



TrustX Praxis Version 4

Schnittstellenbeschreibung für Praxissoftware-Hersteller

Version 1.2
28.07.2015



Ist aus der Zusammenarbeit
von NewIndex, Ärztekasse
und BlueCare entstanden

TrustX Management AG
Postfach 216
8411 Winterthur
trustx@trustx.ch
www.trustx.ch

Inhaltsverzeichnis

1	Einleitung	3
1.1	Hintergrundinformation	3
1.2	TrustCenter	3
1.3	HIN	3
1.4	Ansprechpartner	3
2	TrustX in der Arztpraxis	4
2.1	Übersicht Gesamtsystem	4
2.2	TrustX Praxis	6
2.3	XML-Standards und Wertebereiche	7
3	Zusatzprodukte der TrustCenter	9
3.1	Abrechnungsarten	9
3.2	Zusatzprodukte eDA und Rechnungsdruck	11
3.3	Steuerung der Zusatzprodukte (bis XML 4.1)	12
3.4	Steuerung der Zusatzprodukte ab XML 4.3	14
4	Anforderungen an die Praxissoftware	17
4.1	Übersicht	17
4.2	Elektronische Rechnungen und Mahnungen bereitstellen	17
4.3	Möglichkeiten zur Integration von TrustX Praxis	17
4.4	Eindeutiges Token (Dokumenten-Identifizier) für eDA TG	18
4.5	Hinweise zur Steuerung von Zusatzprodukten	19
4.6	Doppelte Rechnungen / doppelte Mahnungen	19
4.7	Storno	22
4.8	TrustCenter „TC Test“	22
5	Funktionale Beschreibung von TrustX Praxis	23
5.1	Register <Übersicht>	24
5.2	Register <Details>	25
5.3	Register <Konfiguration>	25
5.4	Register <Info>	27
5.5	Mandanten in TrustX Praxis	27
6	Schnittstelle zu TrustX Praxis	28
6.1	Einleitung	28
6.2	Schnittstellendefinition	28
6.3	Events	30
6.4	TrustX einbinden	33
6.5	Dateibenennung	33
6.6	Verzeichnisstruktur	33
6.7	Wichtige Dateien von TrustX Praxis v4	34
6.8	Wichtige Dateien von TrustX Praxis v3	34
7	Anhänge	36
7.1	ESR-Zeile als Dokumenten-Identifizier (Token)	36
7.2	TrustX-Event Beispiel (VB6)	37
7.3	Schematische Darstellung – Rechnungsdruck-Steuerung & eDA TP	39

1 Einleitung

1.1 Hintergrundinformation

Die TrustCenter sind nach einem gemeinsamen Konzept aufgebaut. Sie arbeiten alle mit demselben IT-System "TrustX". Dieses wurde in einem JointVenture von BlueCare und Ärztekasse speziell für die TrustCenter entwickelt.

Das vorliegende Dokument fasst die Spezifikationen für die Schnittstellen der Praxissoftware-Systeme zu TrustX zusammen. Es dient der Information und Dokumentation der Anbieter von Praxissoftware-Lösungen, welche ihren Kunden den technischen Anschluss an deren TrustCenter ermöglichen wollen.

1.2 TrustCenter

Die TrustCenter wurden zum Sammeln und Auswerten von Abrechnungsdaten aus den Praxen errichtet. Mehr Informationen dazu finden Sie unter www.trustx.ch oder bei den einzelnen TrustCentern.

Die TrustCenter bieten auch eine technische Lösung für den elektronischen Austausch der Rechnungsdaten mit den Versicherern. Diese basiert für den Tiers Garant auf dem "Pull-Verfahren", im Tiers Payant auf dem "Push-Verfahren".

1.3 HIN

TrustX verwendet für den Zugang zu den TrustX-Diensten und Applikationen die Sicherheitslösung (Authentifizierung, verschlüsselter Datenaustausch) der Health Infonet AG (HIN). Weitere Informationen zu HIN finden Sie unter www.hin.ch.

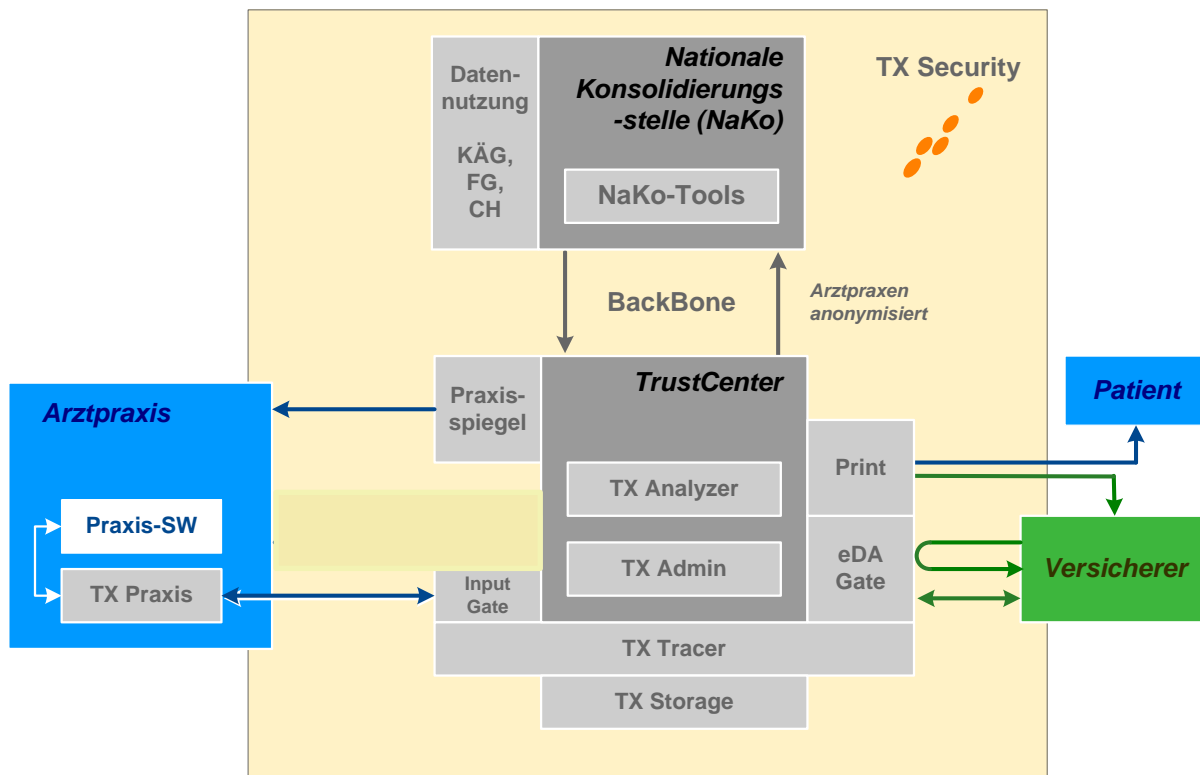
1.4 Ansprechpartner

Ansprechpartner für technische Fragen und Fragen zu den Zusatzprodukten ist unser Support. Sie erreichen diesen zu den üblichen Geschäftszeiten unter 052 234 70 00 oder via E-Mail unter support@trustx.ch.

2 TrustX in der Arztpraxis

2.1 Übersicht Gesamtsystem

Das TrustX-System umfasst eine Vielzahl von Funktionalitäten, welche in Modulen gruppiert sind. Die nachstehende Übersicht zeigt den logischen Aufbau des Systems schematisch auf.



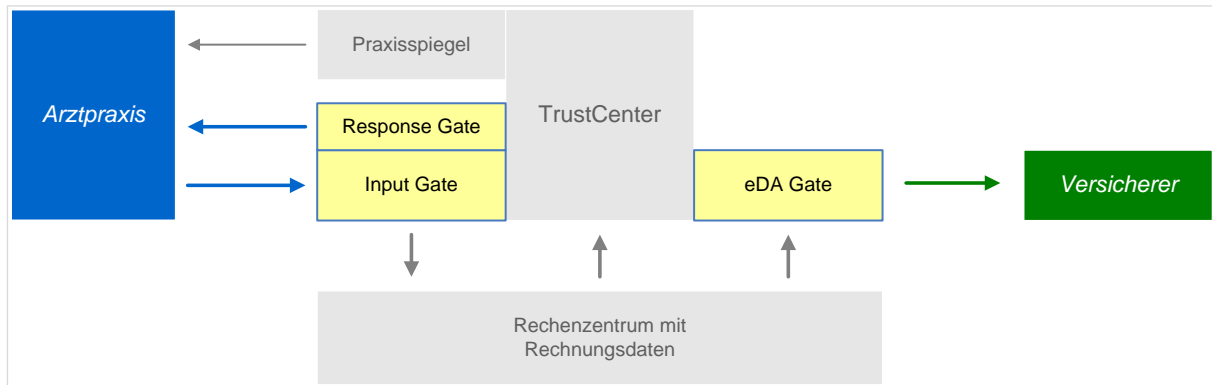
Im Fokus dieses Dokuments stehen die Schnittstelle zwischen der Arztpraxis bzw. der Praxissoftware und dem InputGate von TrustX sowie die zugehörigen Prozesse. Im Dokument finden sich weiter einige Erläuterungen zu den Zusatzprodukten der TrustCenter sowie den daraus folgenden Anforderungen an die Praxissoftware.

2.1.1 Datenaustausch mit dem Input Gate und Response Gate des TrustCenters

Der elektronische Transport der XML-Rechnungen und -Mahnungen aus der Arztpraxis erfolgt mittels TrustX Praxis via Internet an das InputGate des vom Arzt gewählten TrustCenters. Dieses nimmt bei Eingang der Datenlieferungen eine Prüfung des Absenders vor (HIN-Authentifizierung) und bestätigt der Arztpraxis den Empfang der Daten. Bei der Übermittlung von Mahnungen wird zusätzlich in Echtzeit überprüft, ob die zugehörige XML-Rechnung im System bekannt ist.

Mit TrustX Praxis können in der Arztpraxis im Tiers Payant auch Rechnungsantworten vom Response Gate des TrustCenters empfangen werden.

Die übermittelten Rechnungsdaten werden einerseits zur statistischen Auswertung verwendet (Praxispiegel) und andererseits für den elektronischen Datenaustausch (eDA) mit den Versicherern im eDA Gate bereitgestellt.



2.1.2 Security und Anonymisierung

Zur Gewährleistung des Datenschutzes wird die Sicherheitslösung von HIN verwendet (Authentifizierung des Datensenders, Streckenverschlüsselung).

Zusätzlich werden von TrustX Praxis vor der Übermittlung die Patientenangaben verschlüsselt und anonymisiert. Eine Entschlüsselung der Rechnung oder Mahnung findet ausschliesslich in den folgenden zwei Fällen statt:

- Druck & Versand der Rechnung
- Elektronische Übermittlung (eDA) an den Versicherer. Die Versicherer sind für den eDA ebenfalls via HIN gesichert und Datenschutzkonform an TrustX angebunden

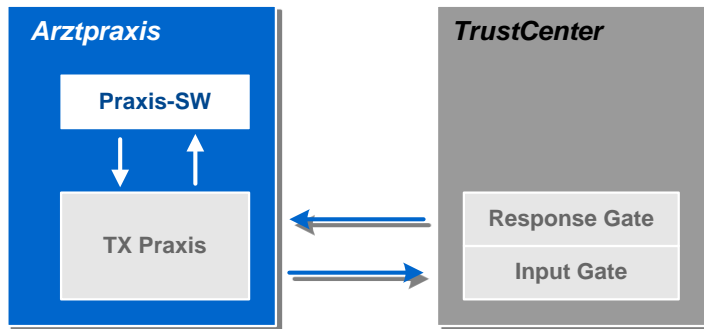
2.1.3 Technische Voraussetzungen zur Datenanlieferung ans TrustCenter

Um aus der Arztpraxis Daten ans TrustCenter liefern zu können, sind folgende technischen Voraussetzungen notwendig:

- Praxissoftware, welche TrustX unterstützt
- TrustX Praxis (www.trustx.ch)
- Internetzugang mit aktuellem HIN-Client (HIN Abonnement, www.hin.ch)

2.2 TrustX Praxis

TrustX Praxis stellt den Datenaustausch zwischen der Arztpraxis und dem Input Gate sowie dem Response Gate des TrustCenters sicher. TrustX Praxis ist ein Softwaremodul mit eigener Benutzeroberfläche und einer Schnittstelle zur direkten Ansteuerung aus der Praxissoftware und kann vollständig in die Praxissoftware integriert werden



Weitere Informationen finden sich unter:

- Kapitel 4.3 „Möglichkeiten zur Integration von TrustX Praxis“
- Kapitel 5 „Funktionale Beschreibung von TrustX Praxis“
- Kapitel 6 "Schnittstelle zu TrustX Praxis“

2.2.1 Datenbereitstellung – Anlieferung an InputGate

Grundsätzlich sind alle abgerechneten Rechnungsdaten unmittelbar nach der Fakturierung in der Arztpraxis an das TrustCenter zu übermitteln. Dies ist unter anderem aus folgenden Gründen wichtig:

- Nur wenn die Arztpraxis alle Rechnungsdaten übermittelt, erhält sie auch eine aussagekräftige Statistik ihrer Leistungen
- Die Qualität der Rechnungskollektive (Statistik) wird markant verbessert
- Je schneller die Rechnungen angeliefert werden, desto schneller stehen diese im Praxisspiegel zur Verfügung.
- Im Tiers Garant ist es wichtig, dass die XML-Rechnungen im TrustCenter bereits vorliegen, wenn sie vom Versicherer mittels Token (Dokumenten-Identifizier) abgeholt werden möchten.

Die XML-Rechnungen/-Mahnungen werden in einem konfigurierbaren Verzeichnis bereitgestellt, so dass sie mit TrustX Praxis ans InputGate des entsprechenden TrustCenters übermittelt werden können.

2.2.2 Rechnungsantworten – Empfang von ResponseGate

TrustX Praxis unterstützt den Tiers Payant vollumfänglich. Rechnungsantworten werden im Anschluss an die Rechnungs- und Mahnungsübermittlung automatisch empfangen und im vorgegebenen Verzeichnis abgelegt. Bedingung ist die korrekte Konfiguration von TrustX Praxis. Siehe dazu auch Kap. 5.3 „Register <Konfiguration>“ sowie Kap. 6.2 „Schnittstellendefinition“.

- ➔ Der Empfang von Rechnungsantworten ist nur relevant, wenn die Arztpraxis TP-Rechnungen via TrustCenter an die Versicherung sendet.

2.2.3 Funktionsweise von TrustX Praxis

TrustX Praxis anonymisiert die Patienteninformationen und übermittelt die elektronischen Rechnungen verschlüsselt an das InputGate des TrustCenters.

Im Einzelnen nimmt TrustX Praxis folgende Aufgaben wahr:

- Lesen der Rechnungsdaten aus dem konfigurierten Verzeichnis;
- Schemavalidierung der bereitgestellten Rechnungen;
- Bestimmung des Patienten-Code¹⁾ sowie Verschlüsselung und Anonymisierung der Personalien (Patient, gesetzlicher Vertreter sowie Versicherter);
 - 1) Für statistische Zwecke wird aus den Patientenangaben ein anonymer Patienten-Code gebildet und im XML-File gespeichert.
- Transport der Daten von der Arztpraxis an das TrustCenter (HIN-gesichert);
- Rechnungsantworten abholen.

2.2.4 Unterstützte Betriebssysteme

TrustX Praxis unterstützt Betriebssysteme, die durch HIN unterstützt werden (Stand Sommer 2014)

- Ab Windows Vista
- Mac OS X 10.5. und höher

Eine aktuelle Liste aller von HIN unterstützten Betriebssystemen finden Sie unter www.hin.ch.

Für folgendes Betriebssystem wurde die grundsätzliche Operabilität getestet. Jedoch wird dies nicht offiziell unterstützt:

- Windows XP

2.3 XML-Standards und Wertebereiche

Folgende XML-Formate werden von TrustX Praxis v4 unterstützt:

- XML 4.3/4.4: Generelle Rechnung und Rechnungsantwort inkl. Mahnung und Storno (generalInvoiceRequest/Response 4.3 und 4.4)
- XML 4.0 Arzt: Arztrechnung und Antwort auf eine Arztrechnung (mdInvoiceRequest/Response 4.0)
- XML 4.0 Apotheke: Apothekerrechnung und Antwort auf eine Apothekerrechnung (PharmacyInvoiceRequest/Response 4.0)
- XML 4.0 Mahnung: Rechnungs-Mahnung und Antwort auf eine Mahnung (InvoiceReminderRequest/Response 4.0)

Für XML Standard 4.3 oder höher können Rechnungen in folgenden Wertebereichen angeliefert werden:

Attribut	Wertebereich	Kommentar
body:role	physician physiotherapist chiropractor ergotherapist nutritionist midwife logotherapist hospital pharmacist dentist labtechnician dentaltechnician othertechnician psychologist wholesaler nursingstaff transport druggist naturopathicdoctor naturopathictherapist other	Gilt für XML Standard 4.3 oder höher. „hospital“ und „other“ sind nicht zulässig. Werte in grau stehen nicht zur Verfügung.
body:place	practice hospital lab association company	Gilt für XML Standard 4.3 oder höher. „hospital“ ist nicht zulässig. Werte in grau stehen nicht zur Verfügung.
payload	invoice reminder credit	Gilt für XML Standard 4.3 oder höher. „credit“ ist nicht zulässig.

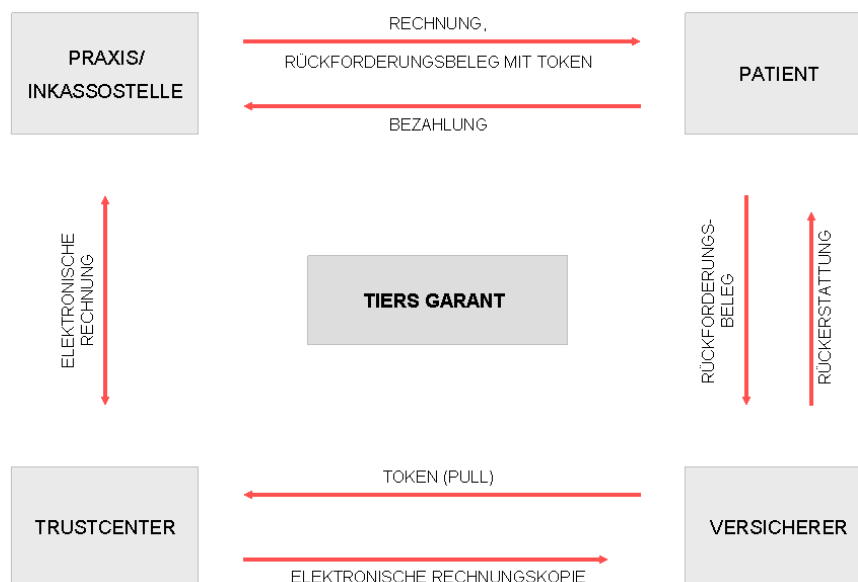
3 Zusatzprodukte der TrustCenter

Das TrustCenter bietet neben dem Praxisspiegel auch verschiedene Zusatzprodukte im Bereich eDA und Rechnungsdruck an. In diesem Kapitel werden die möglichen Abrechnungsarten aus technischer Sicht, die Zusatzprodukte sowie die entsprechende Steuerung über die Praxissoftware erläutert.

3.1 Abrechnungsarten

3.1.1 Tiers Garant – Das "Holprinzip" (Pull)

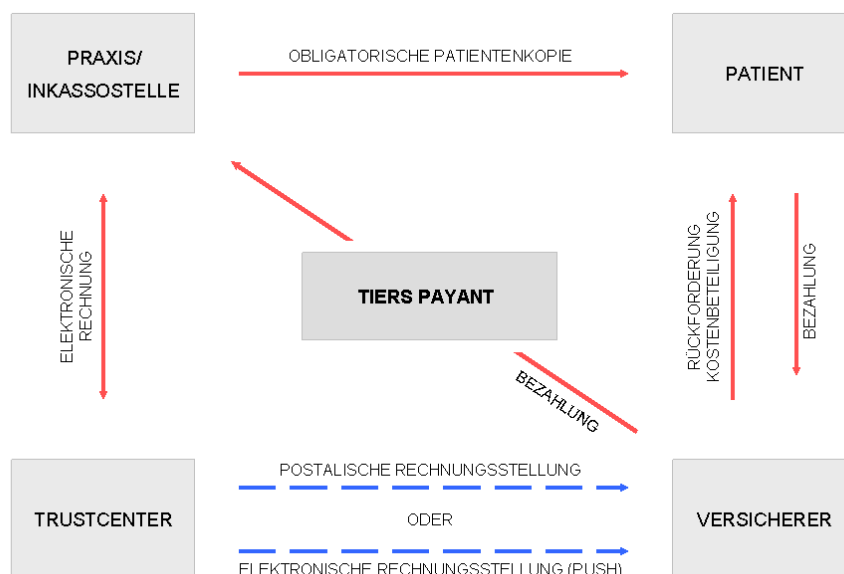
Um den elektronischen Datenaustausch im Tiers Garant umsetzen zu können, wurde das so genannte „Holprinzip“ entwickelt. Dabei wird auf den Rückforderungsbeleg ein eindeutiger Dokumenten-Identifizier (= Token) gedruckt. Reicht der Patient den Rückforderungsbeleg bei seinem Versicherer ein, kann dieser mit dem Dokumenten-Identifizier die elektronische Rechnungskopie im entsprechenden TrustCenter abholen.



3.1.2 Tiers Payant – Das "Bringprinzip" (Push)

Das TrustCenter tritt im Tiers Payant und Soldant gegenüber dem Arzt und Versicherer als Intermediär im Transport der elektronischen Rechnungsdaten auf.

Im Tier Soldant entfallen die Verbindungen zum Patient (z.B. keine Rechnungskopie an den Patienten).



3.1.3 Abhängigkeiten

Die verschiedenen Abrechnungsarten unterliegen verschiedenen Anforderungen und Abhängigkeiten. Diese sind in der nachfolgenden Tabelle zusammengestellt:

	Tiers Garant (TG)	Tiers Payant (TP)	Tiers Soldant (TS)
Originalrechnung	geht an Patient	geht an Versicherer	geht an Versicherer
Format der Originalrechnung	Frei	Gemäss Vorgabe von Forum-Datenaustausch	Gemäss Vorgabe von Forum-Datenaustausch
Rechungsantwort	nicht unterstützt	unterstützt	unterstützt
Mahnung	geht an Patient	unterstützt für elektr. und für Papierrechnung	unterstützt für elektr. und für Papierrechnung
Rechnungskopie	Rückforderungsbeleg	Patientenkopie (im KVG vorgeschrieben)	---
Format der Rechnungskopie	Gemäss Vorgabe von Forum-Datenaustausch	Frei	---
Token	auf Rückforderungsbeleg Nur ausdrucken, falls eine elektronische Rechnungskopie ans TrustCenter geschickt wird und der Arzt am eDA TG teilnimmt	---	---
ESR	---	Ja, auf Originalrechnung	Ja, auf Originalrechnung

	Tiers Garant (TG)	Tiers Payant (TP)	Tiers Soldant (TS)
Gesetze	KVG, VVG, (UVG/MV/IV)	KVG, (VVG)	UVG/MV/IV
eDA-Prinzip	Holprinzip (Pull-Verfahren)	Bringprinzip (Push-Verfahren)	Bringprinzip (Push-Verfahren)

Forum für Datenaustausch: <http://www.forum-datenaustausch.ch/>

3.2 Zusatzprodukte eDA und Rechnungsdruck

Die Zusatzprodukte der TrustCenter im Bereich eDA und Druck&Versand sind in der nachfolgenden Tabelle zusammengefasst:

ID-Nr.	Zusatzprodukt / Option	Produktname	Beschreibung / Bemerkung	Gesetz
A1.1	Zusatzprodukt	eDA TG	Bereitstellung von elektronischen Rechnungen im TG. (eDA TG)	alle Gesetze
A1.2	Option zu A1.1	Rechnungsdruck TG	Druck und Versand des Rechnungspakets TG (Rechnungen & Mahnungen). Token wird immer gedruckt. (TX-Print TG)	alle Gesetze
A2.1	Zusatzprodukt	eDA TP	Lieferung von elektronischen Rechnungen und Mahnungen TP. (eDA TP Push) Lieferung von TP eDA Rechnungen / Mahnungen auf Papier. Nur wenn Versicherung eDA nicht unterstützt. (eDA TP Print)	KVG, (VVG)
A2.2	Option zu A2.1	Patientenkopie TP (obligatorische Rechnungskopie)	TX-Print der obligatorischen Rechnungskopie TP an den Patienten. (TX-Print TP)	KVG
A3	Zusatzprodukt	eDA TS	Lieferung von elektronischen Rechnungen und Mahnungen TS. (eDA TS Push) Lieferung von TS eDA Rechnungen / Mahnungen auf Papier. Nur wenn Versicherung eDA nicht unterstützt. (eDA TS Print)	UVG/MV/IV

Der "Druck und Versand" (D&V) ist jeweils nur als **Option** zu den Zusatzprodukten "eDA TG" und "eDA TP" erhältlich. Die Bestellung eines Zusatzproduktes und dessen Konfiguration sowie Administration geschieht via TrustCenter. Alle für die Verarbeitung der Rechnungen benötigten Angaben werden durch das TrustCenter in TrustX eingetragen.

3.3 Steuerung der Zusatzprodukte (bis XML 4.1)

Die Steuerung der Zusatzprodukte geschieht über das Attribut "resend" (XML-Attribute *resend*, ID 10170; Original: *resend* = false, Kopie: *resend* = true).

Nachfolgend sind die verschiedenen Vertragsmöglichkeiten (Arzt mit oder ohne TC) und deren Steuerungsmöglichkeiten (Original / Kopie) tabellarisch dargestellt.

Hinweise zur Implementation in der Praxissoftware sind in Kap. 4.5 „Hinweise zur Steuerung von Zusatzprodukten“.

	Arzt ohne TC	Arzt mit TC				
		Zusatzprodukte				
		eDA TG (A1.1)	Option Rechnungsdruck TG (A1.2)	eDA TP (A2.1)	Option Patientenkopie TP (A2.2)	eDA TS (A3)
Einfluss von Attrib. Original	--	Wird nicht beachtet	Druck und Versand (D&V) der Rechnung / Mahnung durch das TrustCenter.	Weiterleitung der Rechnung / Mahnung an Versicherer	Druck und Versand durch das TrustCenter	Weiterleitung der Rechnung / Mahnung an Versicherer
Einfluss von Attrib. Kopie	--	Wird nicht beachtet	Rechnung / Mahnung wird nicht gedruckt	Keine Weiterleitung der Rechnung / Mahnung an Versicherer	Es wird keine Patientenkopie erstellt	Keine Weiterleitung der Rechnung / Mahnung an Versicherer
Token drucken?	NEIN	JA ¹⁾	JA	--	--	--

1) Nur, falls sich der Arzt am eDA mit dem Versicherer beteiligt

Unabhängig von allfälligen Steuerungsattributen für Zusatzprodukte fließen alle Rechnungen der einem TrustCenter angeschlossenen Arztpraxen in den Praxisspiegel ein. Weitere Informationen finden Sie in den nachfolgenden Kapiteln.

3.3.1 Elektronischer Datenaustausch im Tiers Garant (eDA TG)

Damit der eDA TG funktioniert, muss die Arztpraxis

- das Zusatzprodukt eDA TG beim TrustCenter freigeben (Vertrag);
- das Token auf den Rückforderungsbeleg drucken;
- die elektronische Rechnung unmittelbar nach der Fakturierung ans TrustCenter senden.

Ist der eDA TG beim TrustCenter nicht freigegeben oder werden die Rechnungen nicht elektronisch ans TrustCenter weitergeleitet, darf kein Token auf den Rückforderungsbeleg gedruckt werden.

Das Token auf dem Rückforderungsbeleg ist für den Versicherer die Schlüsselinformation, dass die entsprechende TG-Rechnung im angegebenen TrustCenter zum Abholen bereit liegt. Werden Rechnungen im TrustCenter nicht gefunden, führt dies zu einem erheblichen Zusatzaufwand für die Versicherer.

3.3.2 eDA TP, eDA TS und Optionen "Druck & Versand"

Zur Nutzung der Zusatzprodukte eDA TP und eDA TS sowie die Optionen Rechnungsdruck TG und Patientenkopie TP müssen folgende Kriterien erfüllt sein:

- Die entsprechenden Zusatzprodukte respektive Optionen müssen von der Arztpraxis vertraglich mit dem TrustCenter vereinbart werden.
- Die Feinststeuerung folgt für die vertraglich vereinbarten Zusatzprodukte und Optionen über das Attribute resend. Es werden nur Original-Rechnungen (resend = false) prozessiert. Wird z.B. fälschlicherweise eine TG-Rechnung zweimal als Original ans TrustCenter geschickt, wird diese auch zweimal gedruckt und verschickt.

Werden Rechnungen als Original ans TrustCenter geschickt, ohne dass eines der oben erwähnten Zusatzprodukte von der Arztpraxis beim TrustCenter bestellt wurde, werden auch keine Zusatzdienstleistungen erbracht.

Wählt eine Arztpraxis den eDA TP und/oder eDA TS, werden wie oben beschrieben alle Originalrechnungen durch das TrustCenter weitergeleitet. Ist der elektronische Weg zu bestimmten Versicherern nicht möglich, wird die Rechnung automatisch gedruckt und per Post verschickt. Damit ist gewährleistet, dass alle Originalrechnungen (unabhängig von der Transportart) zum entsprechenden Versicherer gelangen.

Die Verarbeitung von Mahnungen (nur TP/TS) wird nur vorgenommen, wenn folgende Voraussetzungen erfüllt sind:

- Die Zusatzprodukte eDA TP und/oder eDA TS müssen von der Arztpraxis vertraglich mit dem TrustCenter vereinbart sein;
- Die zur XML-Mahnung gehörende XML-Rechnung muss im TrustX-System vorhanden sein;
- Bei der XML-Mahnung muss es sich um ein Original handeln (resend = false).

3.3.3 Praxisspiegel

Für den Praxisspiegel ist keine Steuerung durch die Praxissoftware notwendig. Grundsätzlich fließen alle Rechnungen in den Praxisspiegel ein. Allfällig doppelt angelieferte Rechnungen werden auf der Ebene Statistik eliminiert. Das Attribute resend wird nicht berücksichtigt.

3.3.4 Zusammenfassung

Neben den vertraglichen Vereinbarungen zwischen Arztpraxis und TrustCenter als Grundbedingung sind für die Steuerung massgebend:

- eDA TG: Das Token auf dem Rückforderungsbeleg.
- eDA TP, eDA TS, Rechnungsdruck TG oder Druck Patientenkopie TP: Es werden nur Originalrechnungen prozessiert.
- Praxisspiegel: keine Steuerung; Token und das Attribute resend sind nicht relevant. Doppelt angelieferte Rechnungen werden für die Statistik eliminiert.

3.4 Steuerung der Zusatzprodukte ab XML 4.3

3.4.1 Relevante Attribute aus der XML 4.4 Dokumentation

requestType/processing

Beschreibung der Steuerfunktionen für den Rechnungsdruck gemäss der XML 4.4 Dokumentation:

Attribut	Beschreibung	required	Default
print_at_intermediate	<p>print_at_intermediate is a processing instruction for the last intermediate in a Tiers Garant case.</p> <p>The attribute denotes if the invoice should be printed and mailed to the guarantor or if this print process was already done.</p> <p>Note, that this instruction is significant for the last intermediate only. If an infoset is sent to any party for statistical purposes then this party must not be included as intermediate.</p>	no	false
print_patient_copy, XML 4.4	<p>print_patient_copy is an instruction for the last intermediate regarding the processing of a patient copy.</p> <p>Specifically, this means that if print_patient_copy is true then the infoset is printed as a copy of the detailed invoice and mailed to the patient in a Tiers Garant case</p> <p>the infoset is printed as a copy of the detailed invoice and mailed to the patient in a Tiers Payant case</p> <p>On this printed copy the ESR coding line must be absent, the copy field should be set to "yes", and the title of the invoice should be adapted accordingly to designate the copy situation.</p> <p>Note, that this instruction is significant for the last intermediate only. If an infoset is sent to any party for statistical purposes then this party must not be included as intermediate.</p>	no	false
language	<p>language defines the used data language (name of the services etc.) encoded by a terminal set of ISO language specifiers [de fr it]</p>	yes	

Hinweis:

Das Attribut „print_at_intermediate“ wird nach Rücksprache mit den grossen Praxissoftware-Anbietern auch zur Steuerung von eDA TP (Weiterleitung von TP-Rechnungen an die Krankenversicherer) verwendet.

transportType/via

Beschreibung des via-Attributes aus dem Element Processing gemäss der XML 4.4 Dokumentation:

Attribut	Beschreibung	required	default
via	<p>via is the EAN of an intermediate</p> <p>The via element defines one intermediate hop through which the infoset was routed. It is in the responsibility of an intermediate to correctly add a further via element if this intermediate has to route the infoset to another intermediate instead of delivery to the recipient.</p>	yes	
sequence_id	<p>sequence_id is a sequence number starting with 1 that defines a certain intermediate hop. In each new via element the sequence number is increased by 1 thus forming a chain of</p>	yes	

Attribut	Beschreibung	required	default
	intermediate hops		

3.4.2 Generelle Voraussetzungen für die Verarbeitung von Rechnungen ab XML 4.3

Jede an das TrustCenter gelieferte XML-Rechnung wird statistisch ausgewertet. Damit XML-Rechnungen gedruckt oder elektronisch an die Krankenversicherer weitergeleitet werden, müssen zusätzlich folgende Bedingungen erfüllt sein:

- die Arztpraxis hat mit dem TrustCenter eine entsprechende Vereinbarung (auf der nachfolgenden Zeichnung als "Flag im TX Statusfile" zu erkennen)
- die letzte via-EAN entspricht jener des TrustCenters
- eines oder mehrere der nachfolgenden beschriebenen Attribute ist/sind in der XML 4.3 oder XML 4.4 Rechnung gesetzt

3.4.3 Verarbeitung von XML 4.0/4.1 Rechnungen

Die neue Steuerung gilt nicht für XML 4.0 bzw. XML 4.1 Rechnungen! Die Drucksteuerung von XML 4.0/4.1 Rechnungen erfolgt wie bisher über das Attribut "resend".

Das in XML 4.0 für die Steuerung verwendete Attribut "copy" wird ab XML 4.3 ignoriert.

3.4.4 Attribute für die Steuerung

Die Attribute für die Rechnungsdruck-Steuerung werden vom TrustCenter gemäss der nachfolgenden Tabelle gehandhabt.

Attribut in XML 4.3/4.4	Patientenkopie TP	Rechnungsdruck TG	eDA TP	Bemerkung für die Praxissoftware-Hersteller und Intermediäre
print_at_intermediate = TRUE	--	Rechnung mit ESR + 1 Expl. Rückforderungs- beleg	Druck und Versand oder elektronische Übermittlung an die Krankenversicherer	Das Attribute wird nur beachtet, wenn im letzten via- Attribut (Intermediär) die EAN des TCs eingetragen ist.
print_patient_copy = TRUE	Rechnungskopie ohne ESR	Kopie von Rückforderungs- beleg	--	Das Attribute wird nur beachtet, wenn im letzten via- Attribut (Intermediär) die EAN des TCs eingetragen ist.
Copy	--	--	--	Das Attribute Copy wird nicht berücksichtigt

3.4.5 Rechnungsweiterleitung von Intermediären an die TrustCenter

Intermediäre, welche Rechnung für statistische Zwecke an die TrustCenter weiterleiten, müssen neu keine Anpassung mehr an der XML-Rechnung machen. Wichtig ist, dass das letzte via-Element unverändert auf der eigenen EAN (jene des Intermediärs) belassen wird.

In XML 4.0 Rechnungen muss das Attribut "resend" weiterhin auf „TRUE“ gesetzt werden, damit keine Verarbeitung im eDA und/oder Druck&Versand durch das TrustCenter erfolgt.

3.4.6 Rechnungsweiterleitung vom TrustCenter an einen Intermediär

In Rechnungen, welche vom TrustCenter zwecks Weiterleitung an die Krankenversicherer an einen Intermediär gesendet werden, wird das letzte via-Element auf die EAN des jeweiligen Intermediärs gesetzt.

3.4.7 Schematische Darstellung der Rechnungsdruck-Steuerung (inkl. eDA TP)

Siehe Schema in Anhang, Kap. 7.3.

4 Anforderungen an die Praxissoftware

4.1 Übersicht

Alle XML-Rechnungsdateien werden von der Praxissoftware via ein Verzeichnis an TrustX Praxis übergeben.

TrustX Praxis kann

- mit eigener Benutzeroberfläche betrieben werden (Kap. 5 „Funktionale Beschreibung von TrustX Praxis“)
- über eine Schnittstelle direkt aus der Praxissoftware angesteuert werden (Kap. 6 „Schnittstelle zu TrustX Praxis“) und so vollständig in die Praxissoftware integriert werden (Kap. 4.3 „Möglichkeiten zur Integration von TrustX Praxis“).

Folgendes ist besonders zu beachten:

- Das Token (= Dokumenten-Identifizier) wird nur auf den Rückforderungsbeleg (TG) gedruckt, falls
 - der Arzt einem TrustCenter angeschlossen ist;
 - der Arzt am eDA TG teilnimmt (Regelfall).
- Das Token muss für jede Rechnung (auch für korrigierte Rechnungen) eindeutig sein.
- Es stehen Steuerungsmöglichkeiten für die Zusatzprodukte zur Verfügung.
- Storni müssen korrekt codiert werden.

In den folgenden Kapiteln finden Sie weitere Ausführungen zu diesen und weiteren Punkten betreffend der Anbindung von TrustX Praxis.

4.2 Elektronische Rechnungen und Mahnungen bereitstellen

Elektronische Rechnungskopien, Storni sowie Mahnungen sind in den unterstützten XML-Formaten im entsprechenden Verzeichnis bereitzustellen. Es sind also keine Modifikationen an den elektronischen Rechnungen notwendig, damit sie von TrustX Praxis ans TrustCenter geschickt werden können. Diese müssen lediglich in das definierte Verzeichnis abgelegt werden. Anonymisierung, Verschlüsselung und Übermittlung ans InputGate wird durch TrustX Praxis vorgenommen.

4.3 Möglichkeiten zur Integration von TrustX Praxis

TrustX Praxis kann verschieden tief in eine Praxissoftware integriert werden.

4.3.1 Keine Integration

Grundsätzlich ist keine Integration von TrustX Praxis in die Praxissoftware notwendig. Es genügt, die elektronischen XML-Rechnungsdateien in einem Verzeichnis für die weiteren Verarbeitungsschritte (Einlesen, Überprüfen, Anonymisieren, Senden) bereitzustellen. TrustX Praxis kann unabhängig von der Praxissoftware – auch auf einem anderen Rechner – gestartet werden und diese Schritte ausführen.

4.3.2 Aufruf von TrustX Praxis aus der Praxissoftware

Eine einfache Möglichkeit, dem Arzt den Datenaustausch mit dem TrustCenter zu erleichtern, ist der Aufruf von TrustX Praxis aus der Praxissoftware. Der Benutzer kann anschliessend die Oberfläche von TrustX Praxis zum Arbeiten nutzen. Es sind keine weiteren Interaktionen zwischen der Praxissoftware und TrustX Praxis notwendig.

Technisch gesehen gibt es zwei Möglichkeiten für eine solche Integration:

- Kommandozeilenaufruf von TrustX Praxis
- Konfiguration via Schnittstelle (siehe Kap. 6 „Schnittstelle zu TrustX Praxis“) Dies erlaubt es zusätzliche Konfigurationsparameter, die in der Praxissoftware bekannt sind – z.B. das Rechnungsverzeichnis – in TrustX Praxis zu setzen. Anschliessend erfolgt der Kommandozeilenaufruf von TrustX Praxis.

4.3.3 Vollständige Integration

TrustX Praxis kann auch vollständig durch die Praxissoftware gesteuert werden:

- die Oberfläche von TrustX Praxis bleibt unsichtbar;
- die Meldungen von TrustX Praxis an Ihre Praxissoftware werden direkt verarbeitet.

Diese Variante hat den Vorteil, dass auf Fehlermeldungen von TrustX Praxis direkt aus der Praxissoftware reagiert werden kann. Die Praxissoftware kann zudem die Funktionalitäten von TrustX Praxis unter der eigenen Oberfläche anbieten. Dies erlaubt einen hohen Komfort für den Benutzer, bedeutet aber auf der anderen Seite einen höheren Entwicklungsaufwand seitens des Praxissoftware-Anbieters. Die korrekte Anzeige von Meldungen wie auch die Softwareaktualisierung liegen bei der vollständigen Integration in der Verantwortung des Praxissoftware-Herstellers. Weitere Informationen zur Schnittstelle finden Sie im Kap. 6 „Schnittstelle zu TrustX Praxis“.

4.4 Eindeutiges Token (Dokumenten-Identifizier) für eDA TG

Die Rückforderungsbelege (TG) sind mit einem Token (Dokumenten-Identifizier) zu versehen, wenn den Versicherern die Möglichkeit geboten werden soll, die Rechnungen nach dem Pull-Verfahren im TrustCenter zu beziehen.

Die Basis für das Token bildet die ESR-Zeile der Rechnung. Das System setzt voraus, dass das Token für jede Rechnung eindeutig ist. Mit der ESR-Nummer ist dies einfach erreichbar, weil diese hierarchisch aufgebaut ist. Die einzelnen ESR-Teilnehmer sind eindeutig identifiziert. Die Vergabe von eindeutigen ESR-Nummern liegt also in der Verantwortung der Arztpraxis. Damit der elektronische Datenaustausch einwandfrei funktioniert, müssen auch korrigierte Rechnungen neue ESR-Nummern erhalten. Dies wird bspw. über eindeutige Rechnungsnummern erreicht.

Für die Erstellung eines Tokens wird die ESR-Zeile modifiziert: an Stelle der Belegart (erste und zweite Stelle) wird das TrustCenter codiert. Die Prüfziffer an Stelle 13 ist entsprechend anzupassen. Details dazu sind in Anhang 1: ESR-Zeile als Dokumenten-Identifizier" zu finden.

Das Token darf ausschliesslich unter folgenden Bedingungen gedruckt werden:

- Nur bei der Abrechnungsart TG;
- Nur wenn der Arzt einem TrustCenter angeschlossen ist;
- Nur wenn sich der Arzt am eDA mit dem Versicherer beteiligt.

4.5 Hinweise zur Steuerung von Zusatzprodukten

Die Mechanik der Steuerung für Zusatzprodukte ist im Abschnitt 3.4 „Steuerung der Zusatzprodukte“ eingehend beschrieben. Im Folgenden sollen einige Hinweise gegeben werden, was dies für die Implementation in der Praxissoftware heissen könnte. Dabei geht es im Wesentlichen darum, via Praxissoftware auf der Ebene der einzelnen Rechnung festlegen zu können, ob diese als Original oder als Kopie ans TrustCenter (TC) geschickt wird und ob die Rechnung in der Arztpraxis im Rahmen der Fakturierung gedruckt werden soll oder nicht.

4.5.1 Option Rechnungsdruck TG

Entscheidet sich eine Arztpraxis für den Rechnungsdruck im TC, ist es wichtig, dass keine doppelten Rechnungen / Rückforderungsbelege an den Patienten gehen.

Diese Gefahr besteht dann, wenn eine Rechnung als "Sofortrechnung" dem Patienten gleich nach der Konsultation mitgegeben wird. In diesem Fall muss die Rechnung ohne Druckauftrag ans TrustCenter geschickt werden.

4.5.2 eDA TP und TS

Im elektronischen Datenaustausch nach dem Push-Prinzip (eDA TP und eDA TS) kann es sinnvoll sein, die Selektion für einzelne Versicherer und Gesetze vorzunehmen. Damit kann der Arzt folgende Fälle unterscheiden/steuern:

- eDA-Vertrag zwischen Versicherer X und TC besteht:
Rechnungen als Original schicken. Die Rechnungen werden somit vom TC an den Versicherer (elektronisch) gepusht.
- kein eDA-Vertrag zwischen Versicherer Y und TC:
Rechnungen als Kopie an das TC schicken und selber drucken.

4.6 Doppelte Rechnungen / doppelte Mahnungen

Die Erkennung von doppelten XML-Rechnungen bzw. -Mahnungen findet in TrustX an zwei Orten statt. Einmal erkennt TrustX Praxis aufgrund unterschiedlicher Kriterien Dubletten noch vor der Anlieferung an das InputGate. Nach erfolgter Anlieferung wird beim Einleseprozess in die Statistikdatenbank ebenfalls ein Doublettencheck angewendet.

Der XML-Standard fordert, dass die folgenden drei Attribute eine Rechnung/Mahnung eindeutig identifizieren. Rechnungen, welche sich in mind. einem dieser drei Attribute unterscheiden, gelten als neue Rechnung.

XML 4.0		ab XML 4.3	
Rechnung	Mahnung	Rechnung	Mahnung
Biller_EAN	Biller_EAN	Biller_EAN (entfällt)	Biller_EAN (entfällt)
Invoice_ID	Reminder_ID	Invoice_ID	Reminder_ID
Invoice_TimeStamp	Reminder_TimeStamp	Invoice_TimeStamp	Reminder_TimeStamp
		<u>Invoice_Date</u> (neu)	<u>Reminder_Date</u> (neu)

4.6.1 Erkennung doppelter Rechnungen in TrustX Praxis

Für die Erkennung von doppelten Rechnungen werden während des Sendens der Rechnungen verschiedene Rechnungsinformationen im trustx.log abgelegt:

Rechnungsinformation	Bemerkung
Biller_EAN (XML 4.0) bzw. Invoice_Date (ab XML 4.3)	
Invoice_ID	
Invoice_TimeStamp	
Provider ZSR	Ist zusammen mit der Angabe des TrustCenters die eindeutige Identifikation des TrustCenter-Kunden innerhalb des TrustX Systems.
Amount	Dient der Unterscheidung von Rechnung und Storno (innerhalb des Logfiles).
Invoice / tiers_payant / purpose	Angabe über den Rechnungszweck im Falle eines TP-Storno zur Unterscheidung von Rechnung und Storno (innerhalb des Logfiles).

In Erweiterung zum XML-Standard werden zusätzlich die Attribute Provider_ZSR, Amount und TP Purpose für die sichere Erkennung geprüft.

Stimmen die Werte der sechs aufgeführten Attribute überein, wird die Rechnung als Doublette erkannt, Weicht mind. einer der sechs Werte ab, wird die Rechnung als neue Rechnung an das TrustCenter übermittelt.

Durch das Löschen des Logfiles trustx.log oder eines einzelnen Eintrages kann die Erkennung von Doubletten umgangen werden. Dies ist jedoch nur in Ausnahmefällen zu empfehlen, bspw. dann, wenn Rechnungen explizit nochmals im TrustCenter gedruckt werden sollen.

4.6.2 Erkennung doppelter Rechnungen in der Statistik-Datenbank

In der Praxis tritt immer wieder mal der Fall auf, dass Rechnungen versehentlich doch mehrmals an das TrustCenter geliefert werden (Löschen des Logfiles, doppelte Anlieferung über einen zweiten PC, usw.). Solche doppelten Rechnungen würden die Auswertungen und Statistiken der Ärzte verfälschen.

Vor dem Einlesen der Rechnungen in die Statistikdatenbank findet deshalb nochmals eine Prüfung auf Doubletten statt. Dabei kommen zum Teil erweiterte Prüfungen zur Anwendung, um auch in Fehlerfällen Doubletten erkennen zu können, z.B. wenn die Arztpraxis die gleichen Rechnungen doppelt oder mehrfach, jedoch jeweils mit unterschiedlichem Timestamp generiert und übermittelt.

Doppelte Rechnungen werden im TX Tracer mit einem Q-Index 4 und der Meldung „Doppelte Rechnung“ ausgewiesen.

→ WICHTIG: Mehrfach ans InputGate gelieferte Rechnungen werden lediglich in der Statistik-DB eliminiert. In der Weiterverarbeitung (eDA TG und eDA TP/TS) werden solche Rechnungen jedoch mehrfach verarbeitet (z.B. gedruckt)!

4.6.3 Erkennung doppelter Mahnungen in TrustX Praxis

Die Erkennung von doppelten Mahnungen geschieht analog jener von Rechnungen. Dazu werden folgende Mahnungsinformationen im trustx.log abgelegt:

Rechnungsinformation	Bemerkung
Biller_EAN (XML 4.0) bzw. Reminder_Date (ab XML 4.3)	
Reminder_ID	
Reminder_TimeStamp	
Amount	Dient der Unterscheidung von Rechnung und Storno (innerhalb des Logfiles)

Da Mahnungen nicht in die Statistik-Datenbank importiert werden, erübrigt sich dort ein Doubletten-check.

4.6.4 Aufbau von trustx.log

Das Logfile trustx.log dient der lokalen Erkennung von Dubletten und ist wie folgt aufgebaut:

Rechnungsnr. (Invoice_ID) / Mahnungsnr. (Reminder_ID)	Amount	TimeStamp senden	ZSR Provider	Biller EAN	Invoice_ TimeStamp / Reminder_ TimeStamp	tp purpose	
0016997 - 00574/17	469.60	1179811727	L123488	7601000030108	1173139917	0	1)
0016997 - 00574/17	-469.60	1179819677	L123488	7601000030108	1173139917	0	2)
0016997 - 00574/17	469.60	117981967	L123488	7601000030108	1173139917	1	3)
REM - 574/17	469.60	1179814541		7601000030108	1173164799	0	4)

1) Rechnung

2) Storno Variante 1 (negative Beträge, für TG- und TP-Rechnungen möglich)

3) Storno Variante 2 (tp purpose, nur für TP-Rechnungen)

4) Mahnung

Ab TX Praxis Version 2.3 wird das trustx.log nach der Übermittlung jeder Rechnung geschrieben (früher erst nach einer Session). Damit wird die Verfügbarkeit von Log-Daten bei einem Übermittlungsunterbruch verbessert.

4.7 Storno

4.7.1 Storno im XML 3.0 / 4.0

Storni sind mit gleicher ESR-Nummer und allen mit Minus eins multiplizierten Betragsfeldern zu übermitteln. Alle anderen Felder bleiben gleich. In der XML-Invoice-Rechnungsdefinition müssen die für die Statistik irrelevanten Felder <Anzahlung> (10335) und <Gerundeter Nettorechnungsbetrag> (10340) 0 oder grösser sein. Diese werden daher auf Null anstatt auf den Minusbetrag gesetzt. Auf diese Art wird eine statistische Korrektur vorgenommen.

4.7.2 Storno im XML 4.0 für TP-Rechnungen

Im XML 4.0-Standard wird eine TP-Rechnung mit dem Attribute purpose = „annulment“ als Storno deklariert (XML 4.0: request / invoice / tiers_payant / purpose; ID 11265).

4.7.3 Storno ab XML 4.3

Ab XML 4.3 wird ein Storno mit dem entsprechenden Attribut „storno = true“ gesetzt. Zusätzlich müssen

- die „quantity“ jeder Leistungsposition mit -1 multipliziert werden
- die Felder „amount_prepaid“ und „amount_reminder“ mit -1 multipliziert werden

Ausführliche Informationen sind der Dokumentation des XML Standards 4.3 zu entnehmen.

4.8 TrustCenter „TC Test“

Zu Testzwecken steht ein Test-TrustCenter „TC Test“ zur Verfügung.

→ **Bitte verwenden Sie für Tests ausschliesslich das "TC Test".**

Führen Sie keine Tests mit produktiven TrustCentern durch, da die Testrechnungen sonst in die Statistiken der TrustCenter eingehen.

Für Testlieferungen an "TC Test" muss der HIN-Client gestartet sein. Eine Anmeldung bei "TC Test" ist nicht notwendig.

Zum InputGate von "TC Test" steht der TrustX Data Monitor (TDM) zur Verfügung. Dieser erlaubt, die übermittelten Testdaten zu verwalten. Durch die Authentisierung mittels HIN-Client ist gewährleistet, dass nur der Sender der Testdaten auf diese zugreifen kann.

Die URL für den Einstieg lautet: <http://iglog-test.trustx.hin.ch/cgi-bin/sitemgr.pl>

Die TrustCenter-Kunden haben zudem die Möglichkeit, die angelieferten Daten im InputGate LogViewer zu betrachten. Der InputGate LogViewer stellt zusätzlich Aussagen zur Qualität der Rechnung zur Verfügung.

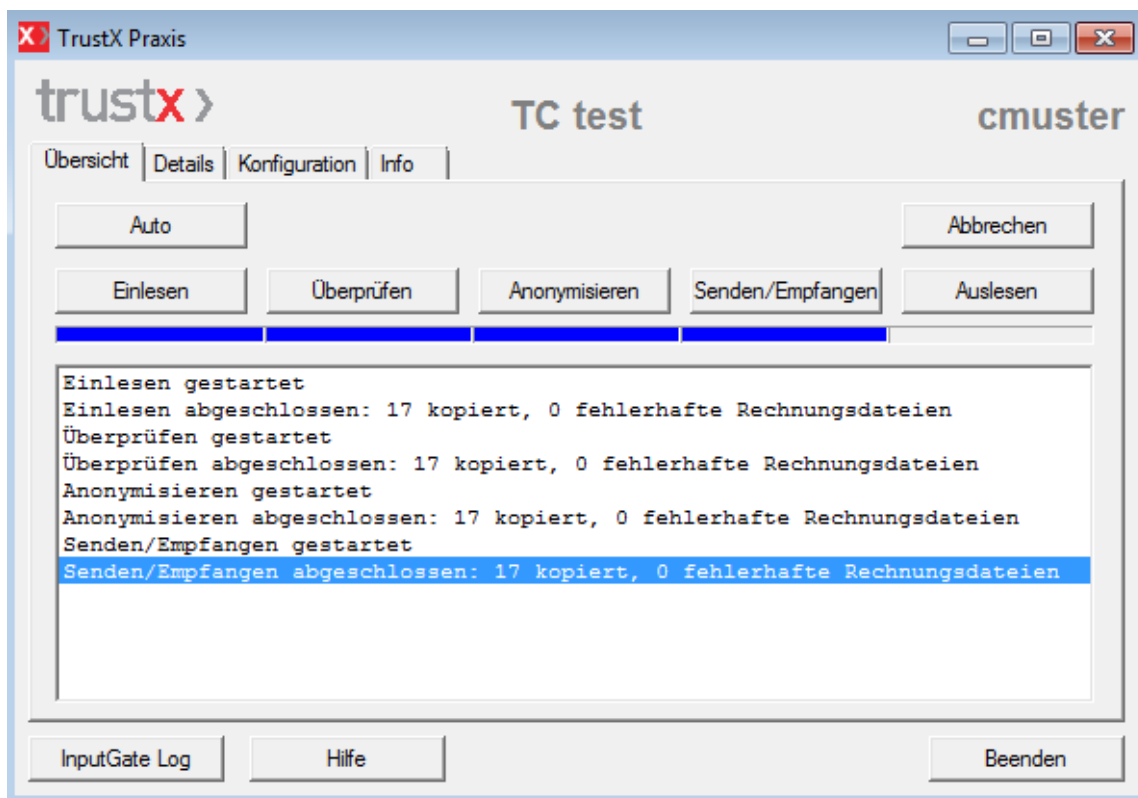
5 Funktionale Beschreibung von TrustX Praxis

TrustX Praxis ist eine Software-Komponente mit eigenem User-Interface und einer Schnittstelle zur direkten Ansteuerung aus dem Praxissoftwaresystem (siehe Kap. 6 „Schnittstelle zu TrustX Praxis“).

TrustX Praxis beinhaltet eine Auto-Update-Funktion über das Internet. Das Update kann zudem auch direkt über die Schnittstelle gesteuert werden. Die neue Version wird im Anschluss an den Datentransfer zum TrustCenter auf den PC geladen, falls der Benutzer damit einverstanden ist. TrustX Praxis kann so mit minimalem Aufwand auf dem neusten Versionsstand gehalten werden.

Die Bedienungsoberfläche von TrustX Praxis zeigt im Kopfbereich das ausgewählte TrustCenter sowie das HIN-Login und ist in vier Register gegliedert:

<Übersicht>	Beinhaltet alle wichtigen Funktionen zur Bedienung
<Details>	Gibt Einblick in verschiedenen Verzeichnisse und Log-Dateien
<Konfiguration>	Erlaubt die Konfiguration von TrustX Praxis
<Info>	Gibt über TrustX Praxis Auskunft (z.B. aktuelle Version)



Unten links befindet sich ein Button, mit welchem der InputGate Log Viewer aufgerufen werden kann, über welchen die Rechnungsanlieferung online eingesehen werden kann.

5.1 Register <Übersicht>

Das Register <Übersicht> umfasst alle wesentlichen Informationen zum Verarbeiten und Senden von Rechnungsdaten an das InputGate Ihres TrustCenters. TrustX Praxis kann in zwei Varianten genutzt werden:

5.1.1 Variante "Auto"

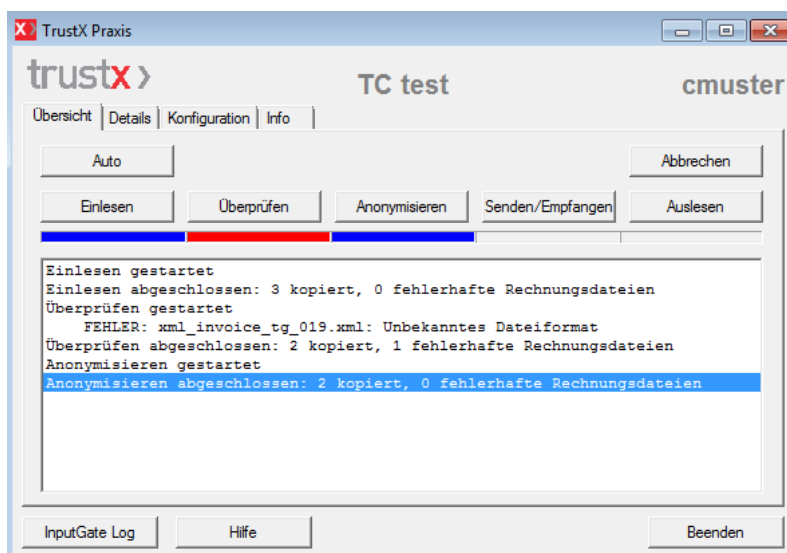
Über den Button <Auto> werden alle Schritte mit den vorgewählten Konfigurationsparametern automatisch abgearbeitet. Der blaue Balken unter den Buttons zeigt den Fortschritt der Verarbeitung an.

5.1.2 Variante "Step by Step"

Die Variante "Step by Step" ermöglicht es, schrittweise vorzugehen. Die einzelnen Schritte werden über die entsprechenden Buttons ausgelöst. Es werden ebenfalls die vorgewählten Konfigurationsparameter verwendet.

<Einlesen>	Daten werden aus dem Rechnungsverzeichnis (im Register <Konfiguration> definiert) eingelesen und dort anschliessend gelöscht.
<Überprüfen>	Daten werden auf formale Korrektheit überprüft.
<Anonymisieren>	Daten werden verschlüsselt und anonymisiert.
<Senden/Empfangen>	Daten werden ans InputGate des TrustCenters gesendet (Rechnungen) oder vom ResponseGate empfangen (Mahnungen). Die Internetverbindung wird – falls notwendig – durch TrustX Praxis gemäss den Einstellungen im Internet Explorer aufgebaut und wieder geschlossen.
<Auslesen>	Daten werden ins Antwortverzeichnis (im Register <Konfiguration> definiert) verschoben.

5.1.3 Darstellung des Fortgangs



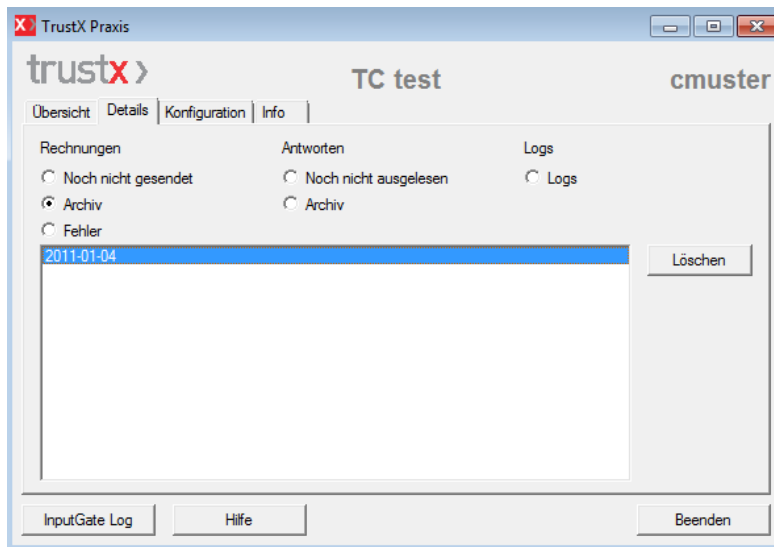
Im weissen Bereich erscheint eine Zusammenfassung über die verarbeiteten Dateien.

Allfällige Fehler werden im entsprechenden Verarbeitungsschritt pro Rechnung angezeigt.

Durch Doppelklicken auf die entsprechende fehlerhafte Rechnung werden die Rechnung und die detaillierten Log-Einträge dargestellt.

Siehe auch Kap. 5.2 „Register <Details>“.

5.2 Register <Details>



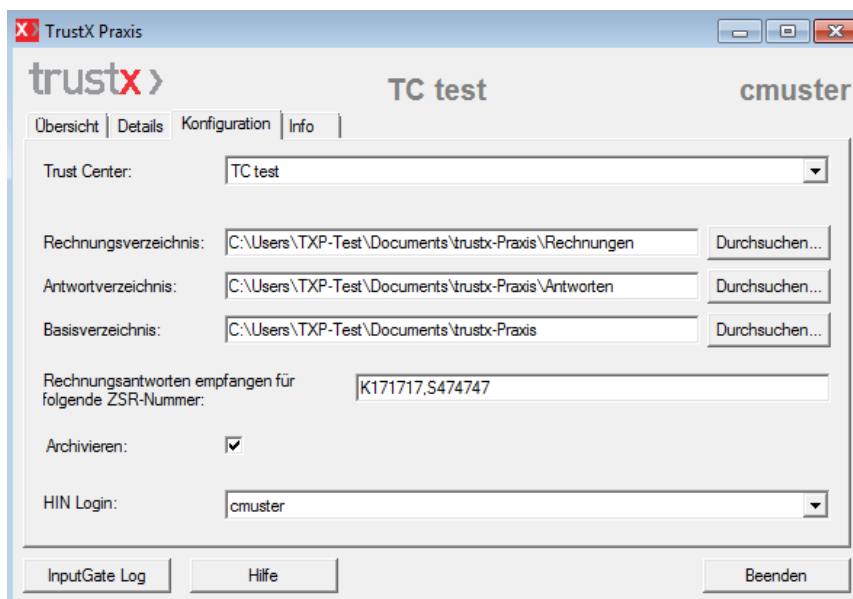
Das Register <Details> stellt verschiedene Detailansichten zur Verfügung.

Über die Auswahlfelder (oben) können entsprechende Listen im darunter liegenden Fenster dargestellt werden.

Durch Doppelklicken auf einen Listeneintrag werden die entsprechenden Rechnungen und die zugehörigen Log-Einträge dargestellt.

- <Noch nicht gesendet> Die bereitstehenden (noch nicht bearbeiteten) Rechnungen, die sich im "Rechnungsverzeichnis" befinden, werden aufgelistet.
- <Archiv> Die erfolgreich gesendeten Rechnungen werden aufgelistet.
- <Fehler> Fehlerhafte Rechnungsdateien aus Schritt 2 <Überprüfen> und Schritt 3 <Anonymisieren> werden hier aufgelistet.
- <Logs> In der Liste erscheinen die vorhandenen Logs, sortiert nach Datum.

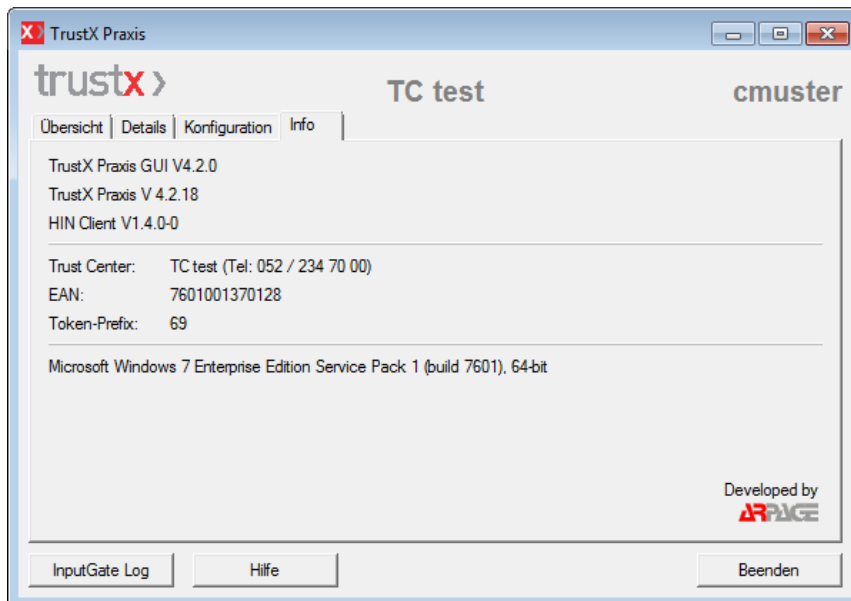
5.3 Register <Konfiguration>



Im Register <Konfiguration> können alle Parameter definiert werden, die für die Verarbeitung und das Versenden von Rechnungsdaten notwendig sind.

<TrustCenter>	Wählen Sie Ihr TrustCenter aus der Liste aus. Für Tests verwenden Sie das "TC Test".
<Rechnungsverzeichnis>	Die Rechnungen werden vom <Rechnungsverzeichnis> ins Verzeichnis "input" verschoben (Schritt "Einlesen"). Das Verzeichnis "input" ist von TrustX Praxis fix vorgegeben und befindet sich im Basisverzeichnis.
<Antwortverzeichnis>	Die Rechnungsantworten werden aus dem Verzeichnis "responses" ins <Antwortverzeichnis> verschoben (Schritt "Auslesen"). Das Verzeichnis "responses" ist von TrustX Praxis fix vorgegeben und befindet sich im Basisverzeichnis.
<Basisverzeichnis>	Unter "Basisverzeichnis" wird ein Verzeichnis vorgegeben, in dem TrustX Praxis verschiedene Unterverzeichnisse anlegt, die zum Betrieb von TrustX Praxis notwendig sind. Siehe Kap. 6.6 "Verzeichnisstruktur".
<Rechnungsantworten>	<p>Tragen Sie hier Ihre ZSR-Nummer(n) ein, für welche Sie Rechnungsantworten empfangen möchten.</p> <ul style="list-style-type: none"> ▪ leer: es werden keine Antworten gesucht ▪ "*": es werden alle Antworten für den gewählte HIN-Login gesucht ▪ ZSR1,ZSR2...: "Komma-delimited" Liste von ZSR-Nummern. Es werden alle Antworten für die angegeben ZSR-Nummern und den eingestellten HIN-Login gesucht
<Archivieren >	Die Option "Archivieren" steuert, ob erfolgreich versendete Rechnungsdateien archiviert oder gelöscht werden. Archivierte Rechnungen liegen in jener Form vor, in der sie versendet worden sind, d.h. anonymisiert und verschlüsselt.
<HIN Login>	In diesem Feld wird definiert, mit welchem HIN-Login TrustX Praxis für die Verschlüsselung und den Datentransfer arbeitet.

5.4 Register <Info>



Das Register <Info> zeigt die Versionsnummern von TrustX Praxis und des HIN-Clients sowie weitere Informationen an.

5.5 Mandanten in TrustX Praxis

TrustX Praxis ist mandantenfähig – alle Daten (Rechnungen, Rechnungsantworten, Logfiles, Einstellungen) werden im jeweiligen Benutzerprofil abgelegt. Dies ist insbesondere dann von Vorteil, wenn TrustX Praxis mit seiner eigenen Benutzeroberfläche bedient wird. Ein Umkonfigurieren erübrigt sich damit, sofern jeder Benutzer mit seinem eigenen Profil arbeitet.

Intermediäre können bei Bedarf mit einem eigenen Basisverzeichnis pro Mandant / TrustCenter arbeiten. Dieses kann über die Schnittstelle gesetzt werden. Die Verwendung von mandantenspezifischen Verzeichnissen hat den Vorteil, dass z.B. fehlerhafte Rechnungen und Logfiles separat gespeichert werden. Die Weiterverarbeitung wird dadurch vereinfacht.

6 Schnittstelle zu TrustX Praxis

6.1 Einleitung

TrustX Praxis kann von Anwendungen über eine Schnittstelle angesteuert werden. Diese ist im "OMG IDL" definiert (siehe Kap.- 6.2 "Schnittstellendefinition"). Aus dieser Definition werden die Schnittstellendefinitionen für die entsprechenden Betriebssysteme und Technologien generiert. Dies sind für Windows "COM" und für Mac OS X "Apple Events". TrustX Praxis generiert detaillierte Meldungen, die an die aufrufende Anwendung zurückgesendet werden (siehe Kap. 6.3 "Events").

6.2 Schnittstellendefinition

Hinweise:

- Änderungen gegenüber TrustX Praxis Version 3 sind gelb hervorgehoben
- Zum besseren Verständnis siehe auch Kap. 5 "Funktionale Beschreibung von TrustX Praxis"

Schnittstellendefinition	Bemerkung
<code>// \$Id: trustx.idl,v 1.6 2007/03/26 07:46:18 franco Exp \$</code>	
<code>#include "client.idl"</code>	
<code>module Trustx {</code>	
<code> interface ITrustx {</code>	
<code> typedef sequence<string> strings;</code>	
<code> readonly attribute string trustxVersion</code>	Version von TrustX Praxis
<code> readonly attribute strings trustCenters</code>	Liste aller TrustCenter
<code> attribute string trustCenter</code>	TrustCenter auswählen
<code> attribute string inputDirectory</code>	Pfad für Rechnungsverzeichnis
<code> attribute string workDirectory</code>	Pfad für Basis- oder Arbeitsverzeichnis
<code> attribute boolean withPDF;</code>	Option elektronische Rechnungskopie und PDF-Datei
<code> attribute boolean withArchive;</code>	Option verschickte Rechnungen archivieren
<code> readonly attribute string esr;</code>	TrustCenter-Code für "Dokumenten-Identifizier ohne PDF" lesen, abhängig vom gewählten TrustCenter
<code> readonly attribute string esrPDF;</code>	TrustCenter-Code für "Dokumenten-Identifizier mit PDF" lesen
<code> readonly attribute string session;</code>	als Sessionbezeichnung wird das Datum verwendet (Bsp.: 2003-08-29). Wird benutzt, um Namen von Unterverzeichnissen und Dateien zu bilden. (Bsp.: ...{\basisverzeichnis}\logs\<session>.log
<code> readonly attribute string asasVersion;</code>	Version des HIN-Client lesen
<code> readonly attribute strings asasLogins;</code>	Liste mit verfügbaren HIN Logins anzeigen
<code> attribute string asasLogin;</code>	HIN Login für TrustX Praxis vorgeben

Schnittstellendefinition	Bemerkung
attribute boolean activateGUI;	Zurzeit nicht implementiert, siehe Kap. 4.3.2 "Aufruf von TrustX Praxis aus der Praxissoftware".
attribute boolean debug;	Debug-Meldungen generieren ja/nein
void Auto();	Automatisches Verarbeiten und Senden der Rechnungsdateien analog dem Button <Auto>
void Read();	Einlesen der Daten analog dem Button <Einlesen>, Rechnungsdateien werden aus ..\{Rechnungsverzeichnis}\.. in ..\{basisverzeichnis}\input verschoben
void Check();	Daten werden auf formale Korrektheit geprüft. Daten werden aus ..\{basisverzeichnis}\Input gelesen und, wenn korrekt in ..\{basisverzeichnis}\valid, sonst in ..\{basisverzeichnis}\err geschrieben.
void Anonymize();	Rechnungsdateien werden anonymisiert (Modul moPAnonymizer) und verschlüsselt (Modul moPEncrypter). Gelesen wird aus ..\{basisverzeichnis}\valid. Geschrieben wird in ..\{basisverzeichnis}\output, resp. im Fehlerfall in ..\{basisverzeichnis}\err
void Send();	Rechnungsdaten werden aus der Verzeichnis ..\{basisverzeichnis}\output ans Inputgate des TrustCenters geschickt. Sind Rechnungsdaten und Bild vorhanden, werden diese zusammen versendet, so dass nur beide oder keines erfolgreich verschickt werden können. Es wird jedoch pro Datei eine entsprechende Meldung ausgegeben.
void Cancel();	Verarbeitung wird abgebrochen
void Write();	Empfangene Rechnungsantworten werden aus ..\{basisverzeichnis}\responses ins eingestellte Antwortverzeichnis (outputDirectory) kopiert. (Seit Version 2.2)
// Added in V2.1	
readonly attribute string ean;	EAN-Nummer des TrustCenters lesen, abhängig vom gewählten TrustCenter
readonly attribute string tcTel;	Telefon-Nummer des TrustCenters lesen, abhängig vom gewählten TrustCenter (für Supportanfragen)
// Added in V2.2	
attribute string outputDirectory;	Pfad für Rechnungsantwortverzeichnis
readonly attribute string igLog;	Link auf den InputGate Log Viewer
attribute boolean autoUpdate;	Gibt an, ob trustx.dll SW-Updates selber behandelt. Default ist „true“
readonly attribute updateAvailable;	Gibt an, ob ein SW-Update vorhanden ist
attribute boolean responseMessagesEnabled;	Gibt an, ob Meldungen, die mit Rechnungsantworten zu tun haben, generiert werden. Default ist "false"
attribute string zsrDownFilter;	<p>Filter für die Suche von Rechnungsantworten. Angabe von ZSR Nummern, für welche Rechnungsantworten empfangen werden sollen. Mögliche Werte:</p> <p>leer es werden keine Antworten gesucht</p> <p>* es werden alle Antworten für das eingestellte HIN-Login gesucht</p> <p>ZSR1,ZSR2... Liste von "Komma-delimited" ZSR-</p>

Schnittstellendefinition	Bemerkung
	Nummern. Es werden alle Antworten für die angegebenen ZSR-Nummern und für den eingestellten ASAS-Login gesucht.
<code>// Added in V3</code>	
<code>readonly attribute string getUpdateMsg;</code>	Gibt die Update-Meldung aus, welche auf dem Server hinterlegt ist.
<code>// Added in V4</code>	
<code>// Added in V4</code>	
<code>void OpenSession()</code>	Öffnet eine Session zur effizienten Verarbeitung von mehreren Rechnungen mittel AutoOne.
<code>void AutoOne(string name_of_invoice)</code>	Verarbeitung einer (1) Rechnung vom Einlesen bis zur Übermittlung Das Sessionhandling ist in der Verantwortung des Schnittstellenbenutzers (siehe OPenSession, CloseSession)
<code>void CloseSession()</code>	Schliesst eine Session, welche mit OpenSession eröffnet wurden.
<code>};</code>	
<code>};</code>	

6.3 Events

Jedes TrustX-Modul in TrustX Praxis generiert für jede bearbeitete Datei Meldungen, die an die aufrufende Anwendung zurückgesendet werden. Die Meldungen sind klassifiziert nach Mitteilungs-Level, -Class und -Ursprung und betreffen in der Regel eine bestimmte Datei. Damit kann genau verfolgt werden, wie weit die Verarbeitung in welchem Modul mit welcher Datei fortgeschritten ist.

Zu Beginn und als Abschluss eines Moduls wird eine entsprechende Meldung mit dem Mitteilungslevel mlInfo ausgegeben. Dazwischen wird für jede Datei zumindest eine explizite Meldung mit MessageLevel "mlInfo & MessageOrigin" des Moduls und dem entsprechenden Code für OK oder eine Fehlermeldung gesendet. Die Event-Klassifizierung sieht wie folgt aus:

6.3.1 Klassifizierung der Events

Klassifizierung	Bemerkung
<code>enum MsgLevel {</code>	Message Level
<code>mlDebug = 0,</code>	Diese Meldungen können im Normalbetrieb ignoriert werden. Sie dienen der Fehlersuche und -analyse.
<code>mlInfo = 1,</code>	Diese Meldungen informieren über den aktuellen Stand der Verarbeitung (z.B. Datenübermittlung gestartet oder OK).
<code>mlWarning = 2,</code>	Mit diesen Meldungen werden Abweichungen vom erwarteten „Normalfall“ beschrieben. Das sind keine Fehler, sie könnten aber zu Folgefehlern führen und sollten abgeklärt werden (z.B. keine Daten im Verzeichnis).
<code>mlError = 3,</code>	Diese Meldungen betreffen Fehler, die behoben werden müssen, um die Verarbeitung durchzuführen (z.B. Modem nicht eingeschaltet).
<code>mlFatal = 4,</code>	Diese Meldungen sollten nie vorkommen. Sie betreffen/beschreiben Fälle, die eine Verarbeitung verunmöglichen (z.B. fehlende DLL)
<code>};</code>	

Klassifizierung	Bemerkung
-----------------	-----------

Klassifizierung	Bemerkung
enum MsgClass {	Message Class
mcSystem = 0,	Meldung kommt vom Betriebssystem (z.B. Schreibfehler) <code> wird vom Betriebssystem übernommen
mcASAS = 1,	Meldung kommt von der ASAS-Infrastruktur (z.B. Zertifikat nicht gefunden); (nur Module moPEncrypter, moPSender)
mcModule = 2,	Meldung kommt von einem trustx-Modul (z.B. Anonymisieren gestartet)
mcCount = 3,	einmal pro Modul sofort nach Start (Anzahl Dateien im Code)
};	

Klassifizierung	Bemerkung
enum MsgOrigin {	Message Origin
moPReader = 0,	Meldung kommt vom TrustX Praxis-Modul - Reader
moPChecker = 1,	Meldung kommt vom TrustX Praxis-Modul - Checker
moPAnonymizer = 2,	Meldung kommt vom TrustX Praxis-Modul - Anonymizer
moPEncrypter = 3,	Meldung kommt vom TrustX Praxis-Modul - Encrypter
moPSender = 4,	Meldung kommt vom TrustX Praxis-Modul - Sender
moIGReceiver = 5,	Meldung kommt vom TrustX-InputGate-Modul - Receiver
moIGChecker = 6,	Meldung kommt vom TrustX-InputGate-Modul - Checker
moPDownloader = 7,	Meldung kommt vom TrustX Praxis-Modul - Downloader
moPWriter = 8,	Meldung kommt vom Auslese-Modul
};	

6.3.2 Schnittstelle zum Empfangen der Events

Um Event-Meldungen von TrustX Praxis zu empfangen, muss die Anwendung eine einzige Schnittstelle implementieren, die wie folgt aussieht:

Empfang von Events	Bemerkung
// \$Id: client.idl,v 1.3 2003/09/01 06:26:26 franco Exp \$	
#include "enums.idl"	
module TrustxClient {	
interface TrustxEvents {	
oneway void trustxEvent(in MsgLevel level,	Message Level
in MsgClass mclass,	Message Class
in MsgOrigin origin,	Message Origin
in long code,	Code, muss zusammen mit Message Class interpretiert werden. Eigentlicher "Fehlercode"
in string fileName,	gibt Datei an, auf welche sich die Meldung bezieht, kann leer sein, wenn sich die Meldung auf Modul als Ganzes bezieht
in string msg);	Beschreibung des Fehlercodes mit Text
};	
};	

Falls level=mlError und mclass=mcModule, so wird die betreffende Datei ins Error-Verzeichnis verschoben (Problem liegt am Inhalt der Datei).

Bei allen anderen Fehlerkombinationen bleibt die Datei im Arbeitsverzeichnis (Fehler liegt nicht an der Datei). Nach Behebung der Fehlerursache kann die Bearbeitung wieder gestartet werden, ohne dass die Rechnung nochmals neu erstellt werden muss. Beispiel: Modem nicht eingeschaltet. Eine entsprechende Fehlermeldung wird mit MessageClass=mcASAS mitgeteilt.

6.3.3 Messages (Code)

Die Codes sind für jede Message Class unterschiedlich:

- mcSystem: der Code enthält den Fehlercode des Betriebssystems
- mcASAS: Fehler Code ist immer -1, weil HIN keine Fehlercodes kennt. Die eigentliche Fehlermeldung wird in den String msg geschrieben
- mcCount: enthält im Code die Anzahl der zu verarbeitenden Dateien
- mcModule: es gelten die nachfolgenden Codes:

Klassifizierung	Bemerkung
enum Msgs {	Messages (Code)
INFO_MODULE_START = 0,	Bezeichnet den Beginn eines Moduls
INFO_MODULE_DONE = 1,	Bezeichnet das Ende des Moduls
INFO_MOVED = 2,	File erfolgreich verarbeitet und ins nächste Verzeichnis verschoben
WARN_NOT_MOVED = 3,	File erfolgreich verarbeitet, konnte aber nicht ins Verzeichnis verschoben werden
WARN_NO_FILES = 4,	Keine Files im "In-Verzeichnis" des Moduls gefunden
WARN_MODULE_USERBREAK = 5,	Modul durch User abgebrochen, es folgt zusätzlich Code 1
ERR_CONF_INPUTDIR = 6,	Konfiguration: Rechnungsverzeichnis nicht gefunden
ERR_CONF_WORKDIR = 7,	Konfiguration: Basisverzeichnis nicht gefunden
ERR_CONF_ASASLOGIN = 8,	Konfiguration: ASAS-Login nicht gefunden
ERR_CONF_TRUSTCENTER = 9,	Konfiguration: TrustCenter nicht gefunden
ERR_UNKNOWN_FILE_FORMAT = 10,	Unbekanntes Fileformat (Checker von TrustX Praxis und Inputgate)
ERR_MISSING_DATA = 11,	Bekanntes Format, aber Daten fehlen oder sind nicht konform
ERR_DUPLICATE_INVOICE = 12,	Doppelte Rechnung (Checker)
ERR_PDF_MISSING = 13,	Nur wenn Option PDF=ja, PDF fehlt
ERR_PDF_UNREF = 14,	Nur wenn Option PDF=ja, überzähliges PDF
ERR_ANONENC_FAILED = 15,	Fehler beim Anonymisieren oder Verschlüsseln
ERR_CONNECTION_FAILED = 16,	Internetverbindung fehlgeschlagen
ERR_SEND_FAILED = 17	Senden ist gescheitert
FATAL_ASAS_NOT_FOUND = 18,	Kein Zugriff auf ASAS-Client
FATAL_MODULE_LOAD_FAILED = 19,	TrustX-Modul konnte nicht geladen werden
ERR_SCHEMA = 20,	XML-Rechnung ist nicht schemakonform
ERR_ZSR_EAN = 21,	ZSR-Nummer ist ungültig (Invoice rejected, no ZSR/EAN found)

Klassifizierung	Bemerkung
ERR_ENC_SEND_TC_MISMATCH = 22,	Rechnung wurde bereits für ein anderes TrustCenter verschlüsselt
ERR_CONF_OUTPUTDIR = 23,	Konfiguration: Antwortverzeichnis nicht gefunden
};	

6.4 TrustX einbinden

Sämtliche Prozesse (Module) wie Auto, Read, usw. arbeiten asynchron. Dies bedeutet, dass das aufrufende Programm sofort vom entsprechenden Modul einen Event mit Code "INFO_MODULE_START" erhält. Damit wird der Start des Moduls gekennzeichnet.

Es gilt immer folgender formaler Ablauf:

Event, der Start des Moduls bestätigt (Code = INFO_MODULE_START).

Anschliessend folgt genau ein Event der Klasse Count. Der Code enthält die Anzahl zu verarbeitender Files.

Pro File erfolgt ein Event mit Code INFO_MOVED oder ein oder mehrere Events mit einer Warnung oder Fehlermeldung.

Das Modul wird immer mit einem Event mit Code INFO_MODULE_DONE beendet.

Der Fortschritt der File-Verarbeitung kann über die Meldungen von TrustX-Event verfolgt werden. Unbedingt zu beachten ist, dass das Event-Konzept von Windows nicht erlaubt, direkt aus Events Befehle an den Bildschirm (z.B. Progress-Bars) abzusetzen (siehe Beispiel in Kap. 7.2 "TrustX-Event Beispiel (VB6)").

Wird ein Prozess durch den Benutzer abgebrochen, wird das sich in Bearbeitung befindende File noch fertig verarbeitet. Anschliessend wird das Modul beendet.

6.5 Dateibenennung

Rechnungsdateien können beliebige Dateinamen haben.

6.6 Verzeichnisstruktur

In TrustX Praxis kann ein Basisverzeichnis {bz} vorgegeben werden. Darunter werden für den TrustX-internen Gebrauch folgende Verzeichnisse angelegt:

..\{bz}\input	Im Schritt <Einlesen> werden die zu verarbeitenden Rechnungsfiles aus einem beliebigen Verzeichnis in dieses Verzeichnis verschoben. In diesem Verzeichnis beginnt der eigentliche Verarbeitungsprozess von TrustX Praxis. Die Methode check() liest aus diesem Verzeichnis.
..\{bz}\responses	In dieses Verzeichnis gelangen die Rechnungsantworten, welche von TrustX empfangen werden. Die Methode write() liest die Dateien aus diesem Verzeichnis und legt sie im eingestellten Rechnungsantwortverzeichnis ab.
..\{bz}\valid	In dieses Verzeichnis gelangen alle durch Check() erfolgreich verarbeiteten Files. Die Methode Anonymize() liest Daten aus diesem Verzeichnis.

..\{bz}\output	In dieses Verzeichnis gelangen alle durch Anonymize() erfolgreich verarbeiteten Files. Die Methode Send() liest Daten aus diesem Verzeichnis.
..\{bz}\archive	In diesem Verzeichnis werden alle erfolgreich ans InputGate versendeten Rechnungen abgespeichert, sofern die Option Archivieren im Register <Konfiguration> angewählt wurde
..\[bz]\responses-archive	In diesem Verzeichnis werden alle empfangenen Rechnungsantworten abgespeichert, sofern die Option Archivieren im Register <Konfiguration> angewählt wurde.
..\{bz}\logs	Enthält die Logfiles.
..\{bz}\err	In dieses Verzeichnis gelangen alle durch Check() oder Anonymize() verarbeiteten Files, die einen Fehler beinhalten, so dass das entsprechende File definitiv nicht verarbeitet werden kann und neu erstellt werden muss.

Vom Basisverzeichnis sind das Rechnungs- und das Antwortverzeichnis zu unterscheiden. Diese können unabhängig vom Basisverzeichnis gewählt werden.

6.7 Wichtige Dateien von TrustX Praxis v4

Keine Änderungen gegenüber V3. Siehe Beschreibung unter V3.

6.8 Wichtige Dateien von TrustX Praxis v3

6.8.1 Programmfiles

trustxui.exe	TrustX Praxis Anwendung / Exe-Datei (Windows)
msvcvp71.dll	DLLs von TrustX Praxis
msvcvr71.dll	
trustx.dll	
trustx-Praxis	TrustX Praxis Anwendung / ausführbare UNIX-Datei (Mac)

6.8.2 Konfigurations- und Logfiles

TrustX Praxis legt verschiedene Files an, um Konfigurationen und Logdaten zu speichern.

Filename	Inhalt	pro Mandant	Pfad
trustx.log	Rechnungs- informationen, welche für den Doublettencheck	Ja	Im Root vom eingestellten Basisverzeichnis, i.d.R. Windows: C:\Dokumente und Einstellungen\MEIN-

	benötigt werden.		PROFIL\Eigene Dateien\trustx-Praxis\ Mac: /Users/MY_USER/Documents/trustx-Praxis
YYYY-MM-DD.log	Detaillierte Log- informationen zu jedem Ver- arbeitungsschritt	Ja	Im Verzeichnis „logs“ unter dem eingestellten Basisverzeichnis, z.B. Windows: C:\Dokumente und Einstellungen\MEIN- PROFIL\Eigene Dateien\trustx- Praxis\logs Mac: /Users/MY_USER/Documents/trustx- Praxis/logs
trustx.ini ch.trustx.praxis.fm wk.plist	Konfigurations- einstellungen	Ja	Windows: C:\Dokumente und Einstellungen\MEIN- PROFIL\Anwendungsdaten\Arpage\trustx- Praxis Mac: /Users/MY_USER/Library/Preferences/ch.t rustx.praxis.fmwk.plist
trustx.xml (2x)	Informationen über die TrustCenter → wird bei der Installation angelegt	Nein	Windows: C:\Programme\Arpage\TrustX Praxis Mac: /Applications/trustx- Praxis.app/Contents/SharedFrameworks/tr ustx.frameworks/Versions/A/Resources/tr ustx.xml
		Ja	Windows: C:\Dokumente und Einstellungen\MEIN- PROFIL\Anwendungsdaten\Arpage\trustx- Praxis Bzw. C:\Users\MEIN- PROFIL\AppData\Roaming\Arpage\trustx- praxis Mac: /Users/MY_USER/Library/Application Support/trustx-Praxis/trustx.xml

7 Anhänge

7.1 ESR-Zeile als Dokumenten-Identifizier (Token)

Als Dokumenten-Identifizier (Token) wird eine ESR-Zeile verwendet, die sich von der ESR-Zeile der Rechnung ableitet. ESR-Zeile: esr9-Verfahren mit 27-stelliger ESR-Ref.-Nr. und Betrag.

An Stelle der Belegart (erste zwei Stellen (A)) wird das TrustCenter codiert. Ebenfalls angepasst werden muss die entsprechende Prüfziffer (C).

Das Token wird auf den Rückforderungsbeleg gedruckt. Für das elektronische Bild der Rechnung dient das Token als Filename (Steuerzeichen durch "_" ersetzen und Leerstellen weglassen).

7.1.1 Spezifikation ESR-Zeile als Token

Die ESR-Zeile, die als Token verwendet wird, entspricht den Vorgaben von SWIFT für ESR (OCR-B Schrift, Positionierung etc.). Das heisst insbesondere:

- Das Token muss immer eindeutig sein. Korrigierte Rechnungen müssen eine andere ESR-Nummer haben, als die erste Rechnung. Dies wird über eindeutige Rechnungsnummern erreicht, wie dies auch im XML-Standard vorgegeben ist.
- Das Token muss auf dem Rückforderungsbeleg des Patienten, welcher beim Versicherer für die Rückerstattung eingereicht wird, an der gleichen Stelle wie auf dem Einzahlungsschein aufgedruckt sein. So kann auf Seite des Versicherers das Token mit konventionellen Lesegeräten einlesen werden.

Bitte beachten Sie betreffend der Positionierung die Richtlinien der Post.

50	0000157050	6	>	332435	00000111112222233333	1	+	010033018	>
A	B	C	D	E	F	G	H	I	J

A	=	TrustCenter	2 Stellen
B	=	Betrag in Rappen rechtsbündig, mit Nullen aufgefüllt	10 Stellen
C	=	Prüfziffer der Felder A u. B (Modulo 10 rekursiv)	1 Stelle
D	=	Steuerzeichen	1 Stelle
E+F	=	26-stellige Referenznummer (Lastdaten) [eindeutig !!!]	26 Stellen
G	=	Prüfziffer der Felder E und F	1 Stelle
H	=	Steuerzeichen und Leerstelle	2 Stellen
I	=	ESR-Teilnehmernummer	9 Stellen
J	=	Steuerzeichen	1 Stelle

7.1.2 Codierung der TrustCenter – Token-Präfix

Der Token, welcher auf den Rückforderungsbeleg gedruckt wird, wird mit einem Token-Präfix codiert. Nachfolgend die Liste aller TrustCenter.

TrustCenter	Token-Präfix
Ctésias	51
GallOnet (eastcare)	52
Hawatrust	53
Medkey	54
PonteNova	55
syndata	56
TC Aargau	57
thurcare (eastcare)	58
TC Ticino	59
Züridoc	60
Trustmed	61
GTS (General Transport Service)	62
Physiotrust	63
TC Test	69

7.2 TrustX-Event Beispiel (VB6)

Der Status INFO_MODULE_DONE kommt immer als letzte Meldung im Event-Ablauf. Die Status-Meldung kommt auch im Error-Fall.

```

Const INFO_MODULE_START = 0
Const INFO_MODULE_DONE = 1
Const INFO_MOVED = 2
Const WARN_NOT_MOVED = 3
Const WARN_NO_FILES = 4
Const WARN_MODULE_USERBREAK = 5
Private Sub trustx_trustxEvent(ByVal level As trustx.tagMsgLevel, _
                               ByVal mclass As trustx.tagMsgClass, _
                               ByVal origin As trustx.tagMsgOrigin, _
                               ByVal code As Long, _
                               ByVal fileName As String, ByVal msg As String)

    log.AddItem (msg)
    log.ListIndex = log.NewIndex

    If mclass = mcModule Then

        If code = INFO_MODULE_START Then
            running = True
            curMod = origin

            If origin = moPReader Then
                action.Text = "Einlesen"
            ElseIf origin = moPChecker Then
                action.Text = "Überprüfen"
            End If
        End If
    End If

```

```
ElseIf origin = moPAnonymizer Then
    action.Text = "Anonymisieren"
ElseIf origin = moPSender Then
    action.Text = "Senden"
End If

ElseIf code = INFO_MODULE_DONE Then
    If waitFor = origin Then
        running = False
        action.Text = "Fertig"
    End If

ElseIf (code <> WARN_NO_FILES) And _
    (origin = moPReader Or origin = moPChecker Or _
    origin = moPAnonymizer Or origin = moPSender) Then
    curTot = curTot + 1
    counter.Text = curTot & "/" & curMax
End If
End If

If mclass = mcCount Then
    curTot = 0
    curMax = code
    counter.Text = curTot & "/" & curMax
End If
End Sub
```

7.3 Schematische Darstellung – Rechnungsdruck-Steuerung & eDA TP

