Beispiel: APERAK - Application error and acknowledgement message

Pos	Tag Name	S	R
0010	UNH Message header	M	1
0020	BGM Beginning of message	M	1
0030	DTM Date/time/period	C	9
0040	FTX Free text	C	9
0050	CNT Control total	C	9
0060	----- Segment group 1 -----	C	99-----+
0070	DOC Document/message details	M	1
0080	DTM Date/time/period	C	99-----+
0090	----- Segment group 2 -----	C	9-----+
0100	RFF Reference	M	1
0110	DTM Date/time/period	C	9-----+



EDIFACT “bottom-up”: UNSM

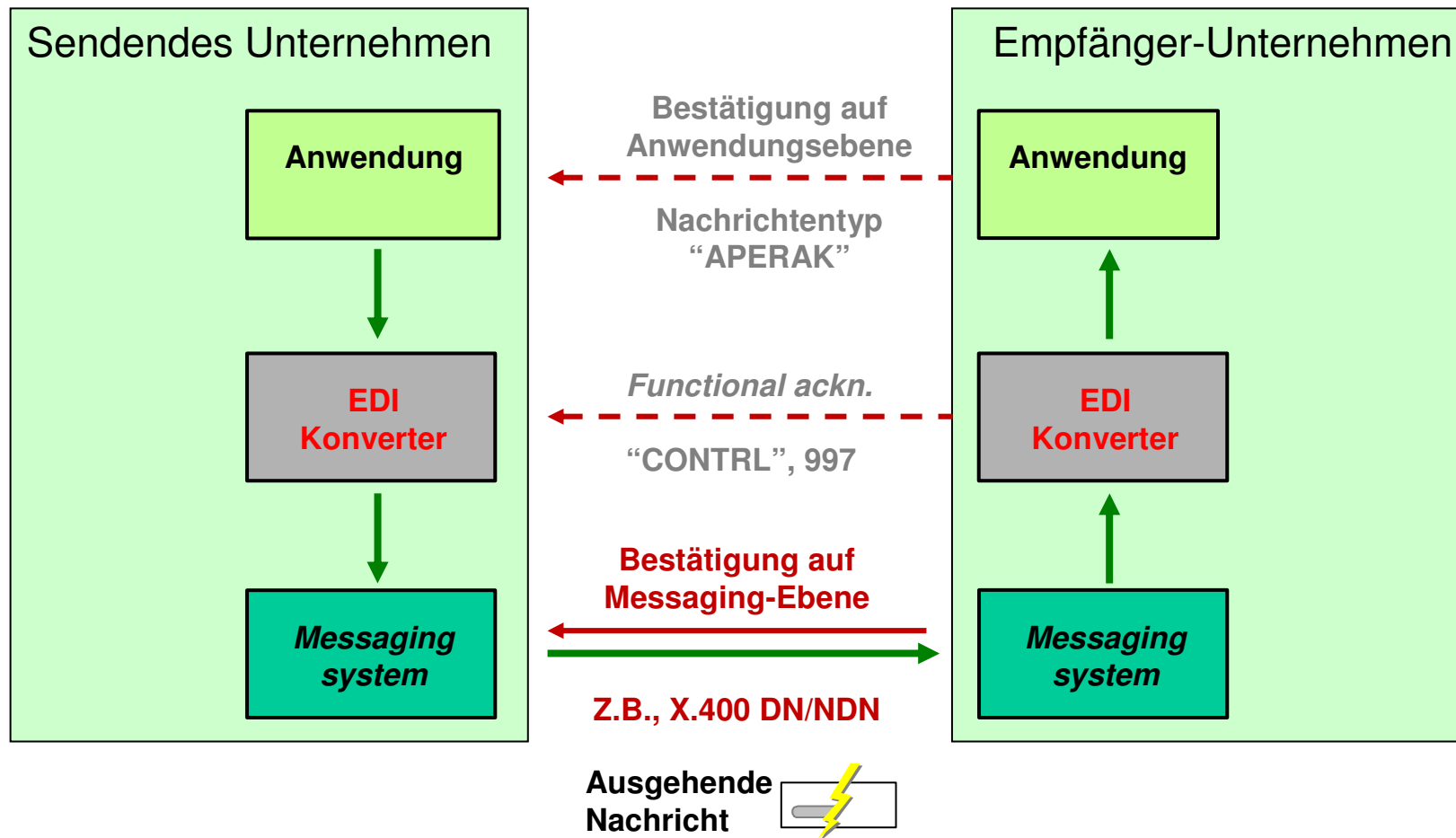


Beispiel: APERAK - Application error and acknowledgement message (Forts.)

Pos	Tag Name	S	R
0120	----- Segment group 3 -----	C	9-----+
0130	NAD Name and address	M	1
0140	CTA Contact information	C	9
0150	COM Communication contact	C	9-----+
0160	----- Segment group 4 -----	C	99999-----+
0170	ERC Application error information	M	1
0180	FTX Free text	C	1
0190	----- Segment group 5 -----	C	9-----+
0200	RFF Reference	M	1
0210	FTX Free text	C	9-----++
0220	UNT Message trailer	M	1



Ebenen der Zustellbestätigung (→ *Messaging*)





- **Das *Branching Diagram***

- Beliebte Übersichtsdarstellung
- Die Bestandteile des Nachrichtentyps werden baumartig dargestellt
- Jeder Knoten des Baums entspricht einem Segment oder einer SG
- Knotenangaben:

- Name/TAG
- Status (M/C)
- Wiederholfaktor

TAG	
M	10

- Vertikale Ausrichtung der Knoten gemäß Hierarchiestufe
- Horizontale Ausrichtung gemäß Sequenzdefinition des Nachrichtentyps

- **Beispiel APERAK**

- (*on-line* Dokumentation, EANCOM-Subset)



- Regeln für das Wiederholen von Segmenten und SG (Bsp.)
 - M 1 An dieser Stelle genau ein Exemplar gefordert
 - M 5 Hier mindestens ein, höchstens 5 Exemplare
 - C 10 Optionaler Knoten, darf hier höchstens 10-mal auftreten
- Das Zusammenspiel von Segmenten und SG
 - Das erste Segment einer SG heißt **Triggersegment** und muss immer auftreten, wenn ein Exemplar der SG beginnt.
 - Trotz der Angabe “M” für das Triggersegment darf die ganze SG durchaus auch vollständig fehlen. Die Regel sagt nur:
 - Wenn ein SG-Exemplar angelegt wird, dann muss es mit dem Triggersegment beginnen.



EDIFACT “*bottom-up*”: Uxx



- **Die Service-Segmente** (hier: Beschränkung auf SV3)
 - Zweck: Geordnete und immer gleichartige Behandlung der in allen Nachrichtentypen notwendigen EDIFACT-internen Verwaltungsinformation
 - Gemeinsames Kennzeichen: Segment TAG-Muster “Uxx”
- **UNS**
 - Kleines Segment, Hilfsmittel zur Belegstrukturierung
 - Trennt wenn nötig zwischen Belegkopf, -positions, und -summenteil
 - Besteht nur aus einem DE: 0081
 - Es gibt nur 2 zulässige Codes (und damit Vorkommnisse) für 0081:
 - **UNS+D** Trennt Kopf- und Positionsteil (*header / detail*)
 - **UNS+S** Trennt Positions- und Summenteil (*detail / summary*)
 - Bem.: Zahlreiche Nachrichtentypen enthalten UNS nicht!



EDIFACT “bottom-up”: Uxx



- **UNH / UNT - Header und Trailer**

- Verwaltungsrahmen um die Nutzsegmente einer Nachricht

- **UNT – Message Trailer**

- Erstes DE: Anzahl Segmente, incl. UNH und UNT selbst

- Zweites DE: Verweis auf UNH (“schließende Klammer”)

- Segment : **UNT, Message Trailer**

Function: To end and check the completeness of a Message

<u>Ref.</u>	<u>Repr.</u>	<u>Name</u>	<u>Remarks</u>
0074	n..6 M	NUMBER OF SEGMENTS IN THE MESSAGE	Control count including UNH and UNT
0062	an..14 M	MESSAGE REFERENCE NUMBER	Shall be identical to 0062 in UNH

Beispiel: **UNT+15+H12345'**

Verständnis-Test:
Was ist der kleinste zulässige Wert für 0074 ?



- **UNH - *Message Header***

- Kennzeichnet die **Nachricht** auf **eindeutige** Weise innerhalb der nächsten übergeordneten Struktur, unabhängig vom Inhalt
- Verweist auf das zugrunde liegende **UN/EDIFACT-Verzeichnis**
- Verweist optional auf ein darauf aufbauendes, im Folgenden zugrunde liegendes **Subset**
- Legt damit die Grundlagen für eine automatische Konformitätsprüfung sowie für *functional acknowledgment*/CONTRL
- Optional, selten verwendet:
 - Weitere Referenz, Schlüssel für spätere Rückverweise
 - Informationen zur Unterscheidung mehrfacher Übermittlungen
- (Konkrete Betrachtung im Normen-Original)

- **Beispiel-Instanz:**

- **UNH+H12345+INVOIC:D:96A:UN:EAN008'**
- Bemerkung: **rot**: DE 0062, vgl. UNT



- **UNG / UNE**

- Zur Bildung von Nachrichtengruppen
- Sollen Nachrichten verschiedener Typen zu einer Übertragungsdatei gebündelt werden, müssen sie zu “sortenreinen” Gruppen sortiert werden
- Gruppen werden mit dem Segment UNG eingeleitet und mit UNE beendet, konzeptionell analog zu UNH und UNT
- UNE zählt analog zu UNT die enthaltenen nächstkleineren Einheiten; dies sind nun die Nachrichten der Gruppe!
- UNG enthält Elemente / Konzepte des UNH als auch der übergeordneten Struktur, UNB (s.u.)
- In der Praxis werden UNG / UNE selten benötigt und sollen daher hier nicht vertieft werden.



EDIFACT “*bottom-up*”: Uxx



- **UNB / UNZ**
 - Begrenzen die Übertragungsdatei (“*interchange*”)

- **UNZ**
 - wird analog zu UNT bzw. UNE gebildet:
 - Verweis auf ID des zugehörigen UNB
 - Zähler der enthaltenen hierarchisch nächstkleineren Unterstrukturen
 - Im Normalfall:
 - Zähler = Anzahl enthaltener Nachrichten
 - Bei Anwendung von UNG/UNE:
 - Zähler = Anzahl enthaltener Gruppen
 - **ACHTUNG:** Mischfall “Gruppen + Nachrichten” nicht zulässig!



EDIFACT “*bottom-up*”: Uxx



- **UNB**: Grundlegendes Segment:
 - Regelt die EDIFACT-Syntaxversion
 - Legt den verwendeten Zeichensatz fest
 - *Routing/Messaging*: Definiert Sender und Empfänger (S/R)
 - *Interchange Control Reference*:
Eindeutiger Schlüssel, zusammen mit S/R !
 - Grundlage für *Functional acknowledgment* / auch “*ack. req.*” hier definiert
 - Ausgangspunkt für Fehlersuche zusammen mit dem Geschäftspartner
 - Weitere Verwaltungsinformation
 - Testkennzeichen



EDIFACT “*bottom-up*”: Uxx



- **UNB / UNZ**
 - (Besprechung der Einzelheiten mit dem Normen-Original)



EDIFACT “*bottom-up*”: Uxx



- **UNA** - *Service String Advice*, im Normalfall: **UNA: + . ? ' ’**
 - UNA ist ein optionales Pseudosegment, das ggf. dem UNB vorausgeht
 - Mit UNA lassen sich die EDIFACT Sonder/-Trennzeichen umdefinieren
 - Vom tatsächlichen Gebrauch dieser Möglichkeit ist aber dringend abzuraten: Sie wird anscheinend so selten eingesetzt, dass trotz offizieller Normung mit nur unzureichender Unterstützung durch Konvertersoftware zu rechnen ist.

<u>Repr.</u>	<u>Name</u>	<u>Remarks</u>
an1 M	COMPONENT DATA ELEMENT SEPARATOR	
an1 M	DATA ELEMENT SEPARATOR	
an1 M	DECIMAL NOTATION	Comma or full stop
an1 M	RELEASE INDICATOR	If not used, insert space character
an1 M	Reserved for future use	Insert space character
an1 M	SEGMENT TERMINATOR	



- **Weitere Bemerkungen zu UNA**

- In SV4 wird über das reservierte Zeichen verfügt: *
- *Release character* - das Fragezeichen
 - Soll eines der (hier) reservierten Zeichen z.B. als Teil eines DE-Inhalts übermittelt werden, so stellt man ihm das “*release character*” voraus
 - Unmittelbar im Anschluss an ein Fragezeichen verlieren Sonderzeichen ihren besonderen Status. Das gilt auch für das Fragezeichen selbst.
 - Das Fragezeichen wird - obwohl zulässig - praktisch nie durch ein *space* ersetzt.

- Beispiele (mit dem Freitext-Segment FTX):

"3 + 5 = 8"

→ **FTX+ZZZ+1++3 ?+ 5 = 8'**

"Wer war das? Bitte melden."

→ **FTX+ZZZ+1++Wer war das?? Bitte melden.+DE'**



Die vollständige Beispiel-Bestellung, korrigiert



**UNB+UNOB:3+2100000000005:14+2900000000001:14+020502:2350+
1234'**

UNH+H123+ORDERS:D:96A:UN:EAN008'

BGM+220+1234567'

DTM+137:20020415:102'

DTM+2:20020420:102'

NAD+BY+2100000000005::9++XY Handels AG'

NAD+SU+2900000000001::9++ABC GmbH'

LIN+1++2900000100008:EN'

QTY+21:20'

LIN+2++2900000200005:EN'

QTY+21:10:PCE'

UNS+S'

CNT+1:30'

UNT+13+H123'

UNZ+1+1234'



Die vollständige Beispiel-Bestellung



Anmerkungen:

- Die EDIFACT-Syntax sieht keinen Umbruch der Segmente in Zeilen vor.
- Die übliche Darstellung “Ein Segment pro Druckzeile” erfolgt nur aus Gründen der Lesbarkeit.
- Tatsächlich bestehen korrekte EDIFACT *interchanges* aus der reinen Folge der Segmente, ohne jegliche weitere Trennzeichen.
- Insbesondere sind <CR> und/oder <LF> nicht zu verwenden!
- Daher lautet unser Beispiel schließlich (Blanks durch ~ ersetzt):

```
UNB+UNOB:3+2100000000005:14+2900000000001:14+020502:2350+1234'UNH+H
123+ORDERS:D:96A:UN:EAN008'BGM+220+1234567'DTM+137:20020415:102'DTM
+2:20020420:102'NAD+BY+2100000000005::9++XY~Handels~AG'NAD+SU+29000
00000001::9++ABC~GmbH'LIN+1++2900000100008:EN'QTY+21:20'LIN+2++2900
000200005:EN'QTY+21:10:PCE'UNS+S'CNT+1:30'UNT+13+H123'UNZ+1+1234'
```



EDIFACT “*bottom-up*”: *Interchange*



- **Ein komplettes *Interchange***
 - Nun *top-down* Sicht
 - Klassischer Fall: *Batch-EDI*, ohne Neuerungen der SV4
- **Anzeige:**
 - OV-Folie mit Explosionszeichnung gemäß ISO 9735-2, oder
 - Direkte Anzeige der entsprechenden Seite aus ISO 9735-2, SV4



UN/EDIFACT aus Entwicklersicht

Die Ruby-Klassenbibliothek “edi4r”:
Wichtige Methoden & Klassen
(Beispiele)



EDIFACT: *Interchange*-Bildung mit edi4r



- Installation von “edi4r”
 - edi4r ist ein normales Gem, veröffentlicht auf Rubyforge, daher installierbar mit

```
$ gem install edi4r # EDIFACT SV1-3 und TDIDs D.96A & D.01
```

```
$ gem install edi4r-tdid # Zusatzpaket für alle TDIDs
```

- Online-Dokumentation von “edi4r”
 - Wie bei allen gems: Per Gem-Server & Browser abrufbar

```
$ gem_server # Startet Web-Server auf Port 8808
```

```
Browser: URL = "http://localhost:8808/"
```

- Web-Adressen:

<http://edi4r.rubyforge.org/>

Zugang zur Doku im Web

<http://www.rubyforge.org/projects/edi4r>

Projekt-Homepage



```
require "rubygems"
```

```
require_gem "edi4r"
```

```
require "edi4r/edifact"
```

- “edi4r” als normales Ruby-Gem laden
 - EDIFACT-Methoden bei Bedarf (meistens!) mit der letzten Zeile hinzuladen.
 - Unterstützung für EDIFACT SV1-3 und TDIDs D.96A, D.01B.

```
require_gem "edi4r-tdid"
```

- Zusatz-Gem “edi4r-tdid” liefert Zugriff auf alle TDIDs (90.2-D.07A)

```
ic = EDI::E::Interchange.new( :charset => 'UNOC' )
```

- Interchange anlegen → Festlegung von:
 - Syntax-Version (1-4, Default: 3)
 - Trennzeichen (UNA, Default (SV1-3): **UNA:+.?'**, Default (SV4): **UNA:+.?'***)
 - Zeichensatz (Default: UNOB) und Batch/Interactive-Betrieb (z.Z. nur “batch”)

```
puts "Unmoeglich!" if ic.charset == 'UNOA'
```

- charset: Getter für den Zeichensatz; kann nachträglich nicht verändert werden.



```
ic.una.decimal_sign = ','; ic.show_una = true;
```

```
ic.una.to_s # "UNA:+,? '"
```

- Sonderzeichen umdefinieren, hier: Dezimalkomma statt -punkt; UNA ausgeben!

```
ic.output_mode = :linebreak # auch: :indented, Default: :verbatim
```

- `ic.to_s` liefert 1 Segment pro Zeile (:indented = zusätzlich eingerückt)

```
msg = ic.new_message
```

- Neue, noch leere Nachricht von `ic` ableiten (erbt UNA, SV, Zeichensatz) → Festlegung von:

- ... zugrunde liegendem UN/TDID der Nachricht, Default: D.96A

- ... Nachrichtentyp, Default: ORDERS

- ... Subset u.a. (optional)

- ... ID der Nachricht, Default: (automat. generiert)

- ... allen erlaubten Segmenten (incl. Wh) und Segmentgruppen

```
bgm, dtm, nad, uns =
```

```
%w(BGM DTM NAD UNS).each { |tag| msg.new_segment(tag) }
```

- Neue, noch leere Segmente von `msg` ableiten (erbt UNA, SV, Zeichensatz) →
 - Festlegung der enthaltenen DE und CDE, incl. Reihenfolge, Status (M/C) und Wh.



```
unh = msg.header; unt = msg.trailer
```

- Service-Segmente UNH, UNT werden automatisch angelegt und sind bei Bedarf über die generischen Getter-Methoden `header` bzw. `trailer` erreichbar

```
cde = bgm.cC002; cde.d1001 = 220; cde.dxxxx = ...
```

```
bgm.d1004 = '1234567'
```

- Befüllung enthaltener Datenelemente, auch solcher in CDE →
 - Getter- und Settermethoden mit dem Namen bzw. Tag des EDIFACT-Objekts und dem Präfix **d**, **c** oder **s** (für DE, CDE, Segment) stehen automatisch zur Verfügung.
 - Bei mehrfachem Zugriff auf DE eines CDE lohnt sich das “Merken” des CDE
- `dtm`, `uns`: Analog befüllen!

```
nad.cC058.a3124[0].value = 'ABC GmbH'
```

```
nad.cC058.a3124[1].value = 'Vertriebsabteilung'
```

- Sonderfall “sich wiederholende DE” →
 - Der Getter (hier: `a3124`) liefert ein Array von DE-Objekten!
 - Zugriff per Array-Index liefert DE selbst. DE-Inhalt lesen/ändern mit `value/value=`



EDIFACT: *Interchange*-Bildung mit edi4r



```
[bgm, dtm, nad, uns].each {|seg| msg.add(seg) }
```

- Segmente an Nachricht anhängen (stets vor 'trailer')

```
ic.add(msg)
```

```
ic.add(msg, false)
```

- Nachricht an Interchange anhängen (stets vor 'trailer')
 - Hier wird per Default validiert, ggf. mit 'false' abschalten!

```
bgm.to_s; cde.to_s; msg.to_s; ic.to_s
```

- EDI-Objekte unterstützen to_s. Ihre String-Darstellung beachtet Escaping und Trennzeichen!

```
print ic
```

```
$stdout.write ic
```

- Wegen der impliziten Verwendung von to_s ist die Ausgabe von Interchanges sehr einfach!

```
print ic.inspect, bgm.inspect
```

- Detaillierte Anzeige der Inhalte, incl. leerer Bestandteile → Debugging!

```
ic.validate
```

- Validierung des gesamten Interchanges – unbedingt durchführen!



EDIFACT: *Interchange*-Bildung mit edi4r



```
unb = ic.header; unz = bgm.root.trailer
```

- Service-Segmente UNB, UNZ des Interchanges über die generischen Getter-Methoden `header` bzw. `trailer` von `ic` erreichbar.
Jede Komponente (hier z.B. Segment `bgm`) gestattet Zugriff auf ihre Interchange-Umgebung über den Getter `root`.

```
bgm.status == 'C' ; bgm.required? ; bgm.level == 0
```

```
cde = bgm.cC002; cde.name == 'C002'
```

- Hilfreiche Methoden für Segmente, CDE, DE →
 - **status**: 'M', 'C', etc. Verwandt: `required?` → true / false
 - **level**: Hierarchielevel des Segments: 0, 1, 2, ...
 - **name**: Name des CDE, DE oder Segments
 - **parent, root**: Eltern-Objekt (z.B. Segment des CDE), Interchange
Rot: Stammdaten erst nach Validierung verfügbar bzw. gefüllt!

```
nad.sg_name ; nad.is_tnode?
```

- Bei Segmenten: Name der Segmentgruppe oder nil; Trigger-Segment?

```
de = bgm['1004'].first # Liefert ein EDI::E::DE-Objekt
```

```
de.format, de.value
```

- Hilfreiche Methoden für Datenelemente →
 - **format**: 'an..35', 'n8', etc. **value**: Wert des DE setzen/abfragen.



EDIFACT-*Subsets*, insb. EANCOM

Kritik an “*plain* EDIFACT”

Unterschiede Subsets / EDIFACT

Kritische Betrachtung der *Subset*-Bildung

EANCOM - ein sehr erfolgreiches *Subset*



Kritik an “*plain* EDIFACT”



- Einerseits: Zu viel Freiraum
 - Von Nachrichtentypen über Segmente / SG bis zu Codes:
Zu viel für einzelne Branchen
 - Mehrdeutigkeiten in der Darstellung
 - Bsp: Datums-/Zeit-Angaben, "Belegdatum, Bestelldatum, ...",
Formatvarianten
 - Bsp: Codes für Preise und Beträge
Ausgewählte Code-Listen und deren Vielfalt zeigen, jeder Teilnehmer soll selbst einen Code für Rechnungsendbetrag, zu versteuerndem Betrag, Netto-Betrag und Summe der Positionen vorschlagen.
 - Dennoch: Manche Codes fehlen noch immer
 - Bsp: Euro-Paletten Typ ..., besondere Belegtypen, ...
 - Lange Entwicklungszyklen bei UN/EDIFACT erschweren einheitliche Erweiterungen
 - Daher Reaktion einzelner Branchen:
 - Bildung von Untermengen ("**Subsets**") auf Basis bestimmter UNTDID/EDIFACT-*Releases*.
-



- Nachrichtentypen
 - Nur ein Teil des EDMD wird in das *Subset* aufgenommen
 - Achtung: Auch die Aufnahme von *non-standard* Nachrichtentypen bzw. Status-0-Typen kann vorkommen!
 - Beispiel: INSDDES (EANCOM '97)
 - An Konsequenzen für Konverter-Software denken
 - “*Subset*” - wirklich noch eine “Untermenge”?
 - Muß/Kann-Regelungen können beim Subset auf allen Ebenen abweichen.
 - Typisch: $C \rightarrow N$ (*not used*) in vielen Fällen, $C \rightarrow M$ gelegentlich, aber: $M \rightarrow C$ oder N zu vermeiden
- Wiederholfaktoren
 - Können sich deutlich ändern, in beide Richtungen
- Abhängigkeitsvermerke
 - Siehe auch SV4. Schon vorher verwendet, aber uneinheitlich.



- Codelisten
 - Erhebliche Unterschiede!
 - Einerseits deutliche Verminderung, andererseits Definition *subset*-spezifischer Codes.
 - Platzhalter-Codes im *Subset*, durch offizielle (aber meist andere Codes) eines späteren Standard EDIFACT-Release zu ersetzen.
 - Beispiel: Euro-Vollpalette in EANCOM '97 und '02
- Der systematische Umgang mit *non-standard* Codes
 - DE 3055 etc. - wenn schon, dann sauber qualifizieren, incl. Urheber
 - Unterscheide drei Stufen
 - Standard-Codes,
 - *Subset*-Codes,
 - bilaterale Codes



- Besonderheiten
 - UNA: Forderung von EANCOM: "M" wenn nicht UNOA, damit Umgehung der UNOB-Standardtrennzeichen IS1, IS3, IS4.
 - CONTRL: EANCOM '97 behandelt dies als normalen Nachrichtentypen, nicht als Teil der Syntax wie UN/EDIFACT
 - Standardkonforme Kennzeichnung von *Subsets*
 - In UNH, DE 0057 in CDE S009 (*association assigned code*)
 - Siehe auch die neue Regelung in SV4
 - "***Restricted code lists***":
 - Kontextabhängige Definition zulässiger Teilmengen spezieller Subset-Codelisten.
 - Je nach Verwendungsort eines DE schränkt das *Subset* die zulässigen Codes stark ein.
 - Dokumentation nicht über *Directory*, sondern MIG-artig (siehe dort)
 - EANCOM-Beispiel: (mit '*' markierte Codes ...)



Kritische Betrachtung der *Subset*-Bildung



- Probleme, Fehlentwicklungen:
 - Keine Koordination zwischen Branchen
 - Teilweise nicht ausreichender Sachverstand
 - Ableitung der *Subsets* von unterschiedlichen *Directories*
- Die Folgen:
 - Der vielfach beklagte "*Subset*-Wildwuchs" in den 1990er Jahren - zahlreiche, hoch redundante aber dennoch inkompatible *Subsets*
 - Denken in Einfluss-Sphären behinderte gemeinsamen Erfolg
 - Defizite einzelner *Subsets* wurden EDIFACT insgesamt angelastet



EANCOM - ein sehr erfolgreiches *Subset*



- EANCOM als "Paradebeispiel" eines gelungenen *Subsets* mit Erfolg am Markt
- EANCOM-Hintergrund
 - Motivation, Urheber
 - Zeitliche Entwicklung, *Releases*
 - Nur vier *Releases* ('90, '94, '97, 2002),
aufbauend auf: 90.1; D.93A, D.96A, D.01B
 - Stabiler als EDIFACT, höhere Investitionssicherheit
 - Dokumentation
 - Sehr gut, leicht erhältlich über die lokale EAN- bzw. GS1-Organisation
- Überleitung zum Übungsteil:
 - Sichten der Dokumentation am Beispiel INVOIC



Interchange agreements und MIGs

***MIG: Message Implementation
Guideline***

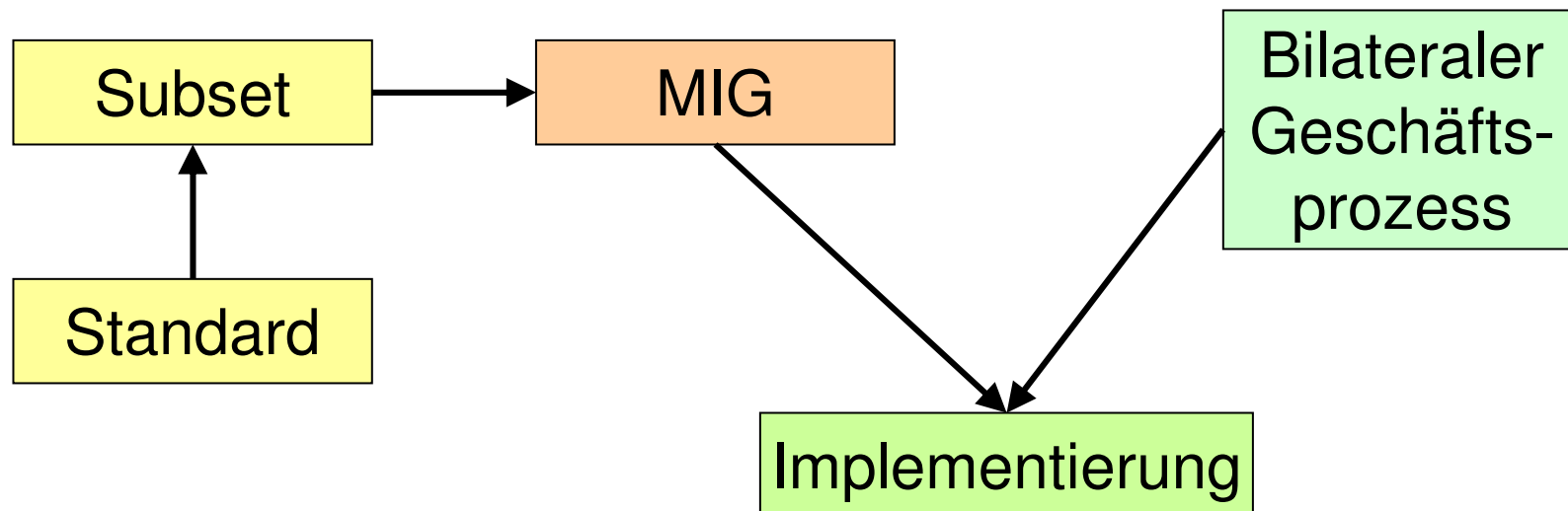
(manchmal auch: “*Guide*”)



- Zur konkreten Implementierung einer EDI-Partnerverbindung sind auch die *Subsets* noch nicht spezifisch genug.
- Es fehlen:
 - die konkreten Festlegungen und Erwartungen an die *Messaging*-Ebene, an die UNB-Inhalte, an organisatorische Regelungen wie z.B. Kontaktadressen im Störfall.
 - Diese werden in interchange agreements (**IAs**) festgehalten.
 - Manchmal nehmen sie auch die Form technischer Anhänge von EDI-Verträgen an.
 - die Einschränkungen / Beschreibungen / besonderen Anforderungen des unterstützten Geschäftsprozesses.
 - In sogenannten message implementation guidelines (**MIGs**) werden diese Einzelheiten festgehalten.
 - Typischerweise bestehen sie aus erläuternden Freitext-Abschnitten sowie einer Dokumentation jeder möglicherweise vorkommenden bzw. unterstützten Instanz der Segmente / SG des festzulegenden Nachrichtentyps.



- Der Weg zur konkreten Implementierung
 - Erst mit dieser genauen Beschreibung der zu erwartenden / erwarteten Daten ist eine konkrete Implementierung möglich.
 - Selbst dann muss oftmals noch intern recherchiert werden, welche der manchmal zahlreichen Sonderfälle eines MIG auf die konkrete EDI-Beziehung zutrifft:





- Merke:
 - Bei ausgehenden Daten hilft ein MIG des Geschäftspartners, noch vor der Implementierungsphase zu erkennen, welche *features* evtl. noch in der Applikationsschnittstelle fehlen. Derartige Lücken können im Idealfall parallel zu den *Mapping*- und *Messaging*-Aktivitäten geschlossen werden.
 - Bei eingehenden Daten ist die Versuchung groß, ohne MIG auszukommen und sich nur auf Beispieldaten des Geschäftspartners zu verlassen.
 - **Risiko:** Selten genutzte Nachrichtenteile können erst nach Produktionsstart auftauchen und führen dann zu Notmaßnahmen unter hohem Zeitdruck, oder bleiben gar unerkannt und verursachen Folgeschäden in der Geschäftsbeziehung.
 - **Beispiel:** Ignorierte Zahlungsbedingungen in ORDERS



Übungen

UN/EDIFACT-Beispiele

Subsets: EANCOM

MIG: Fall Metro Österreich



UN/EDIFACT-Beispiele: ORDERS

Vergleich: Papier (OV-Folien) - EDI:
Fa. Rossmann, Fa. Budnikowski
“EDI-Leseübung”: Otto Versand



Beispiel ORDERS, Rossmann



UNA:+.? '
UNB+UNOA:2+4305615006017:14+400
8888000002:14+020411:0757+53'
UNH+1+ORDERS:D:93A:UN:EAN007'
BGM+220+25935'
DTM+137:20020411:102'
DTM+2:20020416:102'
NAD+SU+4008888000002::9'
NAD+DP+4305615006017::9'
NAD+BY+4305615006017::9'
LIN+1++3014260004873:EN'
QTY+21:1296'
LIN+2++3014260000318:EN'
QTY+21:576'
LIN+3++4103330032461:EN'
... (usw.) ...

LIN+25++3014260257187:EN'
QTY+21:1152'
LIN+26++3014260258276:EN'
QTY+21:252'
LIN+27++3014260236199:EN'
QTY+21:1152'
LIN+28++3014260227043:EN'
QTY+21:5760'
LIN+29++4103330031860:EN'
QTY+21:864'
LIN+30++4103330031761:EN'
QTY+21:1728'
UNS+S'
CNT+2:30'
UNT+70+1'
UNZ+1+53'



Beispiel ORDERS, Budni



UNB+UNOA:2+4305175000005:14:4305175
000005+4008888000002:14:4008888000
002+020415:1446+189++ORDERS+++E
ANCOM'

UNH+0001+ORDERS:D:96A:UN:EAN008'

BGM+220:9+126049+9'

DTM+137:20020415:102'

DTM+2:20020422:102'

DTM+4:20020415:102'

NAD+SU+4008888000002::9'

NAD+BY+4305175009008::9'

NAD+DP+4305175009008::9'

LIN+1++3014260610890:EN'

QTY+21:27'

LIN+2++3014260611248:EN'

QTY+21:24'

LIN+3++3014260606725:EN'

QTY+21:24'

... (usw.) ...

LIN+32++5010622014786:EN'

QTY+21:1'

LIN+33++3014260666217:EN'

QTY+21:4'

LIN+34++3014260666200:EN'

QTY+21:4'

LIN+35++3014260658960:EN'

QTY+21:1'

LIN+36++4210201239239:EN'

QTY+21:9'

LIN+37++4210201210658:EN'

QTY+21:30'

LIN+38++4210201213062:EN'

QTY+21:4'

UNS+S'

UNT+86+0001'

UNZ+1+189'



Beispiel ORDERS, Otto



UNB+UNOA:2+4029686000001:14+
4008888100009:14+020417:1124+
45733'

UNH+10190354591334+
ORDERS:D:96A:UN:EAN008'

BGM+220+11119'

DTM+137:20020417:102'

DTM+2:20020423:102'

RFF+CR:105'

NAD+SU+4008888100009::9'

RFF+API:13679'

NAD+BY+4029686000001::9'

CTA+PD+679'

COM+040/0406461-499:TE'

COM+040/0406464-499:FX'

NAD+DP+4029686000193::9'

CUX+2:EUR:9'

PAT+22++5:3:D:30'

PCD+12:3'

PAT+22++5:3:D:60'

TDT+20++30'

TOD+++CIP'

LOC+1+14::92:LIEFERANSCHRIFT'

LIN+1++4210201298229:EN'

PIA+1+343307:BP+SILK EPIL ER
1250:SA'

IMD+F+ANM+::92:SILK EPIL ER 1250'

IMD+C+TPE+::92:HERZ343307'

QTY+21:400:PCE'

PRI+AAA:35.28'

PAC+++CT+X:51/98/0'

UNS+S'

MOA+86:14112'

CNT+1:400'

CNT+2:1'

UNT+31+10190354591334'

UNZ+1+45733'



- *Online*-Dokumentation: INVOIC
 - *Branching Diagram*
 - Analyse:
 - Systematische Besprechung der Bestandteile des Dokuments,
 - Strukturierung des zunächst sehr komplex aussehenden Gebildes,
 - Modularität und Systematik von UN/EDIFACT in Aktion
 - Einübung des Umgangs mit EDIFACT
 - Wichtige Spezialfälle am Beispiel kennenlernen
 - *Restricted code lists*
 - *Sub-types*: 380, 381, 393
 - Blick auf den betriebswirtschaftlichen Hintergrund schärfen - Umsetzung in einem konkreten Fall
 - Anforderungs-Vielfalt: Die GS1-Tabelle der Partnerspezifika
→ AK EDI Handel.



- Online-Dokumentation: INVOIC
 - *Interchange agreement*-Teil (UNA, UNB, UNZ)
 - Allgemeine Absprachen zu INVOIC
 - Analyse (Beispielbefunde):
 - Einschränkungen gegenüber Subset (Spalte “ME”)
 - Bsp. Längenbeschränkung Rechnungsnummer
 - Anforderungen an Kopf-, Positions- und Summenteil
 - Kundenstamm-bezogene Informationen zu NAD
 - Unterpositionen im LIN-Segment
 - Geschäftsprozessbezogene Angaben
 - Nettoabrechnungsverfahren
 - Sammelrechnungsliste
- Vergleich einer konkreten EDI- und Papierrechnung



Papier-Beispielrechnung Metro AT



- (auf separaten OV-Folien)
- parallel zu EDI (per Beamer/Display) betrachten!
- **Bemerkung:**
 - Daten aus Testsystem,
 - alle Angaben und insbesondere **Preise fiktiv** !
 - **Stammdaten** sind nicht für Veröffentlichung bestimmt -
also: **Nicht aktiv verbreiten!**



EDI-Beispielrechnung Metro AT



UNA:+.? '
UNB+UNOC:3+9004116000016:14+90
00480000003:14+020423:1243+
8++++++1 '
UNH+2+INVOIC:D:96A:UN:EAN008 '
BGM+380+0141078211+9 '
DTM+137:20020422:102 '
DTM+35:20020422:102 '
RFF+ZZZ:18438537 '
RFF+DQ:0026156112 '
DTM+171:20020422:102 '
NAD+SU+9004116000016:::9 '
RFF+VA:ATU14664200 '
NAD+BY+9000480000126:::9 '
RFF+API:1182234 '
NAD+DP+9000480000126:::9 '
RFF+IT:1182234 '
NAD+IV+9000480000126:::9 '
RFF+API:1182234 '
TAX+7+VAT+++:::20 '

CUX+2:EUR:4 '
PAT+3 '
DTM+12:20020522:102 '
PCD+12:3.000 '
PAT+3 '
DTM+13:20020621:102 '
LIN+10++5000394048119:EN'
PIA+1+15064444:SA::91 '
IMD+F++:::FK2002-SB2A WORLD CUP '
IMD+C++IN '
IMD+C++SG '
QTY+47:1:PCE '
MOA+66:1008 '
PRI+AAA:1008:::1:PCE '
LIN+20++5000394037922:EN+1:10'
PIA+1+15034745:SA::91 '
IMD+F++:::DURACELL Ultra M3
MN1500 K4 (AA) '
IMD+C++CU '
QTY+59:100:PCE '
PRI+AAA:3.2:::1:PCE '



EDI-Beispielrechnung Metro AT



LIN+30++5000394037946:EN+1:10'

PIA+1+15034954:SA::91'

IMD+F++:::DURACELL Ultra M3
MN2400 K4 (AAA)'

IMD+C++CU'

QTY+59:60:PCE'

PRI+AAA:3.92:::1:PCE'

LIN+40++5000394038011:EN+1:10'

PIA+1+15034968:SA::91'

IMD+F++:::DURACELL Ultra M3 MN
1400/LR14 C-Ba'

IMD+C++CU'

QTY+59:50:PCE'

PRI+AAA:2.83:::1:PCE'

LIN+50++5000394038486:EN+1:10'

PIA+1+15034976:SA::91'

IMD+F++:::DURACELL Ultra M3
MN1300 K2 (D)'

IMD+C++CU'

QTY+59:20:PCE'

PRI+AAA:3.56:::1:PCE'

LIN+60++5000394038073:EN+1:10'

PIA+1+15034985:SA::91'

IMD+F++:::DURACELL Ultra M3
MN1604 K1 (9V)'

IMD+C++CU'

QTY+59:70:PCE'

PRI+AAA:3.43:::1:PCE'

UNS+S'

CNT+2:6'

MOA+86:1209.6'

MOA+79:1008'

MOA+176:201.6'

TAX+7+VAT+++:::20+S'

MOA+124:201.6'

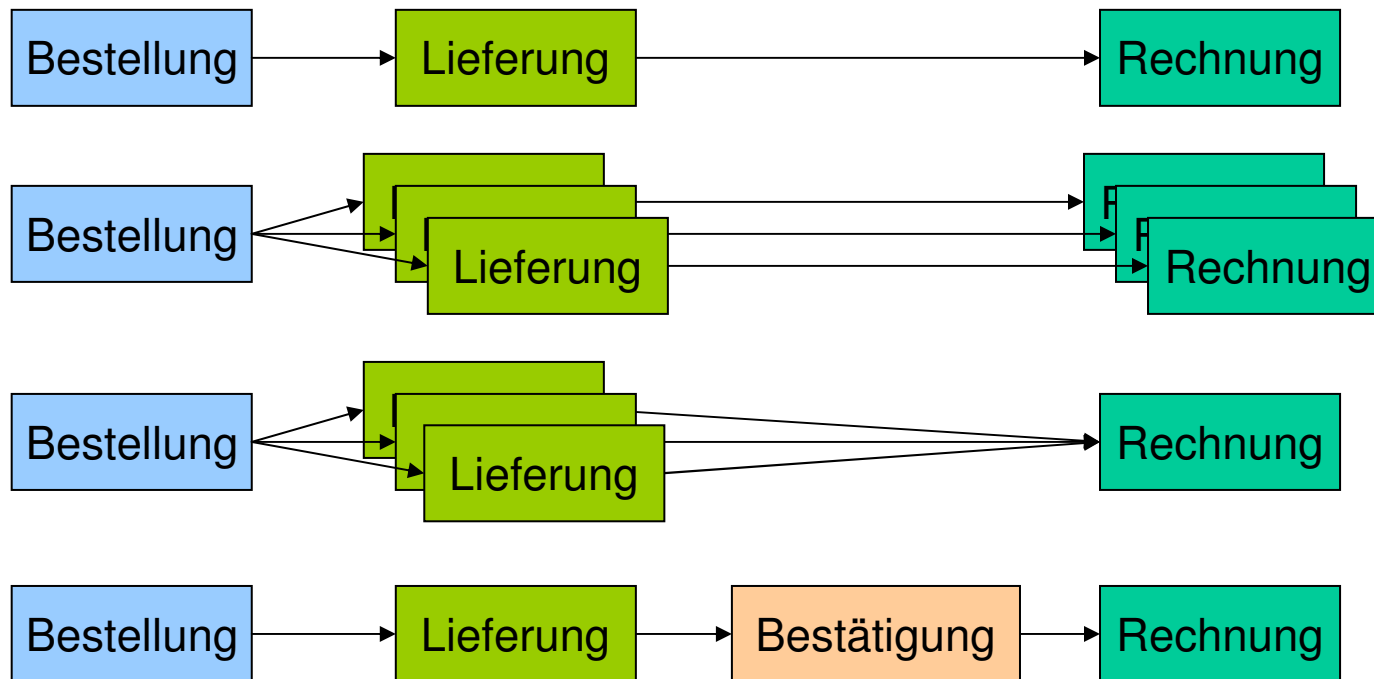
MOA+125:1008'

UNT+69+2'

UNZ+1+8'



- Grundlage / Hintergrund
 - ER-Modell zur Beziehung zwischen Belegarten
 - Geschäftsprozess-Sicht





- Mehrwertsteuer -
Behandlung verschiedener Steuersätze:
 - Belegkopf:
 - Nur Standardsatz TAX
 - Positionen:
 - Ggf. abweichender Satz TAX, opt. MOA
 - Summenteil:
 - Gesamtsteuerbetrag MOA+176
 - SG für Beträge pro Steuersatz TAX-MOA (n-fach)



- Zu- und Abschläge - die ALC-Gruppe
 - Arten / verwendetes Segment
 - Prozentsätze PCD
 - Raten RTE
 - Beträge MOA
 - Die Kalkulationsstufe
 - Bsp.: $(100 \text{ €} - 5\%) - 2\text{€} = 93 \text{ €}$, $(100 \text{ €} - 2\text{€}) - 5\% = 93,10 \text{ €}$ (!)
 - Beispiele für Kopfkonditionen
 - Frachtgebühr
 - Zentrallagerrabatt
 - Sonderskonto
 - Beispiele für Konditionen auf Positionsebene
 - diverse Rabatte
 - Mindermengenzuschlag, Entsorgungsgebühr



- Kalkulationsverfahren
 - Netto MOA+66, PRI+AAA
 - Brutto MOA+203, PRI+AAB, ALC...
 - Problemfall Rundungsdifferenzen
 - Nettopreisberechnung, Rundung, Betragsberechnung vs.
 - Bruttopreis, Zu/Abschläge, Nettobetragsberechnung, Rundung