



# EDI

Electronic  
Data  
Interchange  
(Elektronischer Datenaustausch)



## Klassisches EDI - der Kern

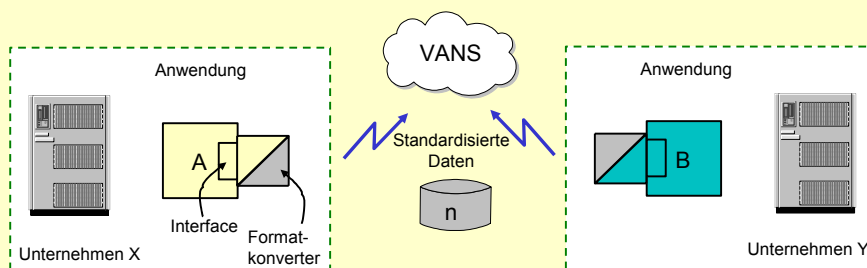
Einleitung - die Kernkomponenten  
File Transfer- und Messaging-Standards  
UN/EDIFACT und EANCOM im Detail  
Applikationsschnittstellen  
Konverter- und Mappingtechniken



- Applikationsschnittstellen
- EDI-Standardaustauschformate
- Mapping
- Routing
- Messaging / File Transfer
- Extras
  - Archivierung
  - Reporting
  - Alarmierung
  - Tracking & Tracing



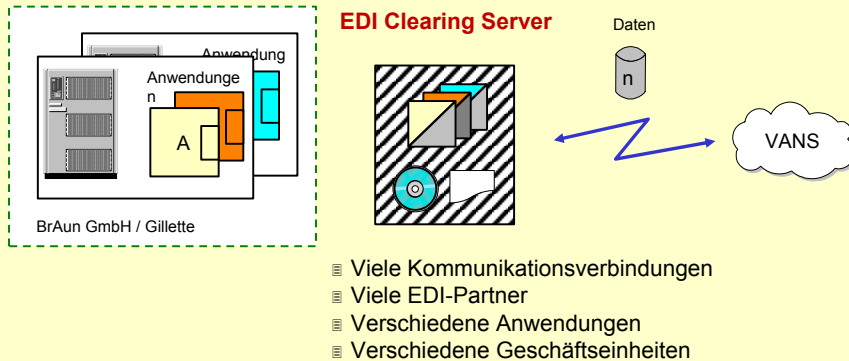
## Front-End Konzept (veraltet)



- Viele Kommunikationsverbindungen
- Viele EDI-Partner



## Der EDI-Server - Eine zentrale IT-Ressource



## File Transfer- und Messaging-Standards

File transfer - messaging - mailboxing  
Automatisierungshürden  
Topologien für den Datenaustausch



- **File transfer**

- Nur Inhalt wird transferiert
- Routing: Implizit, durch Verbindungsaufbau
- Beispiele:
  - OSI: FTAM
  - IP: ftp, http
  - ODETTE: OFTP

- **Messaging**

- Separate Header-Information, Inhalt als Body/Attachment(s)
- Routing: Per Adressierung (Sender, Empfänger)
- Beispiele:
  - OSI: X.400
  - IP: SMTP; aktuell: JMS
  - VANS: IBM-IE



- **Mailboxing**

- Nur Inhalt wird transferiert
- Nur bestimmte standardisierte Inhalte zulässig!
- Typisch für VANS (Value Added Network Services)
- Routing: Implizit, durch Sender / Empfänger-Codes im Header des Inhalts
- Beispiele:
  - VANS: EDI\*Express (GXS, weltweit)  
Tradanet (GXS, i.w. UK)  
DanNet (DanNet, Dänemark)  
Ecodex (IBM-IE, Österreich)
  - Innerhalb X.400:  
Telebox-400 (Telekom, Deutschland)  
Allegro (Allegro, Frankreich bzw. Spanien)



- ...oder: Warum reichen ftp oder http denn nicht?
  - ftp und http sind konzeptionell C/S-Protokolle mit einem manuell zu bedienenden “*client*”
  - *High-end* EDI-Anforderungen sind dagegen
    - vollautomatischer 7\*24 Std.-Betrieb
    - Kapazität für tausende Dateien pro Tag oder gar Stunde
  - Konsequenzen:  
Zahlreiche Zusatzanforderungen “um ftp/http herum”



- Zugriffsberechtigungen (*remote side*) für ftp
  - bilateral abzustimmen und einzurichten
- Konventionen für Datei- und Verzeichnisnamen
  - Absprachen notwendig
- Berücksichtigung von *cross-platform* Aspekten
  - Bsp: Unix-to-VMS, Win2000-to-AS/400
  - Binär/ASCII, ASCII/EBCDIC, besondere Zeichensätze
  - Unterstützte ftp-Kommandos?
- Lokale Pufferung
  - Übertragungsstörungen dürfen sendende Prozesse nicht blockieren
- Organisation mehrerer Austauschkanäle



- Serialisierung (FIFO)
  - Empfangsreihenfolge = Sendereihenfolge der Dateien!
  - Bsp.: Bestelländerung darf "ihre" Bestellung nicht überholen
- Wahrung der Eindeutigkeit
  - keine Datei doppelt senden
  - keine auslassen
  - kein Überschreiben durch andere Datei gleichen Namens
- Koordination mehrerer Quellen pro Kanal
  - Locking, gemeinsamer Server für Seriennr./ Dateinamen?
- Synchronisierung von Sender und Empfänger
  - "Atomare" Übergaberegeln verhindert Annahme von Fragmenten



- Störungs-Management
  - Entfernung der "Reste" nach Verbindungsabbruch?
  - Wiederholung der gesamten Übertragung notwendig, oder Wiederaufsetzen am Punkt des Abbruchs möglich?
  - Überwachung blockierender Serverprozesse (z.B. ftpd)
  - Automatischer Wiederanlauf nach temporären Störungen wie Netzwerkausfall
  - Warnung/Alarmierung bei persistenten Problemen
    - incl. Definition eines Schwellenwerts, evtl. pro Kanal



- Ablaufsteuerung
  - Batch: Übertragung nur zu bestimmten Zeiten
    - z.B. zur Bündelung, Last- und Kostenoptimierung
  - Event: Ereignisgesteuertes Auslösen von Aktionsketten
- Protokollierung
  - Insb. lückenloser Nachweis der erfolgten Zustellungen



## Topologien für den Datenaustausch

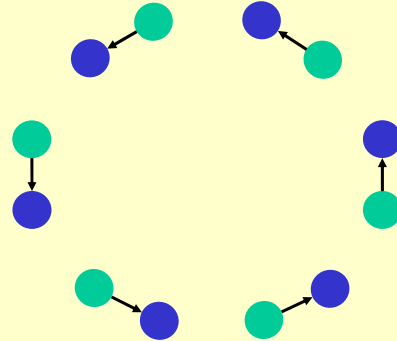
Point-to-point (P2P)  
P2P-Netzwerk  
Hub-and-spoke  
Hub-and-spoke plus Gateways  
Vernetzte C/S-Modelle



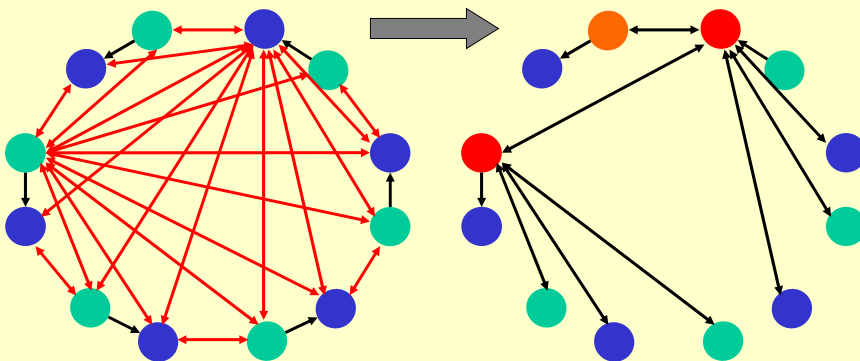
## Point-to-point Modell



- Typisch für Initialphase
  - Erste Schritte einfach
  - Bilaterale Abstimmung ohnehin noch die Regel
- Ideal für *File transfer*
  - Kein *Routing* notwendig
- Potentielle Probleme:
  - Verfügbarkeit der Partnersysteme
  - Standardisierung
  - *Routing*
  - Skalierung



## P2P-Netzwerke



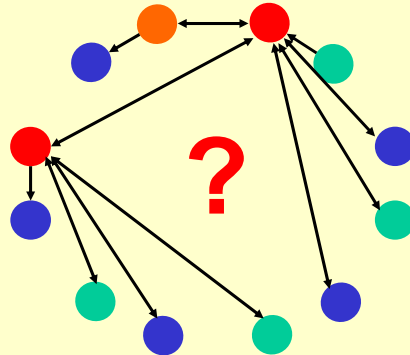




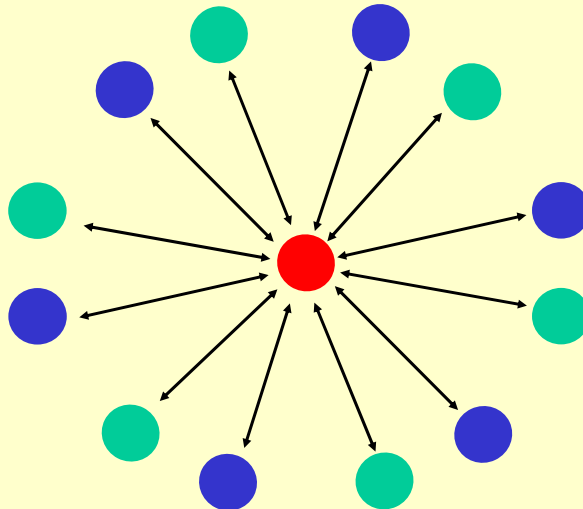
## Point-to-point Netzwerke



- Skalierungsproblem  
reduzierbar durch  
Spezialisierung
  - Randknoten, *clients*
  - Hauptknoten, *server*
  - Neue Aufgabe: *Routing*
- Probleme
  - Routingkriterien?
  - Verfügbarkeit?
  - Standardisierung?
  - Skalierung (große Systeme)
- Fazit:
  - Die meisten P2P-Probleme  
bleiben bestehen!



## Hub-and-spoke Modell

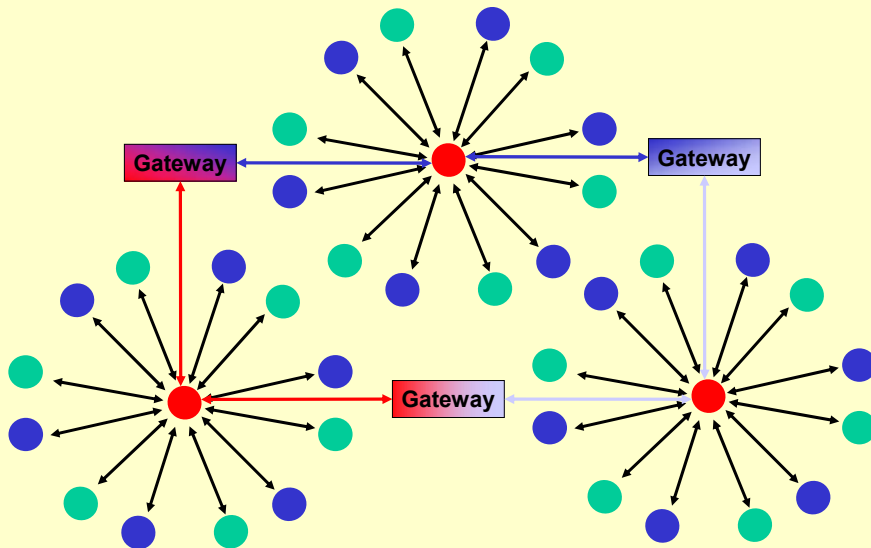




- Extremfall eines P2P-Netzwerks:
  - 1 Server ("Hub")
  - viele Clients ("Spokes")
- Die Grundidee der Value Added Network Services (VANS)
- Ideal für *Mailboxing*
- Löst das *Routing*-, Skalierungs- und Verfügbarkeitsproblem
- Standardisierung?
  - Nur per Industriestandard und/oder Marktbeherrschung
  - Proprietäre Zugriffstechniken, nicht für alle Plattformen verfügbar
- Kostenmodelle
  - Einmalige Setupkosten
  - Monatliche Fixkosten, z.B. pro Mailbox, Freivolumen
  - Variable Kosten, z.B. pro kB, "Sender zahlt", S/E 1:1, pro Zugriff, ...



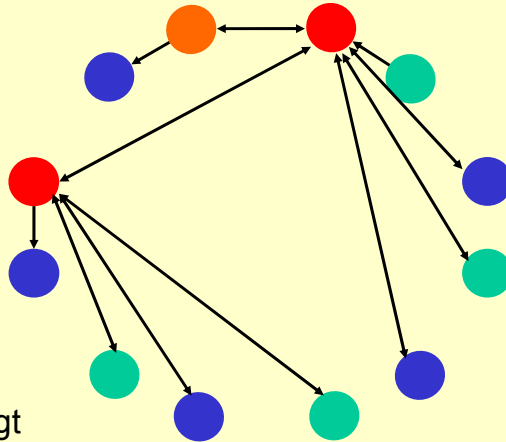
- Zusatzleistungen einiger VANS
  - Zustellungsnachweis
  - Berichte
    - Datenmengen
    - Einzelnachweise mit Zeiten
  - Andere Leistungen
    - Syntax-Check
    - Teils Konvertierung (Inter-Standard, Inter-Subset)
    - Überwachung von "closed user groups" (Bsp: Phönix)
    - Ablehnung von "Doppelten"
    - Zwischenspeicherung, Archivierung
    - Registrierung / Überwachung zugelassener Partnerkennungen
- Probleme:
  - Vernetzung verschiedener proprietärer "Inseln"??
  - Kosten, insb. bei erzwungener Mehrfach-Mitgliedschaft



- Vernetzung mehrere Hub-and-spoke-Inseln (z.B. VANS) über Gateways
  - Theoretisch von Vorteil:
    - Nur einmal Mitgliedschaft nötig
  - Praxis: "Coopetition"-Situation ungünstig
    - Gateways zwischen Wettbewerbern entstanden durch Kundendruck
    - Laufen den Interessen der Betreiber meist zuwider
    - Daher schlechter unterstützt als der Normalbetrieb im eigenen VANS
    - Aber: Meist problemlos möglich zwischen VANS *desselben* Betreibers
  - Größte Nachteile:
    - Informationsverlust an den Gateways, z.B. Zustellungsnachweis nur bis Gateway - wertlos!
    - Keine klare Verantwortlichkeitsregelung mehr
    - In der Praxis doch Rücksichtnahme auf / Kenntnis von Regeln anderer VANS notwendig
    - Teuer im Betrieb aufgrund abschreckender Tarifstruktur



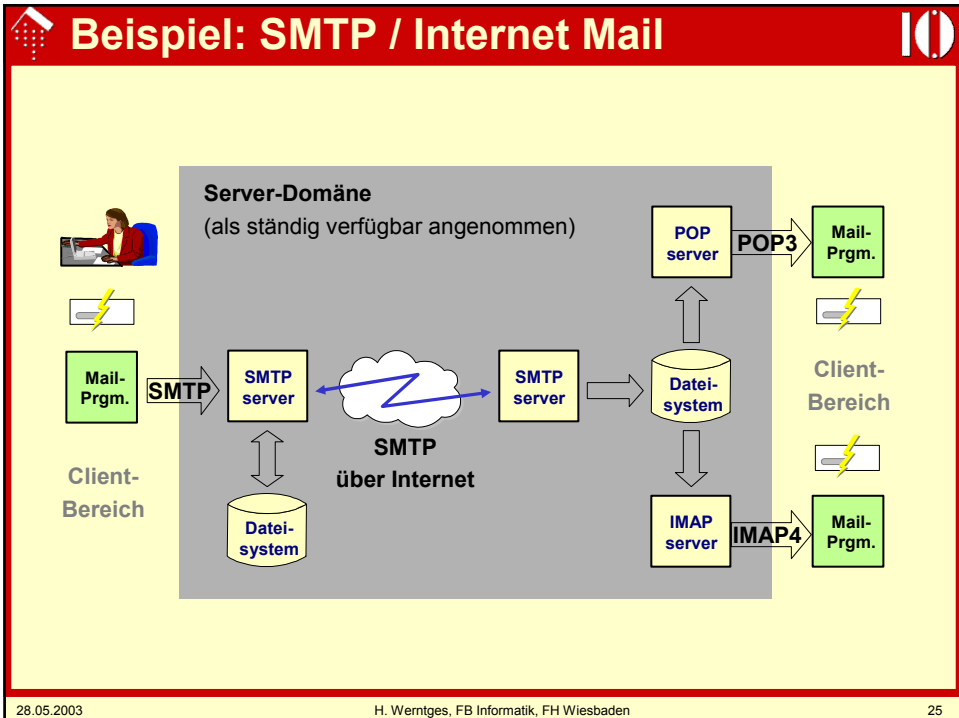
Vielleicht doch?



Wenn ja, was bringt den Erfolg?



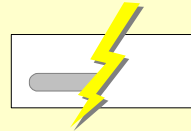
- Strenge Spezialisierung:
  - Entweder *C(lient)* oder *S(erver)*
- Konsequente Anforderungen an *Server*
  - Verfügbarkeit, Belastbarkeit, Vernetzbarkeit
- Lösung des Skalierungsproblems durch Protokoll-Standardisierung
  - zwischen *Clients* und *Server*
  - zwischen verschiedenen *Servern*
- Lösung des Routingproblems durch neue Ebene:
  - Pflege einer separaten Adressierungsebene seitens der Applikationen erforderlich
- Resultierendes Konzept:
  - *Messaging*
  - *Store-and-forward* Prinzip



- ## Begriffsbildung zu Messaging
- **MTA** - Message Transfer Agent
    - hier: Der SMTP Server
    - Beispiel: sendmail
  - **MS** - Message Store
    - hier: Das Dateisystem
    - Allgemein: Ein Subsystem für "Langzeit"-Speicherung
  - **UA** - User agent
    - hier: Das Mail-Programm
    - Beispiele:
      - Outlook Express, Netscape Messenger, elm, mail, Eudora, ...
    - Varianten:
      - **RUA** - Remote User Agent (heute der Normalfall)
      - **AU** - Access Unit, z.B. ein automatisches Mail-Gateway



# Einführung in X.400



Hintergrund  
Analyse  
Organisatorisches  
Vergleich X.400 - Internet Mail



# X.400: Hintergrund

Historische Entwicklung  
X.400 im OSI-Referenzmodell  
Grundlage: ASN.1



- Literatur:
  - B. Plattner, C. Lanz, H. Lubich, M. Müller and T. Walter, Elektronische Post und Datenkommunikation. X.400: Die Normen und ihre Anwendung. Bonn: Addison-Wesley, 1989
  - Th. Schmoll, Handelsverkehr, elektronisch, weltweit: Nachrichtenaustausch mit EDI/EDIFACT, Markt & Technik, München 1994
- WWW:
  - <http://www.alvestrand.no/x400/>
  - <http://www.oppenheimer-software.com/x400.html>
  - <http://www.dfn.de/service/x400/>



IFIP	1979	TC6 WG6.5 MHS (Forschung)
CCITT (später ITU-T)	1980	Beginn der MHS-Arbeiten
	1984	erste Empfehlung X.400-Serie MHS (sog. Rotbuch)
	1988	stark überarbeitete Version X.400-Serie MHS (sog. Blaubuch)
	1992	ergänzte Version
ISO	1980	Beginn der MOTIS-Arbeiten
	1986	Abstimmung zu MHS
	1988	IS 10021 MOTIS



- Vorgeschichte
- Organisatorisches Vorbild: "Gelbe Post"
- 1984
  - Für IPM ausgelegt: P2
- 1988
  - EDI-Besonderheiten standardisiert: P-EDI
  - IPM-Verbesserungen: P22
- 1992
  - Sicherheitsanforderungen ergänzt, z.B. X.509



- |   |                                |
|---|--------------------------------|
| • 7: Anwendungsschicht                    | • 7: <i>Application layer</i>  |
| • 6: Darstellungsschicht                  | • 6: <i>Presentation layer</i> |
| • 5: Kommunikations-<br>steuerungsschicht | • 5: <i>Session layer</i>      |
| • 4: Transportschicht                     | • 4: <i>Transport layer</i>    |
| • 3: Vermittlungsschicht                  | • 3: <i>Network layer</i>      |
| • 2: Sicherungsschicht                    | • 2: <i>Data link layer</i>    |
| • 1: Bitübertragungs-<br>schicht          | • 1: <i>Physical layer</i>     |
| • 1-4: Transportdienste                   |                                |
| • 5-7: Anwendungs-D.                      |                                |





- Das OSI-Referenzmodell erschien 1984.
- Es hatte große ordnende Wirkung auf die weitere Entwicklung der Datenkommunikation.
- OSI-Protokolle und -Normen sind heute praktisch ohne Bedeutung, aber das OSI-Referenzmodell wird auch heute noch vielfältig zitiert.
- X.400: Eine "echte" Schicht-7 OSI-Protokollfamilie, einer der wenigen noch nicht von der "IP-Welt" verdrängten OSI-Standards.
- Quelle: Stöttinger, K.-H., Das OSI-Referenzmodell, Bergheim 1989



		Anwendungsdienste				Datenaustauschformate	
						EDIFACT	ODA
7.2	X.400	FTAM	VTP	TP	X.500	ODIF	
7.1	ACSE		RTSE	ROSE		CCR	

### Schicht 7.1: Service Elements

- ACSE: Association Control SE
- ROSE: Remote Operations SE
- RTSE: Reliable Transfer SE
- CCR: Commitment, Concurrency, Recovery

### Schicht 7.2: Application Service Elements

- X.400 Electronic mail
- FTAM File Transfer, Access, Management
- VTP Virtual Terminal
- TP Transaction Processing
- X.500 Directory System



- ODA - Office Document Architecture
  - ISO 8613, für Briefe, Memos, Berichte, ...
  - Logische Struktur
  - Layoutstruktur
  - Inhalt
- ODIF - O.D. Interchange Format
  - Für elektronischen Austausch von ODA-Dokumenten
  - Begrifflich schwer von EDI(FACT) zu trennen

ODA und ODIF sind heute ohne große Bedeutung für EDI, besitzen aber gemeinsame Ursprünge



- ASN.1 - eine formale Beschreibung von Daten für Telekommunikations-Protokolle, unabhängig von
  - Implementierungssprachen
  - Plattformen, Applikationen
  - Details des physischen Datenaustauschs
  - Konzeptionelle Parallelen zu XML Schemas, SOAP; (MIME)
- Standardisiert & bewährt seit 1984, aktuelles Release: 1997
- Typisierung:
  - Basistypen wie `int`, `boolean`, `char strings`, `bit strings`, ...
  - Konstrukte: `structure`, `list`, `choice`, ...
- Kombinierbar mit verschiedenen "encoding rules" wie z.B. PER (Packet encoding rules) - "Kompressionsstandard" (!)
- X.400-Protokolle basieren auf ASN.1 (!)

Quellen: <http://www.asn1.org>



# X.400: Analyse

Aufbau einer X.400 Mail  
Komponenten eines MHS  
Die Protokolle



## Aufbau einer X.400 Mail



- Generelle Struktur:
  - Umschlag
  - Inhalt
    - Kopf
    - Rumpf
- Umschlag (*envelope*)
  - Absenderadresse (*Originator*)
  - Empfängeradressen (*Recipients*)
  - Dringlichkeit (*Priority*)
  - Poststempel der MTA's auf dem Übertragungsweg (*Trace information*)
  - Netzweit eindeutige Kennzeichnung (*MPDU-ID*)
  - Angaben zur Art des Inhaltes (*Content type* und *EIT (Encoded Information Types)*)



- Inhalt (*Content*)
  - Kopf (*Header*)
    - Absenderadresse (*Originator*)
    - Adressen der Bevollmächtigenden (*AuthorizingUsers*)
    - Hauptempfängeradressen (*PrimaryRecipients*)
    - Empfängeradressen für Kopien (*CopyRecipients*)
    - Betreff (*Subject*)
    - Wichtigkeit (*Importance*)
    - Vertraulichkeit (*Sensitivity*)
    - Antwort an (*ReplyToUsers*)
    - Antwort erwartet bis (*ReplyBy*)
    - in Bezug auf (*CrossReferences*)
    - Verfallsdatum (*ExpiryDate*)
    - eindeutige Kennzeichnung (*IPM-ID*)
- [Forts.]



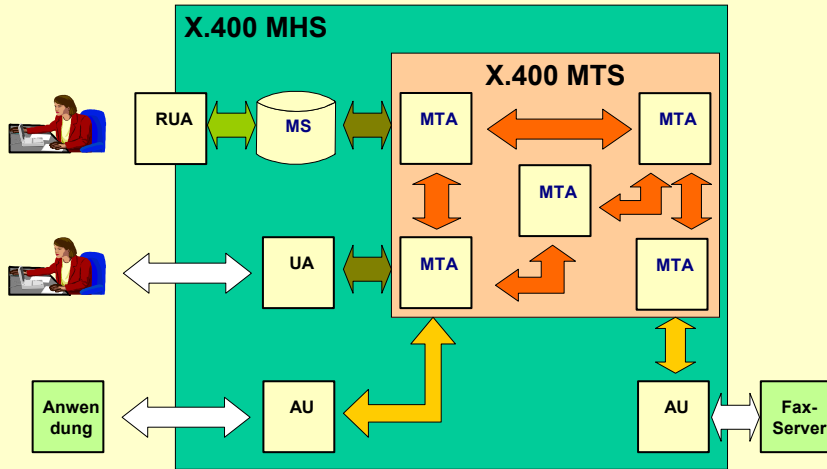
- Inhalt (*Content*) [Forts.]
  - Rumpf (*Body*)
    - *Body part 1 (type t[1]), ..., body part n, type t[n]*
- *Body part types* (Auswahl):
  - *a5text* (BP 0)
    - nur US-ASCII-Zeichen
  - *forwarded* oder *message* (BP 7)
    - der Inhalt einer anderen Mail mit *Header* und *Body*
  - *undefined* (auch binär, *bilaterally defined*) (BP 14)
    - im X.400(84) noch nicht erwähnt, aber in jeder Software nach X.400(84) möglich, d.h. sog. X.400(86)-Software ist X.400(84) mit Ergänzungen von 1986
  - *extended* (auch *externally defined*) (BP 15)
    - nur in Software ab X.400(88), sehr viele Untertypen; hauptsächlich mit Untertyp *Generaltext* benutzt (Text mit deutschen Umlauten)



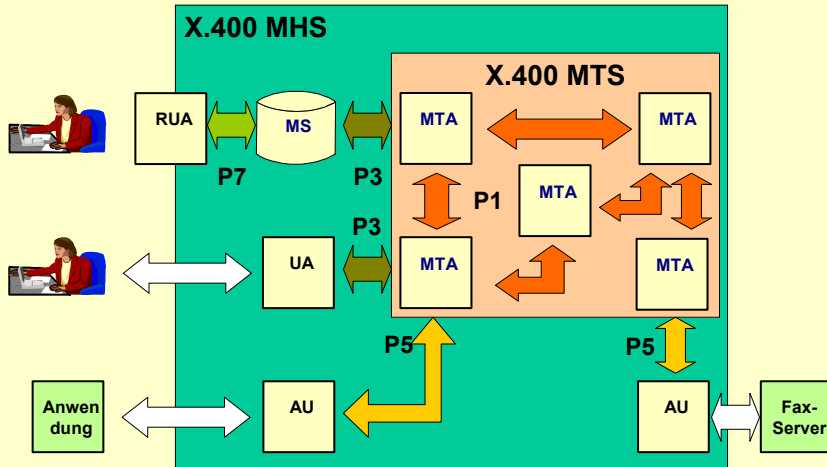
# Komponenten eines MHS



- Anwender bzw. Anwendung, nutzen:
- **MHS** - *Message Handling System*, bestehend aus:
  - **MTS** - *Message Transfer System*, ein Graph bestehend aus:
    - MTA - *Message Transfer Agents* als "Knoten"
    - Direkte Verbindungen zwischen einigen MTAs ("Kanten")
  - **MS** - *Message Stores*, für
    - geordnete Persistenzsicherung im Zusammenspiel mit RUAs
    - langfristige Aufbewahrung von Nachrichten
    - persönliche Mailboxen
  - **UA** - *User Agents*, die *Clients / User interfaces*
    - Variante: **RUA** - *Remote UA* (nicht immer *online*, heute üblich)
  - **AU** - *Access Units, Program interfaces* bzw. Dienstübergänge
    - Beispiele: Anbindung eines EDI-Konverters oder FAX-Servers
    - Variante: **PDAU** - *Physical Delivery AU* (Druck & Postzustellung)



## X.400: Die Protokolle



- P1
  - beschreibt den Aufbau der Informationsobjekte, die zwischen zwei MTA's ausgetauscht werden
- P3
  - beschreibt die Funktionen beim Informationsaustausch zwischen UA bzw. MS und MTS
- P5
  - beschreibt die Funktionen zwischen AU und MTS, z.B. Übergang zu Telex
- P7
  - das Zugriffprotokoll eines UA auf den MS



- P2
  - beschreibt den Aufbau der Informationsobjekte, die zwischen zwei UA's für Personen ausgetauscht werden, gemäß X.400(84)
- P22
  - wie P2, aber mit Ergänzungen aus X.400(88)
  - erweitert um Multimedia-Unterstützung ("MIME"-artig)
- P35, P-EDI, P<sub>edi</sub>
  - beschreibt den Aufbau der Informationsobjekte, die zwischen zwei AU's für Anwendungen ausgetauscht werden, gemäß X.400(88), mit Ergänzungen von 1992
  - explizite EDI-Ergänzung, parallel zu IPM
  - Wertet *envelope*-Informationen aus EDI-Standards aus (z.B. UNB)
  - Verbessert die Statusrückmeldungen über Gateways



## X.400: Organisatorische Aspekte

ADMDs, PRMDs

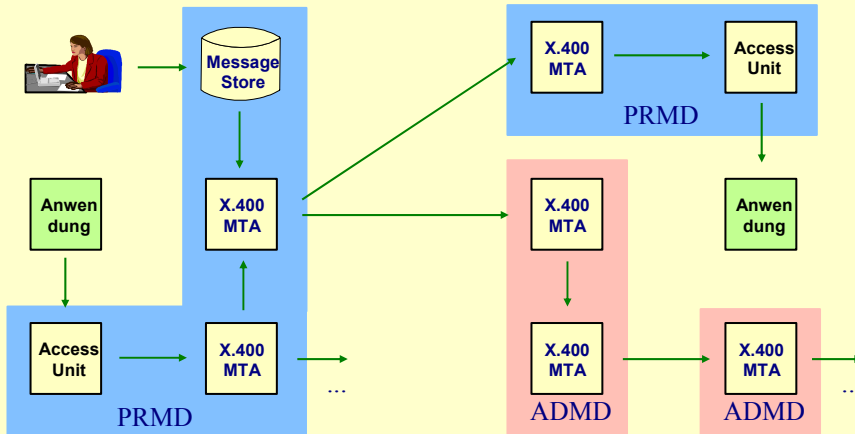
Regelung der Verantwortlichkeiten, Tracking,  
(N)DNs

O/R-Adressen: Aufbau, Parameter

MTA-Routing, Fallbacks

Gebühren (Beispiel)





- ADMDs bilden das Rückgrat des X.400 MHS
  - Ein MTA oder ein MTA-Netz können als *administrative domain* (ADMD) zertifiziert und zugelassen werden, wenn sie bestimmte Qualitätsmerkmale erfüllen.
  - ADMD-Dienstleistungen: Routing, Mailbox-Betrieb, Tracing, Gateway-Dienste
  - ADMDs sind untereinander direkt oder indirekt zu einem globalen MHS verbunden
- Kommerzielle Aspekte
  - ADMDs sind die "VANS" eines einzigen globalen X.400 MHS
  - Wegen der Standardisierung von X.400 herrscht Wettbewerb, analog z.B. zu privaten Paketdiensten
  - ADMD-Gebührenprinzip analog Porto: Sender zahlt alles
  - Schutzfunktion der Kosten: Spamming / Missbrauch werden für den Verursacher teuer!



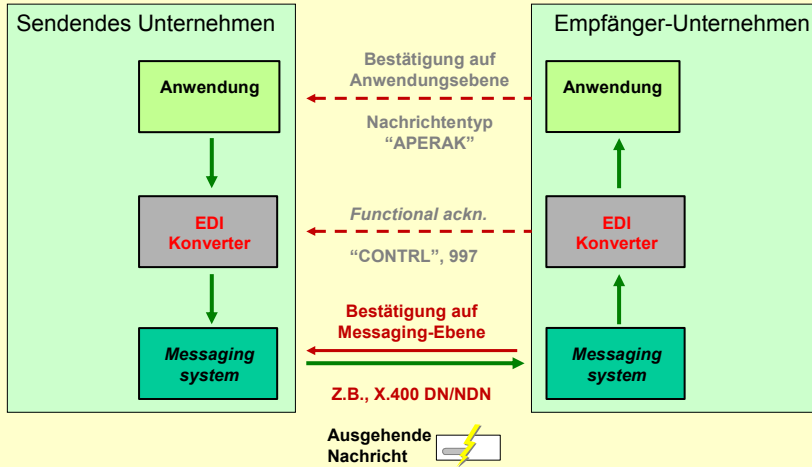
- PRMDs:
  - Private Organisationen können MTAs in eigener Regie betreiben und bilden dann PRMDs (*private management domains*)
  - Inseln: Es steht Anwendern frei, PRMD-Inseln ohne Verbindung zum globalen MHS zu bilden, z.B. als reine "Hauspost"
  - Eine PRMD darf nur an höchstens eine ADMD angebunden sein
- Direktverbindungen zwischen PRMDs
  - Direktverbindungen zwischen PRMDs (auch solcher, die an verschiedene ADMDs angebunden sind) sind zulässig
  - Vorteile
    - Schnellstmögliche Zustellung, ohne Umwege
    - Direkte Kontrolle über Erfolg
    - Nur Leitungskosten, keine ADMD-Gebühren
  - Nachteile
    - Einrichtung und Prüfung der Direktverbindungen
    - Ständige Verfügbarkeit der direkt angeschlossenen Partnersysteme



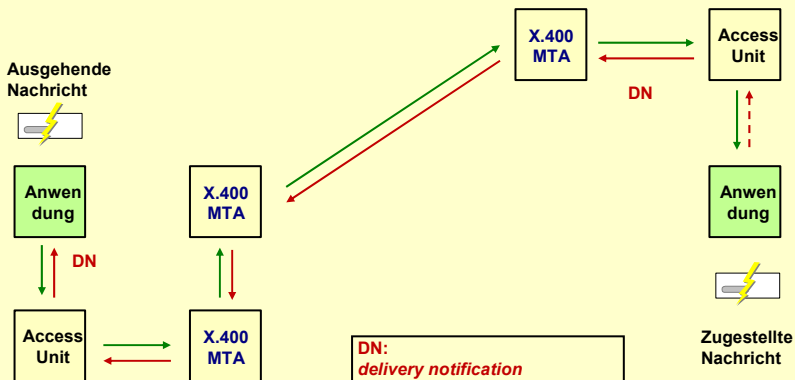
- Jede X.400-Nachricht trägt eine weltweit eindeutige Kennung auf *envelope*-Ebene, die MPDU-ID
  - Analogie zur NVE
  - *Tracking*: ADMDs sind verpflichtet, MPDU-IDs verfolgen zu können und dies auf Kundenwunsch auch zu tun
  - MTA-Zwischenstationen fügen ihre Signaturen hinzu zwecks Rückverfolgbarkeit des Weges einer Nachricht
- Eine Anwendung / ein Anwender kann den Nachrichtenkopf eindeutig kennzeichnen: *IPM-ID*
  - Grundlage für Nachweis der (Nicht-)Zustellung gegenüber dem Anwender / der Anwendung:  
(*non-*) *delivery notification*, (N)DN
  - Grundlage für Empfangsbestätigungen (*receipts*)
- Das MHS sendet (N)DNs als spezielle Nachrichtentypen automatisch bzw. auf Anforderung in standardisierter Form

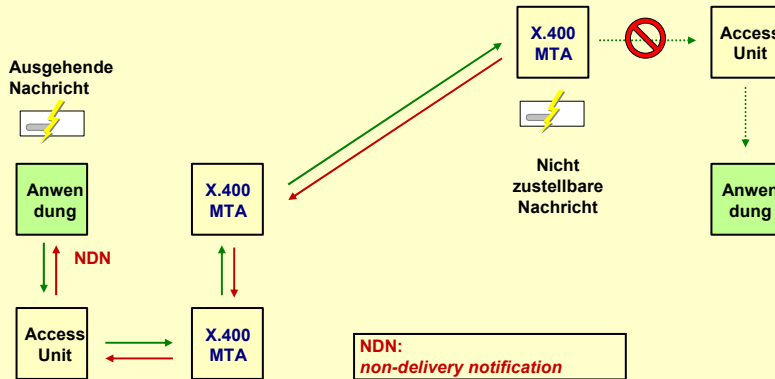


# Ebenen der Zustellbestätigung



# X.400-Zustellbestätigung





- X.400 O/R Adressen kennzeichnen i.w. eine Person
  - Hierarchisches Namensschema
  - Parametervererbung
  - “Top-level domains”? - Länder, dann ADMDs !
  - Vorlage “Gelbe Post”: Aufbau an Eigennamen orientiert
- Beispiele
  - Gillette's ADMD: C=DE / A=LION
  - Gillette's PRMD: C=DE / A=LION / P=GILLETTE
  - EDI Testadresse des Metro-Vertriebskanals“real,-”:  
C=DE / A=VIAT / P=MGI / O=EDI / OU=REAL / S=TEST



- Übliche Parameter
  - C (country)
  - A (ADMD)
  - P (PRMD)
  - O (organization)
  - OU1 ... OU4 (org. unit 1...4)
  - S (surname)
  - g (given name)
  - I (middle initial)
  - CN (common name)
  - ... (generational qualifier), etc. ...
- Besondere Parameter
  - DDAN, DDAV
    - (*domain defined attrib.*) - z.B. zur Abbildung der Parameter von VANS hinter Gateways, etwa von IBM-IE
  - X.121
    - Für die Angabe von Faxnummern und Klartext-Empfängernamen
  - (viele weitere, selten benötigt)



- Routing
  - Grundlage: Die O/R-Namen
  - Nur zwischen "benachbarten" MTAs
    - im Gegensatz etwa zu SMTP!
    - Nachbarbeziehungen erfordern i.d.R. Verträge und Einmalaufwände
    - Analogie: IP-Router im LAN, explizite Regeln notwendig für Übergänge
  - *Store-and-forward* Prinzip
    - Normalerweise baut der sendende MTA die Verbindung auf
  - Üblich: Default-Route anlegen, bei PRMDs meist die zur ADMD
- Robustheit:
  - Regelungen für mehrfache Versuche eines Verbindungsaufbaus ("wann gibt der MTA auf?")
  - "next hop list":
    - Wenn die Hauptstrecke streikt, aktiviere die Backup-Strecke
    - Wenn die Direktverbindung streikt, route an den Default (z.B. die ADMD)



- Ablehnung bei endgültiger Unzustellbarkeit über NDN
  - kein stilles Verwerfen
  - geordnetes Zeitverhalten
  - standardisierte NDN, detaillierte Standardcodes für Ablehnungsgrund, z.B.
    - Route unbekannt
    - Empfängeradresse existiert nicht
    - Empfängeradresse nicht erreichbar
    - Überschreitung der max. erlaubten Zustellzeit
  - Kein unnötiges Rücksenden des Inhalts - nur NDN
- Zustellprioritäten
  - niedrig
  - normal
  - dringend



- Traditionell
  - X.25-Stack: **TP2**, native X.25-Kopplung
  - Einfach zu konfigurieren
  - Sicher, da virtuelle P2P-Verbindung
  - Leider teuer
- Per RFC, inzwischen sehr verbreitet
  - IP-Stack: **TP0**
  - Für X.400 reservierter Port: 102
  - Kopplung flexibel realisierbar, z.B.
    - IP über *Dial-up* ISDN
    - IP über VPN
    - IP direkt über Internet (ggf. via *Firewall*)



- X.400 (class 4 - reicht für EU)
  - 0,125 € / kb (erstes kB)
  - 0,075 € / kb (ab 2. kB)
- VANS am Bsp. Phoenix / GE-GXS
  - 40000 Belege à 2 kB = 13000 €  
==> 0,165 € / kB
  - Abrechnung 100-Byte-weise
  - separat: Gebühr pro Anwahl der Mailbox
- In beiden Fällen zusätzlich:
  - Monatliche Grundgebühren
  - Leitungskosten, z.B. X.25, ISDN



## X.400 vs. Internet Mail

Vergleich X.400 - Internet Mail  
Die Antwort der IETF: EDI-INT



## Vergleich X.400 - Internet Mail



- **Secure**
  - Documents managed by secure systems
- **Traceable**
  - Misrouted mail can be tracked down
- **Receipts readily available**
  - no more "I never got it"
- **Sender Certified by originating e-mail carrier**
- **Not Secure**
  - can not be trusted with confidential information
- **Not Traceable**
  - lost messages are permanently lost
- **Not Receiptable**
  - You'll never know if the mail arrived
- **Sender Spoofable**
  - You're never sure who really sent the message



## Vergleich X.400 - Internet Mail



- **Known path**
  - Only handled by responsible commercial e-mail firms
- **Fast**
  - X.400 standards require 95% of mail delivered within 45 minutes. With the Internet becoming increasingly bogged down, X.400s timely delivery becomes increasingly important.
- **Unknown path and handling**
- **May be fast or very slow**
  - mail may take days to be delivered



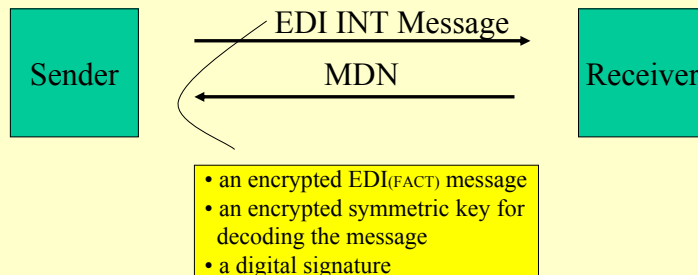


# Internet Messaging

## EDI-INT Trends



- Standard der IETF (Internet Engineering TaskForce)
- Übertragung von EDI Nachrichten (EDIFACT, ANSI X.12) über Internet
- Transportprotokoll
  - eMail (EDI-INT AS1) oder
  - HTTPS (EDI-INT AS2)
- Nachrichtenrouting
  - durch Auswertung der EDI Service Segmente (UNB,...)
- Verschlüsselung und Absenderauthentifizierung
  - S/MIME, RSA, X.509 Zertifikate für *Public Key Management*
- Sicherung der Übermittlung
  - durch „*Message Disposition Notification*“ (MDN) = standardisierte, verschlüsselt übertragene Zustellberichte
- **Schon wieder veraltet oder kurz vor dem Durchbruch?**
  - Verbreitung „nur“ in USA, aber nun Bewegung auch in Europa
  - Weitere Verbreitung und Reifung behindert durch neue Entwicklungen



*The conversion and transmission of data to an EDI-INT compliant message type is comprised of the following steps:*

- 1. Deliver data to NetIXServer*
- 2. Convert XML, EDI-X12, EDIFACT, binary, or other types of data to standard S/MIME format*
- 3. Encrypt, sign and route data*
- 4. Transport data over a network*
- 5. Receive data at NetIXServer*
- 6. Perform verification checks*
- 7. Re-convert data from standard S/MIME to XML, EDI-X12, EDIFACT, binary, or other types of data.*
- 8. Deliver data to back-end system*
- 9. Send signed return receipt to sender*



- https
  - Interessant als low-level Protokoll
- SOAP
  - Problem: Nur für XML-Daten geeignet
- XML Frameworks
  - Bisher allgemeinsten Ansatz: ebXML

(später zu besprechen)

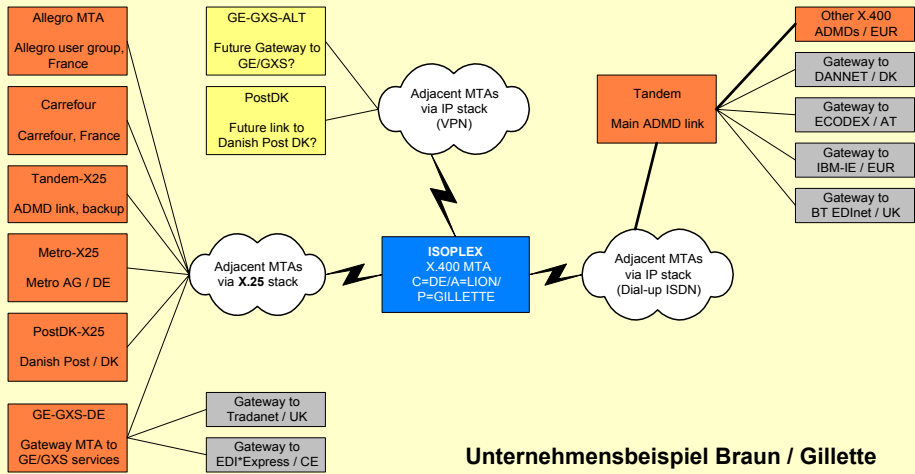


## X.400: Das Bindeglied

Abschließende Betrachtung zum  
*Messaging* anhand eines  
Unternehmensbeispiels



# X.400: Das Bindeglied aller VANS



Unternehmensbeispiel Braun / Gillette