



# TX-Praxis Version 4 / 5

## Schnittstellenbeschreibung für Praxissoftware-Hersteller

Version 1.5.2  
10.11.2022



TrustX Management AG  
8400 Winterthur  
trustx@trustx.ch | www.trustx.ch

Das vorliegende Dokument ist für den internen Gebrauch bestimmt und deshalb vertraulich zu behandeln. Es darf nicht ohne die Zustimmung der Verfasser kopiert oder weitergeleitet werden.

---

## Inhaltsverzeichnis

---

<b>1</b>	<b>Einleitung</b>	<b>3</b>
1.1	TrustX	3
1.2	TrustCenter und NewIndex	3
1.3	HIN	3
1.4	Ansprechpartner	3
<b>2</b>	<b>TrustX in der Praxis</b>	<b>4</b>
2.1	Übersicht TrustX-System	4
2.2	Rechnungsanlieferung	4
2.3	TX Praxis	5
<b>3</b>	<b>Zusatzdienste</b>	<b>8</b>
3.1	Abrechnungsarten	8
3.2	Zusatzdienste eDA und Druck&Versand	9
3.3	Steuerung eDA und Druck&Versand bis XML 4.1	9
3.4	Steuerung eDA und Druck&Versand ab XML 4.3	10
3.5	eDA TG	11
<b>4</b>	<b>Anforderungen an die Praxissoftware</b>	<b>12</b>
4.1	Übersicht	12
4.2	Möglichkeiten zur Integration des TX Praxis Client	12
4.3	Doppelte Rechnungen / Mahnungen	13
4.4	Storno	15
4.5	Eindeutiges Token im eDA TG	15
4.6	TrustCenter «TC Test»	15
<b>5</b>	<b>Funktionale Beschreibung von TX Praxis Client</b>	<b>16</b>
5.1	Benutzeroberfläche	16
5.2	Mandanten in TX Praxis	20
5.3	Auto-Update-Funktion	20
<b>6</b>	<b>Spezifikation TX Praxis Client – Schnittstelle</b>	<b>21</b>
6.1	Schnittstellendefinition	21
6.2	Events	23
6.3	TX Praxis Client einbinden	25
6.4	Dateibenennung	26
6.5	Verzeichnisstruktur	26
6.6	Wichtige Dateien von TX Praxis Client	27
<b>7</b>	<b>Anhang – Codierung Token</b>	<b>29</b>
<b>8</b>	<b>Anhang – Beispiel TrustX-Event (VB6)</b>	<b>30</b>
<b>9</b>	<b>Anhang – Schema Steuerung eDA und Druck&amp;Versand ab XML 4.3</b>	<b>31</b>

---

## 1 Einleitung

---

Das vorliegende Dokument spezifiziert die TrustX-Schnittstelle zu den Praxissoftware-Systemen. Es stellt den Anbietern von Praxissoftware-Lösungen, Intermediären und Abrechnungsstellen, welche ihren Kunden den technischen Anschluss an den ärzteigenen Datenpool ermöglichen wollen, als Information und Dokumentation.

### 1.1 TrustX

TrustX entwickelt und betreibt den ärzteigenen Datensammlung im ambulanten Sektor, für und im Auftrag der Ärzteschaft. In die Datensammlung fließen die Abrechnungsdaten der teilnehmenden Arztpraxen. Die Datensammlung ist die Basis für die zahlreichen statistischen Auswertungen und Dienste zuhanden der Ärzteorganisationen und der Arztpraxen.

Ergänzend zur Datensammlung bietet TrustX als Zusatzdienste auch den elektronischen Austausch von Rechnungen mit den Versicherern sowie den Druck & Versand von Rechnungen an die Patienten.

Für die Teilnahme an der Ärzteneigenen Datensammlung schliessen sich die Arztpraxen an einem der regionalen, ärzteigenen TrustCenter oder der Zentralen Datensammelstelle der NewIndex an.

Das TrustX-System steht einheitlich allen Datenlieferanten zur Verfügung. Die Anlieferung von Rechnungsdaten an TrustX erfolgt immer unter dem TrustCenter (Mandant im TrustX-System), an welchem die Arztpraxis angeschlossen ist.

Neben der Ärzteneigenen Datensammlung entwickelt und betreibt TrustX analog zur Ärzteneigenen Datensammlung auch Datensammlungen für weitere Berufsgruppen.

### 1.2 TrustCenter und NewIndex

Die TrustCenter sind für den Anschluss der Arztpraxen an die Ärzteneigene Datensammlung, die Administration, den Support und die Beratung der angeschlossenen Kunden zuständig. Mehr Informationen zu den TrustCentern finden Sie auf den Webseiten der einzelnen TrustCentern.

Alternativ können sich Arztpraxen auch der Zentralen Datensammelstelle von NewIndex anschliessen. Informationen zu NewIndex finden Sie unter: [www.newindex.ch](http://www.newindex.ch).

### 1.3 HIN

Der Zugang zu den TrustX Diensten und Applikationen sind HIN geschützt (Authentifizierung, verschlüsselter Datenaustausch). Weitere Informationen zur der Health Infonet AG (HIN) finden Sie unter [www.hin.ch](http://www.hin.ch).

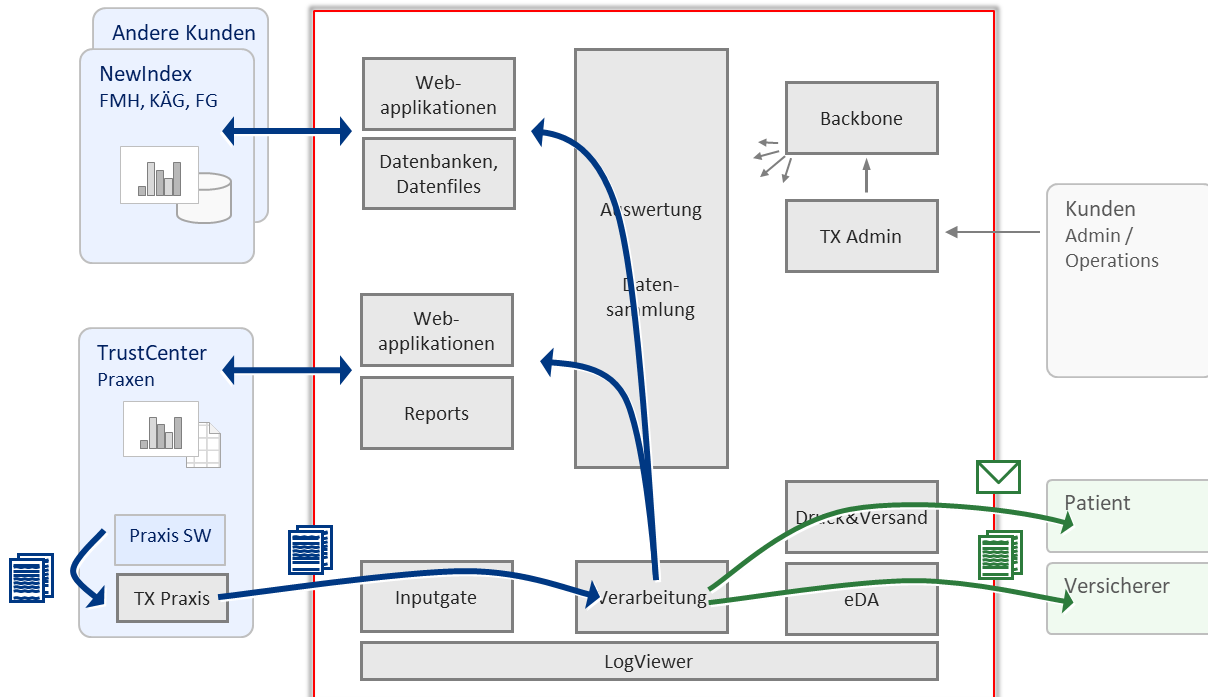
### 1.4 Ansprechpartner

Ansprechpartner für technische Fragen ist unser Support. Sie erreichen den Support zu den üblichen Geschäftszeiten unter 052 234 70 00 oder via E-Mail unter [support@trustx.ch](mailto:support@trustx.ch).

## 2 TrustX in der Praxis

### 2.1 Übersicht TrustX-System

Das TrustX-System umfasst eine Reihe von Diensten und Applikationen, wobei im Kern die Datensammlung und die darauf basierenden Auswertungen stehen. Als Zusatzdienste unterstützt TrustX den eDA mit den Versicherern und den Druck&Versand an die Patienten.



TrustX-System – Schematische Übersicht

Im Fokus des vorliegenden Dokuments stehen die Schnittstelle zwischen der Praxis bzw. der Praxissoftware und dem InputGate von TrustX sowie die zugehörigen Prozesse. Im Dokument finden sich weiter einige Erläuterungen zu den Zusatzdiensten und deren Nutzung über die TrustX-Schnittstelle.

### 2.2 Rechnungsanlieferung

Die elektronische Übermittlung von Rechnungen, Storni und Mahnungen (allgemein «Rechnungen») an TrustX erfolgt im XML-Format gemäss dem Forum für Datenaustausch.

Für die Übermittlung wird auf Seite des Senders TX Praxis benötigt. TX Praxis prüft den Absender (HIN-Authentifizierung), verschlüsselt die Patientenangaben und übermittelt die Rechnung dann via Internet (gesicherte HIN-Verbindung) ans Inputgate von TrustX. Bei Mahnungen wird zusätzlich in Echtzeit überprüft, ob die zugehörige XML-Rechnung im TrustX-System bekannt ist.

Beim Zusatzdienst eDA TP stellt TrustX zudem die Rechnungsantworten der Versicherer der Praxis zur Verfügung.

## 2.2.1 Sicherheit und Datenschutz

Zur Gewährleistung des Datenschutzes werden unter anderem die Dienste von HIN genutzt (Authentifizierung, Streckenverschlüsselung).

Weiter werden vor der Übermittlung an das TrustX-System die Patientenangaben verschlüsselt. Eine Entschlüsselung findet ausschliesslich in den folgenden zwei Fällen und als letzter Schritt in der Verarbeitung statt:

- Druck & Versand der Rechnung
- elektronische Übermittlung an den Versicherer. Die Versicherer sind für den eDA ebenfalls Datenschutzkonform an TrustX angebunden.

Das gleiche gilt für die Rechnungsantworten vom Versicherer an die Arztpraxis.

In der Datensammlung selbst sind die Patienten anonymisiert.

## 2.2.2 Technische Voraussetzungen zur Datenanlieferung

Um Rechnungsdaten an TrustX liefern zu können, sind folgende technischen Voraussetzungen notwendig:

- aktuelle TX Praxis Version (TX Praxis Client verfügbar unter [www.trustx.ch](http://www.trustx.ch))
- aktueller HIN-Client bzw. gültiges HIN Login (HIN Abonnement, [www.hin.ch](http://www.hin.ch))

Ab TX Praxis Version 5.x wird die HIN Open Authorization (OAuth 2.0) unterstützt. Damit entfällt die Notwendigkeit, für die Rechnungsanlieferung einen HIN Client installieren zu müssen.

- Internetzugang

## 2.2.3 Anlieferung mit Fakturierung

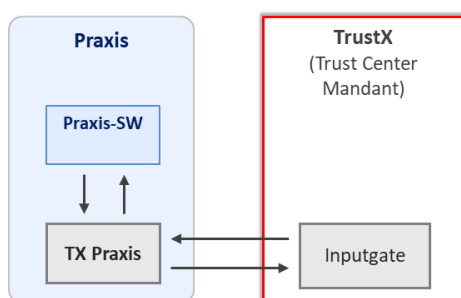
Grundsätzlich sind alle abgerechneten Leistungen unmittelbar nach der Fakturierung in der Praxis an TrustX zu übermitteln. Dies ist unter anderem aus folgenden Gründen wichtig:

- Nur wenn die Praxis alle Rechnungen vollständig übermittelt, erhält sie auch eine aussagekräftige Statistik ihrer Leistungen – und je schneller die Rechnungen angeliefert werden, desto schneller stehen diese der Praxis und in den Vergleichskollektiven, zum Beispiel im Praxisspiegel zur Verfügung.
- Die Verfügbarkeit und Qualität der zentralen Kollektivauswertungen wird deutlich verbessert

Bei TG Rechnungen ist es zudem wichtig, dass diese zusammen mit der Rechnungsstellung an den Patienten auch an TrustX übermittelt werden. Nur so stehen die Rechnungen für eine elektronische Abholung durch die Versicherer rechtzeitig zur Verfügung.

## 2.3 TX Praxis

TX Praxis stellt den Datenaustausch zwischen der Praxis und dem TrustX-System (Inputgate) sicher.



Der TX Praxis Client ist ein SW-Client mit eigener Benutzeroberfläche sowie einer Schnittstelle zur direkten Ansteuerung aus der Praxissoftware oder anderen SW-Lösungen. Mit der Schnittstelle kann TX Praxis vollständig in die Praxissoftware integriert werden.

TX Praxis nimmt im Einzelnen die folgenden Aufgaben wahr

- Lesen der Rechnungen aus dem konfigurierten Rechnungsverzeichnis
- Schemavalidierung der bereitgestellten Rechnungen
- Bestimmung eines anonymen Patienten-Codes (für die statistisch Auswertung benötigt) sowie Verschlüsselung der Patientenangaben (Angaben zum Patient, gesetzlichem Vertreter sowie zum Versicherten)
- Gesicherte Übermittlung der Rechnungen von der Praxis ans TrustX-System (Inputgate)
- Abholen von Rechnungsantworten vom Versicherer (eDA TP)

Nähere Informationen finden sich unter:

- Kap. 4.2, «Möglichkeiten zur Integration des TX Praxis Client»
- Kap. 5, «Funktionale Beschreibung von TX Praxis Client»
- Kap. 6, «Spezifikation TX Praxis Client – Schnittstelle»

### 2.3.1 Rechnungsantworten vom Versicherer

TrustX Praxis unterstützt den Tiers Payant vollumfänglich. Rechnungsantworten werden von TX Praxis im Anschluss an die Übermittlung der Rechnungen (Rechnungen, Storni, Mahnungen) automatisch empfangen und lokal im definierten Verzeichnis abgelegt. Voraussetzung ist die korrekte Konfiguration von TX Praxis. Siehe dazu auch Kap. 5.1.3 «Register <Konfiguration>» sowie Kap. 6.1 «Schnittstellendefinition».

Der Empfang von Rechnungsantworten ist nur relevant, wenn die Praxis den Zusatzdienst «eDA TP» gewählt hat.

### 2.3.2 Unterstützte Betriebssysteme

Der TX Praxis Client unterstützt grundsätzlich alle Betriebssysteme, welche von der eingesetzten HIN-Lösung unterstützt werden (Stand Juli 2018)

- **Windows:** ab Windows 7
- **Mac:** Mac OS X 10.10 und höher

Eine aktuelle Liste aller von HIN unterstützten Betriebssystemen finden Sie unter [www.hin.ch](http://www.hin.ch).

### 2.3.3 Unterstützte XML-Standards und Wertebereiche

Folgende XML-Standards werden von TX Praxis V4 und V5 unterstützt:

- **XML 4.5:** Generelle Rechnung und Rechnungsantwort inkl. Mahnung und Storno (generalInvoiceRequest/Response 4.5)
- **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)

Beim XML Standard 4.3 oder höher können Rechnungen mit den folgenden Werten angeliefert werden:

Attribut	Wertebereich	Kommentar
<b>body:role</b>	<b>physician</b> <del>hospital</del> <del>other</del> physiotherapist chiropractor ergotherapist nutritionist midwife logotherapist pharmacist dentist labtechnician dentaltechnician othertechnician psychologist wholesaler nursingstaff transport druggist naturopathicdoctor naturopathictherapist	Gilt für XML Standard 4.3 oder höher. «hospital» und «other» sind nicht zulässig. Werte in grau sowie «chiropractor» werden bei der Arztpraxis nicht unterstützt.
<b>body:place</b>	<b>practice</b> <del>hospital</del> lab association company	Gilt für XML Standard 4.3 oder höher. «hospital» ist nicht zulässig. Werte in grau werden bei der Arztpraxis nicht unterstützt.
<b>payload</b>	<b>Invoice, reminder</b> <del>credit</del>	Gilt für XML Standard 4.3 oder höher. «credit» ist nicht zulässig.

## 3 Zusatzdienste

TrustX unterstützt Dienste im Bereich eDA und Rechnungsdruck. In diesem Kapitel werden die Dienste und deren Steuerung aus technischer Sicht erläutert.

### 3.1 Abrechnungsarten

#### 3.1.1 Tiers Garant – Token auf Rückforderungsbeleg

Im Tiers Garant geht die Rechnung an den Patienten, welcher den beiliegenden Rückforderungsbeleg beim Versicherer einreichen kann. Damit der Versicherer eine elektronische Rechnungskopie abholen kann, braucht es auf dem Rückforderungsbeleg einen eindeutigen Identifikator, den sogenannten «Token». Der Token wird auf den Rückforderungsbeleg aufgedruckt (siehe XML-Standard, Forum für Datenaustausch).

#### 3.1.2 Tiers Payant

Im Tiers Payant (und Tiers Soldant) geht die Rechnung an den Versicherer, der Patient erhält eine Rechnungskopie (Patientenkopie, nur im Tier Payant). TrustX nimmt beim elektronischen Austausch der Rechnung und Rechnungsantwort zwischen Praxis und Versicherer die Rolle als Intermediär ein.

#### 3.1.3 Übersicht Anforderungen und Abhängigkeiten

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

Item	Abrechnungsart		
	Tiers Garant	Tiers Payant	Tiers Soldant
<b>Rechnung</b>	geht an Patient Format: frei	geht an Versicherer Format gemäss Vorgabe Forum Datenaustausch	geht an Versicherer Format gemäss Vorgabe Forum Datenaustausch
<b>Rückforderungsbeleg</b>	geht an Patient Format gemäss Vorgabe Forum Datenaustausch	---	---
<b>Rechnungskopie</b>	---	geht an Patient Format gemäss Vorgabe Forum Datenaustausch	---
<b>Rechnungsantwort</b>	----	geht an Praxis	geht an Praxis
<b>Mahnung</b>	geht an Patient	geht an Versicherer	geht an Versicherer
<b>Token</b>	auf Rückforderungsbeleg	---	---
<b>Gesetz</b>	KVG, VVG, (UVG / MVG / IVG)	KVG, (VVG)	UVG / MVG / IVG

Patient: Patient bzw. Garant

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



## 3.2 Zusatzdienste eDA und Druck&Versand

### 3.2.1 Übersicht

Zusatzdienst	Beschreibung
<b>eDA TG</b>	elektronische Bereitstellung von TG-Rechnungen an den Versicherer.
<b>eDA TP</b>	Weiterleitung von TP-/TS-Rechnungen (Rechnung, Storno, Mahnung) an den Versicherer. Druck & Versand an den Versicherer, falls dieser den eDA nicht unterstützt (eDA TP Druck&Versand). Weiterleitung von TP-Rechnungsantworten an die Praxis.
<b>D&amp;V TG</b>	Druck&Versand von TG-Rechnungen (Rechnung, Mahnung) an den Patienten.
<b>D&amp;V TP</b>	Druck&Versand von TP-Rechnungskopien an den Patienten (Patientenkopie). Der Druck&Versand TP ist nur in Kombination mit dem eDA TP möglich.

Eine Weiterleitung an den Versicherer und/oder ein Druck&Versand erfolgt nur, wenn die Praxis den entsprechenden Dienst beim TrustCenter aktiviert hat.

Unabhängig von den Zusatzdiensten und entsprechenden Steuerungsattributen in der XML-Rechnung fließen alle Rechnungen der angeschlossenen Arztpraxen in die Ärzteneigene Datensammlung ein.

### 3.3 Steuerung eDA und Druck&Versand bis XML 4.1

Die Steuerung der Zusatzdienste erfolgt bis und mit XML-Standard 4.1 über das Attribut «resend» in der XML-Rechnung (XML-Attribute *resend*, ID 10170):

- Originalrechnung : *resend* = false
- Kopie der Rechnung: *resend* = true

Rechnung	von der Praxis abonnierter Zusatzdienst			
	eDA TG	D&V TG	eDA TP / TS	D&V TP
<b>Original</b> (resend = false)	-- (wird nicht beachtet)	Druck & Versand der Rechnung / Mahnung an den Patienten durch TrustX	Weiterleitung der Rechnung / Mahnung an Versicherer	Nur wenn auch Dienst eDA TP abonniert: Druck & Versand der Rechnungskopie an den Patienten durch TrustX
<b>Kopie</b> (resend = true)	-- (wird nicht beachtet)	-- (kein Druck durch TrustX)	-- (keine Weiterleitung durch TrustX)	-- (kein Druck durch TrustX)
<b>Token</b>	<b>Token drucken</b> (Rückforderungsbeleg; durch Druckstelle) 1)		--	--

#### Verarbeitung

- Es werden nur Original-Rechnungen und -Mahnungen (XML-Attribute *resend* = false) für den eDA und Druck&Versand verarbeitet.

- Mahnungen werden nur an den Versicherer weitergeleitet, wenn die dazugehörige Rechnung in TrustX vorhanden ist.
- Wird eine Rechnung / Mahnung zweimal als Original angeliefert, so wird diese auch zweimal an den Versicherer weitergeleitet bzw. gedruckt und versendet.

Falls ein Versicherer die elektronische Zustellung nicht unterstützt, werden die Rechnungen im eDA TP / TS gedruckt und dem Versicherer per Post zugestellt.

### 3.3.1 Hinweise an Intermediäre

In XML 4.0 Rechnungen muss das Attribut `resend = true` gesetzt werden, wenn keine Verarbeitung im eDA und/oder Druck&Versand durch TrustX erfolgen soll.

## 3.4 Steuerung eDA und Druck&Versand ab XML 4.3

Die Steuerung der Zusatzdienste erfolgt ab XML 4.3 über die folgenden XML-Attribute. Eine schematische Darstellung findet sich im Anhang, Kap. 9.

### TG – Tiers Garant Rechnung

XML Attribut	von Praxis abonniertes Dienst	
	D&V TG	--
<code>print_at_intermediate = TRUE</code>	Druck & Versand der Rechnung mit Einzahlungsschein + Rückforderungsbeleg und Kopie an den Patienten	--

### TP – Tiers Payant Rechnung

XML Attribut	von Praxis abonniertes Dienst	
	eDA TP/TS	D&V TP
<code>print_at_intermediate = TRUE 1)</code>	elektronische Übermittlung der Rechnung an den Versicherer	--
<code>print_patient_copy / print_copy_to_guarantor = TRUE 2)</code>	--	Nur wenn auch Dienst eDA TP abonniert und <code>print_at_intermediate = TRUE</code> (Weiterleitung an Versicherer KVG): Druck und Versand einer Rechnungskopie ohne Einzahlungsschein an den Patienten.

- 1) Das Attribut «`print_at_intermediate`» wird nach Rücksprache mit den grossen Praxissoftware-Anbietern auch zur Steuerung des eDA TP (Weiterleitung von TP-Rechnungen an die Versicherer) verwendet.
- 2) XML Standard 4.3, 4.4: `print_patient_copy`, XML Standard 4.5: `print_copy_to_guarantor`

#### Verarbeitung

- Mahnungen werden nur an den Versicherer weitergeleitet, wenn die dazugehörige Rechnung in TrustX vorhanden ist.

- Wird eine Rechnung / Mahnung zweimal als Original angeliefert, so wird diese auch zweimal an den Versicherer weitergeleitet bzw. gedruckt und versendet.
- Falls ein Versicherer die elektronische Zustellung nicht unterstützt, werden die Rechnungen im eDA TP / TS gedruckt und dem Versicherer per Post zugestellt.

### 3.4.1 Hinweise an Intermediäre

- In TP-Rechnungen, welche von TrustX an einen Intermediär gesendet werden zwecks Weiterleitung an die Versicherer, wird das letzte via-Element auf die EAN des Intermediärs gesetzt.

## 3.5 eDA TG

Damit der elektronische Datenaustausch von TG-Rechnungen mit den Versicherern funktioniert, muss die Praxis

- den Zusatzdienst eDA TG aktiviert haben
- die Druckstelle (Praxis oder beauftragter Dritter) das Token auf den Rückforderungsbeleg drucken

Das Token soll nur aufgedruckt werden, wenn die Praxis einem TrustCenter oder der Zentrale Datensammelstelle angeschlossen ist und damit Rechnungen an TrustX anliedert sowie den Dienst eDA TG aktiviert hat. Falsch bzw. fälschlicherweise aufgedruckte Token auf dem Rückforderungsbeleg können zu erheblichen Zusatzaufwand beim Versicherer führen.

- die Praxis die Rechnung unmittelbar nach der Fakturierung an TrustX anliefern

---

## 4 Anforderungen an die Praxissoftware

---

### 4.1 Übersicht

Der TX Praxis Client kann mit eigener Benutzeroberfläche betrieben werden (Kap. 5, «Funktionale Beschreibung von TX Praxis») oder über eine Schnittstelle direkt aus der Praxissoftware angesteuert werden (Kap. 6, «Spezifikation TX Praxis») und so vollständig in die Praxissoftware integriert werden (Kap. 4.2, «Möglichkeiten zur Integration des TX Praxis»).

Für die Übermittlung sind Rechnungen, Storni sowie Mahnungen elektronisch in den unterstützten XML-Formaten im entsprechenden TX Praxis Verzeichnis bereitzustellen. Es sind keine Modifikationen an den Rechnungen / Storni / Mahnungen notwendig, damit diese von TX Praxis verarbeitet werden können. Prüfung, Verschlüsselung / Anonymisierung und Übermittlung an TrustX wird durch TX Praxis vorgenommen.

Folgendes ist besonders zu beachten:

- Beim Druck von Rechnungen soll das Token nur auf den Rückforderungsbeleg (TG) gedruckt, falls die Arztpraxis einem TrustCenter oder der Zentralen Datensammelstelle angeschlossen ist.  
Das Token muss für jede Rechnung (auch für korrigierte Rechnungen) eindeutig sein.
- Auf die korrekte Codierung von Storni gemäss XML-Standard ist zu achten.

### 4.2 Möglichkeiten zur Integration des TX Praxis Client

Der TX Praxis Client kann verschieden tief in die Praxissoftware integriert werden.

#### 4.2.1 Keine Integration

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

#### 4.2.2 Aufruf des TX Praxis Client aus der Praxissoftware

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

Technisch gesehen gibt es zwei Möglichkeiten für eine solche Integration:

- Kommandozeilenaufruf des TX Praxis Client
- Konfiguration via Schnittstelle (siehe Kap. 6, «Spezifikation TX Praxis»). Dies erlaubt es zusätzliche Konfigurationsparameter – z.B. das Rechnungsverzeichnis – vorab im TX Praxis Client zu setzen. Anschliessend erfolgt der Kommandozeilenaufruf des TX Praxis Client.

#### 4.2.3 Vollständige Integration

Der TX Praxis Client kann auch vollständig durch die Praxissoftware gesteuert werden:

- die Oberfläche vom TX Praxis Client bleibt unsichtbar

- die Meldungen vom TX Praxis Client können von der Praxissoftware direkt empfangen und verarbeitet werden

Vorteil der vollständigen Integration ist, dass auf (Fehler-)Meldungen vom TX Praxis Client direkt aus der Praxissoftware reagiert werden kann. Die Praxissoftware kann zudem die Funktionalitäten vom TX Praxis Client unter der eigenen Oberfläche anbieten.

Die vollständige Integration bietet dem Benutzer einen höheren Komfort, bedeutet aber auch einen etwas höheren Entwicklungsaufwand seitens des Praxissoftware-Anbieters. Die korrekte Ansteuerung und die Verarbeitung von TrustX-Meldungen sowie die Aktualisierung der Praxissoftware liegen bei in der Verantwortung des Praxissoftware-Anbieters.

### 4.3 Doppelte Rechnungen / Mahnungen

Die Erkennung von doppelten XML-Rechnungen bzw. -Mahnungen findet in TrustX an zwei Orten statt. Der TX Praxis Client erkennt aufgrund unterschiedlicher Kriterien Dubletten noch vor der Anlieferung an TrustX. Nach erfolgter Anlieferung wird vor der statistischen Auswertung nochmals ausführlich auf Doubletten geprüft.

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

XML 4.0		ab XML 4.3	
Rechnung	Mahnung	Rechnung	Mahnung
Biller_EAN	Biller_EAN	<del>Biller_EAN</del> (entfällt)	<del>Biller_EAN</del> (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)

#### 4.3.1 Erkennung doppelter Rechnungen / Mahnungen in TX Praxis

Für die Erkennung von doppelten Rechnungen / Mahnungen werden bei der Übermittlung der Rechnungen bzw. Mahnungen die nachfolgenden Informationen lokal im trustx.log abgelegt.

In Erweiterung zum XML-Standard werden zusätzlich die Attribute Provider\_ZSR, Amount und TP Purpose gemäss folgender Tabelle für die sichere Erkennung geprüft.

Rechnungsinformation	Mahnungsinformation	Bemerkung
Biller_EAN (XML 4.0) bzw. Invoice_Date (ab XML 4.3)	Biller_EAN (XML 4.0) bzw. Reminder_Date (ab XML 4.3)	--
Invoice_ID	Reminder_ID	--
Invoice_TimeStamp	Reminder_TimeStamp	--
Provider ZSR	--	Ist zusammen mit der Angabe des TrustCenters (Mandant) die eindeutige Identifikation des Kunden (Praxis) innerhalb des TrustX Systems.

Amount	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).

Stimmen die Werte der aufgeführten Attribute überein, wird die Rechnung bzw. die Mahnung als Doublette erkannt. Weicht mind. einer der Werte ab, wird die Rechnung / Mahnung als neue Rechnung bzw. Mahnung an TrustX übermittelt.

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

### 4.3.2 Erkennung doppelter Rechnungen vor der statistischen Auswertung

Im täglichen Betrieb tritt immer wieder mal der Fall auf, dass Rechnungen versehentlich oder willentlich doch mehrmals angeliefert werden (Löschen des Logfiles, doppelte Anlieferung über einen zweiten PC, usw.). Solche doppelten Rechnungen würden die Auswertungen und Statistiken verfälschen.

Vor der statistischen Auswertung findet deshalb nochmals eine Prüfung auf Doubletten statt. Dabei kommen erweiterte Prüfungen zur Anwendung, um auch in 'verdeckte' Doubletten erkennen zu können, z.B. die ein und dieselbe Rechnung wird mehrfach, jedoch mit unterschiedlichem Timestamp generiert und übermittelt.

Da Mahnungen nicht in die statistischen Auswertungen einfließen, erübrigt sich hier ein weiterer Doublettencheck.

→ Mehrfach angelieferte Rechnungen und Mahnungen allgemein werden nur für die statistischen Auswertungen eliminiert. Bei den Zusatzdiensten werden solche Rechnungen / Mahnungen mehrfach verarbeitet (gedruckt und/oder versendet).

### 4.3.3 Aufbau von trustx.log (TX Praxis Client)

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

Invoice_ID / Reminder_ID	Amount	TimeStamp	ZSR Provider	Biller EAN	Invoice_ Reminder_ TimeStamp	Pur- pose	
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

## 4.4 Storno

### 4.4.1 Storno im XML 3.0 / 4.0

Gemäss XML-Rechnungsstandard 3.0 und 4.0 sind Storni mit gleicher ESR-Nummer und allen mit -1 multiplizierten Beträgen zu übermitteln. Da die Attribute «Anzahlung» (10335) und «Gerundeter Nettorechnungsbetrag» (10340) grösser oder gleich 0 sein müssen, werden diese auf 0 gesetzt (anstatt Minus-Betrag). Alle anderen Attribute bleiben identisch wie in der zu stornierenden Rechnung.

### 4.4.2 Storno im XML 4.0 – TP-Rechnungen

Im XML-Rechnungsstandard 4.0 wird eine TP-Rechnung mit dem Attribute purpose = «annulment» als Storno deklariert (XML 4.0: request / invoice / tiers\_payant / purpose; ID 11265).

### 4.4.3 Storno ab XML 4.3

Ab XML-Rechnungsstandard 4.3 wird ein Storno mit dem Attribut storno = «true» gekennzeichnet. Zusätzlich müssen

- der Werte des Attributes quantity bei jeder Leistungsposition mit -1 multipliziert werden
- die Werte bei den Attributen amount\_prepaid und amount\_reminder mit -1 multipliziert werden.

Ausführliche Informationen finden sich im XML-Rechnungsstandard 4.3/4.4/4.5.

## 4.5 Eindeutiges Token im eDA TG

Basis für das Token auf dem Rückforderungsbeleg 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 und sich die einzelnen ESR-Teilnehmer grundsätzlich eindeutig identifizieren lassen.

Voraussetzung ist die Vergabe von eindeutigen ESR-Nummern. Deren Vergabe liegt in der Verantwortung der Praxis. Damit der elektronische Datenaustausch einwandfrei funktioniert, müssen auch korrigierte Rechnungen neue ESR-Nummern erhalten. Dies wird beispielsweise 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 finden sich in Kap. 7, «Anhang – Codierung Token».

## 4.6 TrustCenter «TC Test»

Zu Testzwecken steht das TrustCenter «TC Test» zur Verfügung.

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

Bitte führen Sie keine Tests unter produktiven TrustCentern durch, da die Testrechnungen sonst in die Statistiken und die Verarbeitung der Zusatzdienste eingehen.

Für Testlieferungen an das «TC Test» wird ein gültiges HIN-Login. Eine Anmeldung beim «TC Test» ist nicht notwendig.

Unter dem «TC Test» steht der TrustX-Data-Monitor 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.

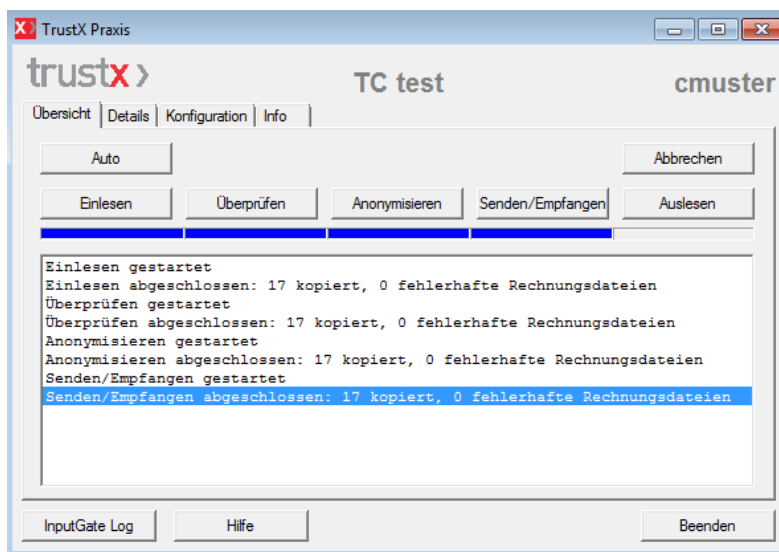
Die URL für den Aufruf des Data-Monitor lautet: <http://iglog-test.trustx.hin.ch/cgi-bin/sitemgr.pl>

## 5 Funktionale Beschreibung von TX Praxis Client

Der TX Praxis Client ist eine Software-Komponente mit eigener Benutzeroberfläche und einer Schnittstelle (API) zur direkten Ansteuerung aus dem Praxissoftwaresystem oder anderen SW-Lösungen.

### 5.1 Benutzeroberfläche

Die Benutzeroberfläche von TX 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 die verschiedenen Verzeichnisse und Log-Dateien
- <Konfiguration> Erlaubt die Konfiguration von TX Praxis
- <Info> Gibt über TX Praxis Auskunft (z.B. aktuelle Version)

Unten links befindet sich ein Button, mit welchem der TrustX InputGate Log Viewer aufgerufen werden kann. Im InputGate Log Viewer können die Rechnungsanlieferungen online eingesehen werden.

#### 5.1.1 Register <Übersicht>

Das Register <Übersicht> umfasst alle Funktionen zur Verarbeitung und Übermittlung der Rechnungen an TrustX. Die Verarbeitung und Übermittlung steht in zwei Varianten zur Verfügung:

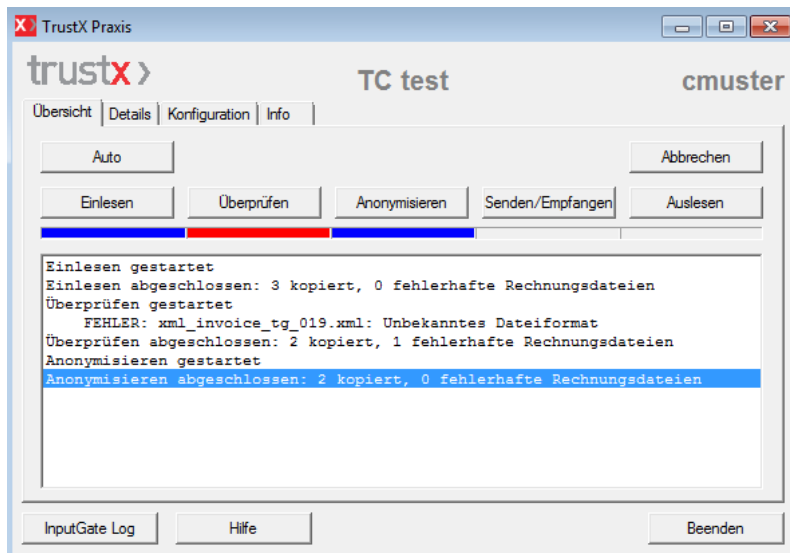
##### Variante «Auto»

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.



## 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.



Im weissen Fenster wird eine Zusammenfassung der Verarbeitung der Daten angezeigt.

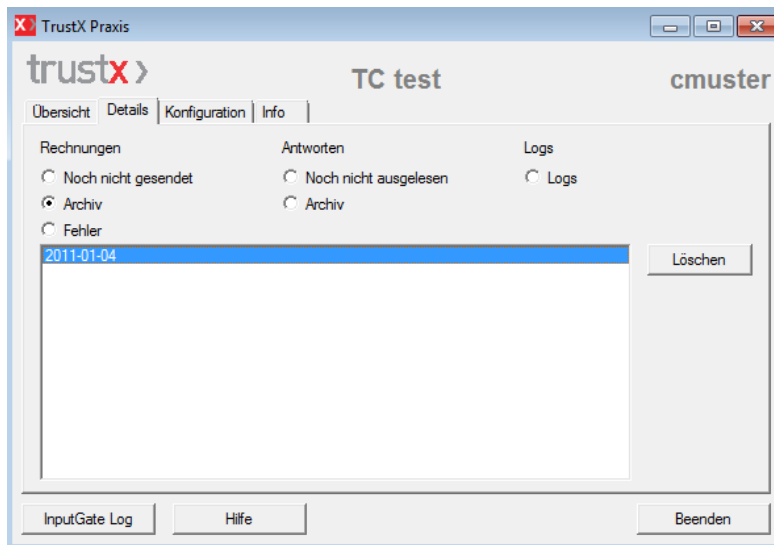
Allfällige Fehler werden beim entsprechenden Verarbeitungsschritt pro Rechnung angezeigt.

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

Siehe auch Kap. 5.1.2 «Register <Details>».

- <Einlesen>                    Daten werden aus dem Rechnungsverzeichnis (im Register <Konfiguration> definiert) eingelesen und dort anschliessend gelöscht.
- <Überprüfen>                XML-Schemavalidierung.
- <Anonymisieren>            Patientenangaben werden verschlüsselt.
- <Senden/Empfangen>        Daten werden an TrustX gesendet (Rechnungen) oder von TrustX empfangen (Rechnungsantworten). Die Internetverbindung wird – falls notwendig – durch TX Praxis gemäss den Einstellungen im Internet Explorer aufgebaut und wieder geschlossen.
- <Auslesen>                    Rechnungsantworten werden ins Antwortverzeichnis (im Register <Konfiguration> definiert) verschoben.

## 5.1.2 Register <Details>



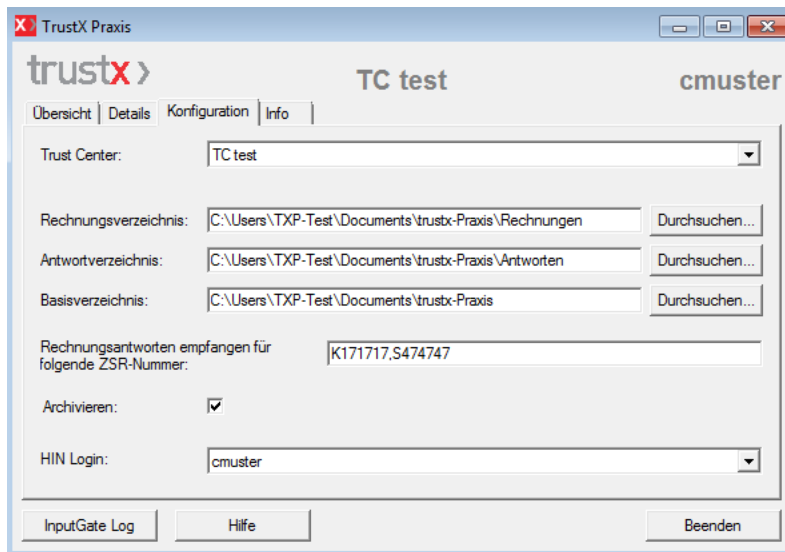
Das Register <Details> stellt verschiedene Detailansichten zur Verfügung.

Über die Auswahlfelder (oben) können die gewünschten Informationen abgefragt 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.       |
| <Noch nicht ausgelesen> | Die bereitstehenden (noch nicht bearbeiteten) Rechnungsantworten, die sich im «Antwortverzeichnis» befinden, werden aufgelistet. |
| <Archiv>                | Die erfolgreich gesendeten Rechnungen / ausgelesenen Rechnungsantworten 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.   |

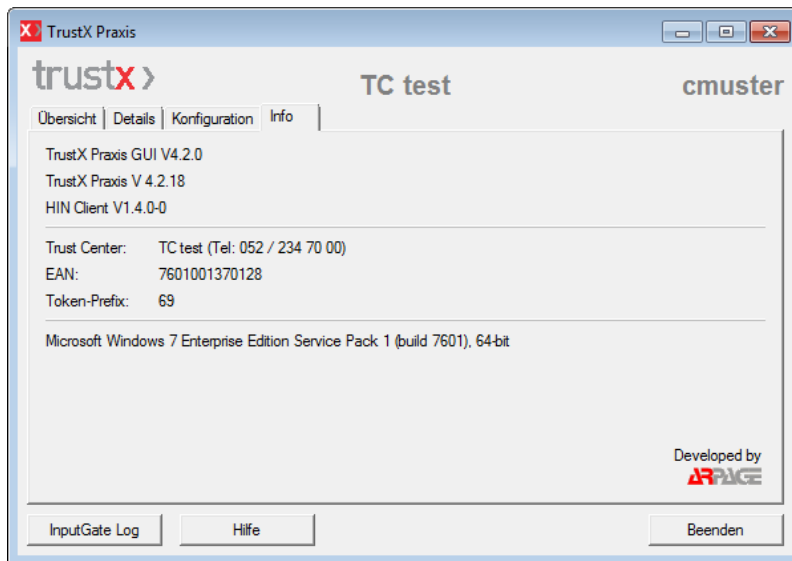
### 5.1.3 Register <Konfiguration>



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

- <TrustCenter> Wählen Sie Ihr TrustCenter (TrustCenter, welchem die Arztpraxis angeschlossen ist) aus der Liste aus. Für Tests verwenden Sie das «TC Test».
- <Rechnungsverzeichnis> Verzeichnis, in welchem die Rechnungen bereitgestellt werden. Beim Schritt «Einlesen» werden die Rechnungen in diesem Verzeichnis gelesen und ins Verzeichnis «input» verschoben. Das Verzeichnis «input» ist von TX Praxis fix vorgegeben und ein Unterverzeichnis des TrustX Basisverzeichnisses.
- <Antwortverzeichnis> Verzeichnis, in welchem die Rechnungsantworten bereitgestellt werden sollen. Beim Schritt «Auslesen» werden Rechnungsantworten aus dem TrustX Verzeichnis «responses» gelesen und ins <Antwortverzeichnis> verschoben. Das Verzeichnis «responses» ist von TX Praxis fix vorgegeben und ein Unterverzeichnis des TrustX Basisverzeichnisses.
- <Basisverzeichnis> Verzeichnis, welches für den Betrieb TrustX notwendig ist. Siehe Kap. 6.5, «Verzeichnisstruktur».
- <Rechnungsantworten empfangen für ...> Tragen Sie hier Ihre ZSR-Nummer(n) ein, für welche Sie Rechnungsantworten empfangen möchten
  - kein Eintrag es werden keine Rechnungsantworten gesucht
  - "\*" es werden alle Rechnungsantworten für das gesetzte HIN-Login gesucht
  - ZSR1,ZSR2.. ein oder mehrere, durch ein Komma getrennte ZSR-Nummern: es werden alle Rechnungsantworten für die angegebenen ZSR-Nummern und das gesetzte HIN-Login gesucht
- <Archivieren > Über die Option «Archivieren» kann gesteuert werden, ob erfolgreich versendete Rechnungsdateien archiviert oder gelöscht werden sollen. Archivierte Rechnungen liegen in der Form vor, in der sie versendet wurden, d.h. verschlüsselt / anonymisiert.
- <HIN Login> In diesem Feld wird definiert, mit welchem HIN-Login TX Praxis für die Verschlüsselung und den Datentransfer arbeitet.

### 5.1.4 Register <Info>



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

## 5.2 Mandanten in TX Praxis

TX Praxis ist mandantenfähig – alle Daten (Rechnungen, Rechnungsantworten, Logfiles, Einstellungen) werden im jeweiligen Benutzerprofil abgelegt. Dies ist insbesondere dann von Vorteil, wenn TX 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.

## 5.3 Auto-Update-Funktion

TX Praxis beinhaltet eine Auto-Update-Funktion über das Internet. Das Update kann auch direkt über die Schnittstelle gesteuert werden. Die neue Version wird im Anschluss an die Rechnungsübermittlung an TrustX auf den PC des Benutzers geladen, falls dieser einverstanden ist. TX Praxis kann so mit minimalem Aufwand auf dem neusten Versionsstand gehalten werden.

## 6 Spezifikation TX Praxis Client – Schnittstelle

Der TX Praxis Client kann von der Praxissoftware (allgemein von Anwendungen) über eine Schnittstelle angesteuert werden. Diese ist im «OMG IDL» definiert. Aus dieser Definition werden die Schnittstellendefinitionen für die entsprechenden Betriebssysteme und Technologien generiert. Dies sind «COM» für Windows und «Apple Events» für Mac OS X. TX Praxis generiert detaillierte Meldungen, die an die aufrufende Anwendung zurückgesendet werden.

### 6.1 Schnittstellendefinition

Hinweis: Zum besseren Verständnis siehe auch Kap. 5, «Funktionale Beschreibung von TX 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&lt;string&gt; strings;</code>	
<code>    readonly attribute string trustxVersion</code>	Version von TX 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 TX Praxis vorgeben
<code>    attribute boolean activateGUI;</code>	Zurzeit nicht implementiert, siehe Kap. 4.2.2 "Aufruf des TX Praxis Client aus der Praxissoftware".
<code>    attribute boolean debug;</code>	Debug-Meldungen generieren ja/nein
<code>void Auto();</code>	Automatisches Verarbeiten und Senden der Rechnungs-

Schnittstellendefinition	Bemerkung
	dateien analog dem Button <Auto>
void Read();	Einlesen der Daten analog dem Button <Einlesen>, Rechnungsdateien werden von ..\{Rechnungsverzeichnis}\.. nach ..\{basisverzeichnis}\input verschoben
void Check();	Daten werden auf formale Korrektheit geprüft (XML-Schemavalidierung). 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 an TrustX @ TrustCenter (Mandant) gesendet.
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-Nummern. Es werden alle Antworten für die angegebenen ZSR-Nummern und das eingestellte HIN-Login gesucht.</p>
// Added in V3	
readonly attribute string getUpdateMsg;	Gibt die Update-Meldung aus, welche auf dem Server hinterlegt ist.
// Added in V4	
// Added in V4	
void OpenSession()	Öffnet eine Session zur effizienten Verarbeitung von

Schnittstellendefinition	Bemerkung
	mehreren Rechnungen mittels 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>	

## 6.2 Events

Jedes Modul in TX 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:

### 6.2.1 Klassifizierung der Events

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

Klassifizierung	Bemerkung
enum MsgOrigin {	<b>Message Origin</b>
moPReader = 0,	Meldung kommt vom TX Praxis Modul - Reader
moPChecker = 1,	Meldung kommt vom TX Praxis Modul - Checker
moPAnonymizer = 2,	Meldung kommt vom TX Praxis Modul - Anonymizer
moPEncrypter = 3,	Meldung kommt vom TX Praxis Modul - Encrypter
moPSender = 4,	Meldung kommt vom TX 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 TX Praxis Modul - Downloader
moPWriter = 8,	Meldung kommt vom Auslese-Modul
};	

## 6.2.2 Schnittstelle zum Empfangen der Events

Um Event-Meldungen von TX 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 trustxEvt(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 die Datei an, auf welche sich die Meldung bezieht. Kann leer sein, wenn sich die Meldung auf das 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: keine Verbindung. Eine entsprechende Fehlermeldung wird mit MessageClass=mcASAS mitgeteilt.

## 6.2.3 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 {	<b>Messages (Code)</b>
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: HIN-Login nicht gefunden
ERR_CONF_TRUSTCENTER = 9,	Konfiguration: TrustCenter nicht gefunden
ERR_UNKNOWN_FILE_FORMAT = 10,	Unbekanntes Fileformat (Checker von TX 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 HIN-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)
ERR_ENC_SEND_TC_MISMATCH = 22,	Rechnung wurde bereits für ein anderes TrustCenter verschlüsselt
ERR_CONF_OUTPUTDIR = 23,	Konfiguration: Antwortverzeichnis nicht gefunden
};	

### 6.3 TX Praxis Client einbinden

Sämtliche Prozesse (Module) wie Auto, Einlesen, usw. arbeiten asynchron. Dies bedeutet, dass die aufrufende Anwendung sofort vom entsprechenden TrustX-Modul einen Event mit Code INFO\_MODULE\_START erhält. Damit wird der Start des Moduls gekennzeichnet.

Es gilt immer folgender formaler Ablauf:

- Erster Event, welchen den Start des Moduls bestätigt (Code = INFO\_MODULE\_START).
- Anschliessend folgt genau ein Event der Klasse «Count». Der Code enthält die Anzahl der zu verarbeitenden Dateien.
- Pro Datei 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. 8, «Beispiel TrustX-Event (VB6)»).

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

## 6.4 Dateibenennung

Rechnungsdateien können beliebige Dateinamen haben.

## 6.5 Verzeichnisstruktur

In TX Praxis kann ein Basisverzeichnis {bz} vorgegeben werden. Darunter werden die folgenden «internen», für den Betrieb von TX Praxis notwendigen Verzeichnisse angelegt:

..\{bz}\input	Im Schritt <Einlesen> werden die zu verarbeitenden Rechnungsfiles aus dem eingestellten Rechnungsverzeichnis in dieses Verzeichnis verschoben. In diesem Verzeichnis beginnt der eigentliche Verarbeitungsprozess von TX 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 an TrustX versendeten Rechnungen abgespeichert, sofern die Option Archivieren im Register <Konfiguration> ausgewählt wurde.
..\{bz}\responses-archive	In diesem Verzeichnis werden alle empfangenen Rechnungsantworten abgespeichert, sofern die Option Archivieren im Register <Konfiguration> ausgewählt wurde.
..\{bz}\logs	Enthält die Logfiles.
..\{bz}\err	In dieses Verzeichnis gelangen alle durch check() oder anonymize() verarbeiteten Files, die einen Fehler beinhalten.

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

## 6.6 Wichtige Dateien von TX Praxis Client

### 6.6.1 Programmfiles

trustxui.exe	TX Praxis User Interface (Windows)
trustx.exe	TX Praxis COM Server. Enthält alle Funktionen und die Schnittstelle (API)
msvcpr140.dll	Microsoft Visual C/C++ Runtime Libraries
vcruntime140.dll	
vcruntime140_1.dll	
<andere>.dll	TX Praxis Support-Libraries

### 6.6.2 Konfigurations- und Logfiles

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

Filename	Inhalt	pro User	Pfad
trustx.log	Rechnungs- informationen, welche für den Doubletten- check benötigt werden.	Ja	Im Root vom eingestellten Basisverzeichnis, i.d.R.  Windows: C:\Users\<<user>\Documents\trustx-Praxis  Mac: /Users/MY_USER/Documents/trustx-Praxis
YYYY-MM-DD.log	Detaillierte Log- informationen zu jedem Verarbeitungs- schritt	Ja	Im Verzeichnis "logs" unter dem eingestellten Basisverzeichnis, z.B.  Windows: C:\Users\<<user>\Documents\trustx- Praxis\logs  Mac: /Users/MY_USER/Documents/trustx- Praxis/logs
trustx.ini (Win) ch.trustx.praxis.fmwk.plist (Mac)	Konfigurations- einstellungen	Ja	Windows: C:\Users\<<user>\WorkFolders\AppData\Roaming\Arpage\trustx-praxis  Mac: /Users/MY_USER/Library/Preferences/ch.trustx.praxis.fmwk.plist
trustx.xml (2x) trustx.xsd (2x)	Default-Konfiguration  Wird bei der Installation angelegt	Nein	Windows: C:\Program Files (x86)\Arpage\TrustX Praxis  Mac: /Applications/trustx- Praxis.app/Contents/SharedFrameworks/tr ustx.frameworks/Versions/A/Resources/tr ustx.xml

Filename	Inhalt	pro User	Pfad
	Wird im User-Verzeichnis bei jedem Start von TX Praxis (Schritt Einlesen) aktualisiert	Ja	Windows: C:\Users\ <user&gt;\workfolders\appdata\roaming\arpage\trustx-praxis </user&gt;\workfolders\appdata\roaming\arpage\trustx-praxis  Mac: /Users/MY_USER/Library/Application Support/trustx-Praxis/trustx.xml

## 7 Anhang – Codierung Token

Als Token (Rechnungs-Identifikator auf dem Rückforderungsbeleg) wird eine modifizierte ESR-Zeile verwendet. ESR-Zeile: esr9-Verfahren mit 27-stelliger ESR-Ref.-Nr. und Betrag.

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

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 für den Patienten 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.

### Codierung ESR / Token:

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

mit folgenden Codierung des TrustCenters (A)

TrustCenter	Code (A)	TrustCenter	Code (A)
Ctésias	51	Züridoc	60
GallOnet (eastcare)	52	Trustmed	61
Hawatrust	53	GTS (General Transport Service)	62
Medkey	54	Physiotrust	63
PonteNova	55	DS Liechtenstein	64
Syndata	56	DS Chiropraktoren	65
TC Aargau	57	TC Test	69
thurcare (eastcare)	58		
TC Ticino	59		

## 8 Anhang – Beispiel TrustX-Event (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"
            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

```

## 9 Anhang – Schema Steuerung eDA und Druck&Versand ab XML 4.3

