

TrustX Praxis Version 3.0

Schnittstellenbeschreibung für Praxissoftware-Hersteller

Dokumentversion 1.0
Januar 2011



Ist aus der Zusammenarbeit
von NewIndex, Ärztekasse
und BlueCare entstanden

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

Änderungen

Dokument-Version	Änderung	Datum
1.0	Schnittstellenbeschreibung von TrustX Praxis Version 3	04.01.2011

Inhaltsverzeichnis

1	<i>Einleitung</i>	4
1.1	Hintergrundinformation	4
1.2	TrustCenter	4
1.3	HIN Security	4
1.4	Ansprechpartner	4
2	<i>TrustX in der Arztpraxis</i>	5
2.1	Übersicht über Gesamtsystem	5
2.2	TrustX Praxis	7
2.3	XML-Standards	8
3	<i>Zusatzprodukte der TrustCenter</i>	9
3.1	Abrechnungsarten.....	9
3.2	Zusatzprodukte eDA und Rechnungsdruck	11
3.3	Steuerung der Zusatzprodukte	12
4	<i>Anforderungen an die Praxissoftware</i>	14
4.1	Übersicht.....	14
4.2	Elektronische Rechnungen und Mahnungen bereitstellen	14
4.3	Möglichkeiten zur Integration von TrustX Praxis	14
4.4	Eindeutiges Token (Dokumenten-Identifizier) für eDA TG	15
4.5	Hinweise zur Steuerung von Zusatzprodukten	15
4.6	Doppelte Rechnungen / doppelte Mahnungen	16
4.7	Storno	19
4.8	XML 4.0.....	20
4.9	Hinweise zu PLZ in XML-Rechnungen	20
4.10	Neue AHV-Nummer	20
4.11	Datensammlung für Praxisspiegel und eDA über einen Intermediär	20
4.12	Tests und Zertifizierung	21
5	<i>Funktionale Beschreibung von TrustX Praxis</i>	22
5.1	Register <Übersicht>	23
5.2	Register <Details>	24
5.3	Register <Konfiguration>	25
5.4	Register <Info>	26
5.5	Mandanten in TrustX Praxis	26
6	<i>Schnittstelle zu TrustX Praxis</i>	27
6.1	Einleitung	27
6.2	Schnittstellendefinition	27
6.3	Events	29
6.4	TrustX einbinden.....	31
6.5	Dateibenennung.....	31
6.6	Verzeichnisstruktur	31
6.7	Wichtige Dateien von TrustX Praxis	33
7	<i>Anhänge</i>	34
	Anhang 1: ESR-Zeile als Dokumenten-Identifizier (Token).....	34
	Anhang 2: TrustX-Event Beispiel (VB6).....	36

1 Einleitung

1.1 Hintergrundinformation

Die von NewIndex initiierten 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 NewIndex www.newindex.ch oder bei den einzelnen TrustCentern. Eine Liste aller mit *TrustX* arbeitenden TrustCenter finden Sie auch unter www.trustx.ch.

Die TrustCenter bieten zugleich 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 Security

TrustX verwendet die Security von HIN¹. Weitere Informationen finden Sie unter www.hin.ch.

1.4 Ansprechpartner

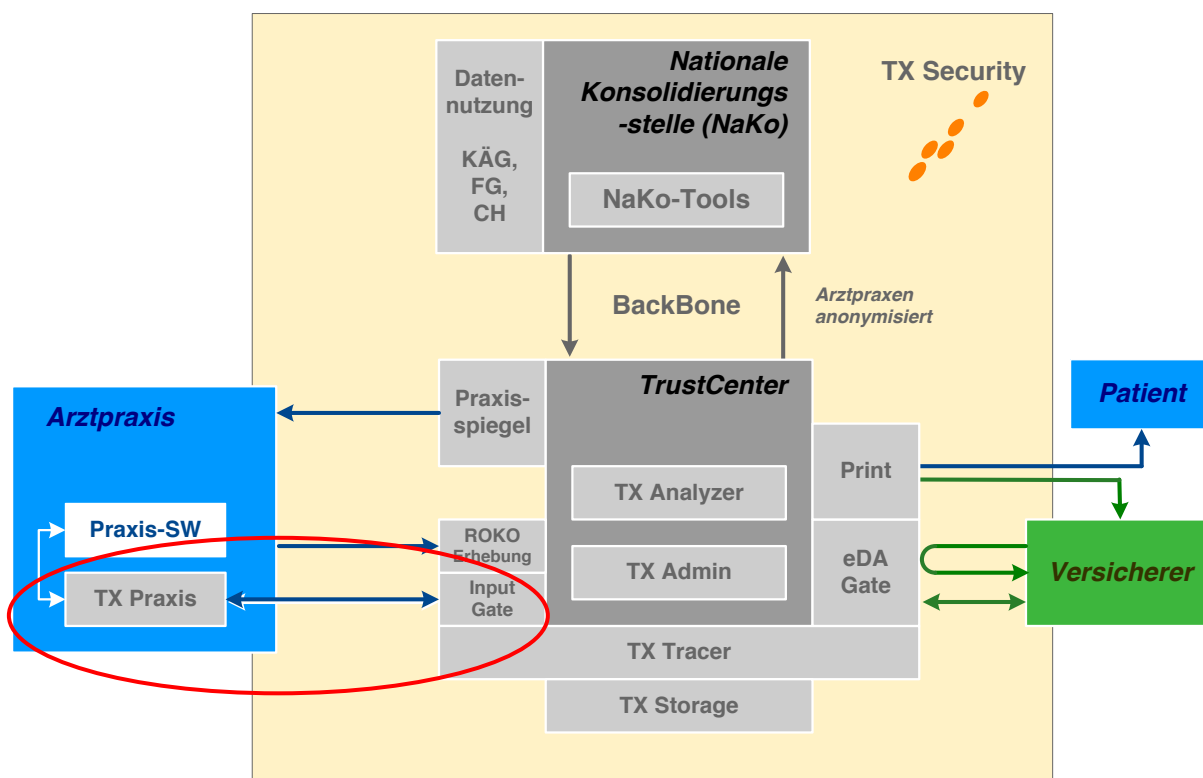
Ansprechpartner für technische Fragen und Zusatzprodukte ist unser Support. Sie erreichen diesen zu den üblichen Geschäftszeiten unter 052 234 70 00 oder via E-Mail unter support@trustx.ch.

¹ Health Info Net

2 TrustX in der Arztpraxis

2.1 Übersicht über Gesamtsystem

Das IT-System *TrustX* umfasst eine Vielzahl von Modulen und Funktionalitäten. Die nachstehende Übersicht zeigt den logischen Aufbau des Systems schematisch auf.



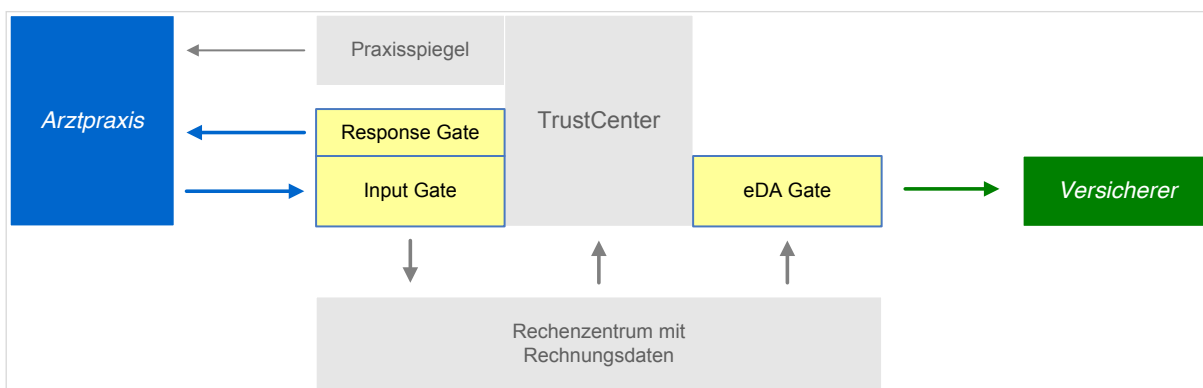
Im Fokus dieses Dokuments stehen die Schnittstelle zwischen der Arztpraxis und dem InputGate von *TrustX* sowie die zugehörigen Prozesse. Sie finden zudem einige Erläuterungen zu den Zusatzprodukten der TrustCenter sowie den daraus folgenden Anforderungen an die Praxissoftware.

Neben dem elektronischen Datenaustausch profitiert die Arztpraxis vom Praxis-Spiegel. Der Praxis-Spiegel erlaubt der Arztpraxis, sich mit anderen Arztpraxen des gleichen Typs (Referenzkollektiv) zu vergleichen.

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 Identitätsprüfung 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 auch Rechnungsantworten (nur Tiers payant) vom Response Gate des TrustCenters empfangen werden.

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



2.1.2 Security und Anonymisierung

Zur Gewährleistung der Internet-Security (Streckenverschlüsselung und Authentifizierung des Datensenders) und zur Verschlüsselung der Patienteninformationen wird die HIN-Security verwendet.

Die Rechnungen und Mahnungen werden verschlüsselt und anonymisiert in *TrustX* abgelegt. Eine Entschlüsselung der Rechnung oder Mahnung findet ausschliesslich in den folgenden zwei Fällen statt:

- Druck der Rechnung durch ein Verarbeitungszentrum
- Versand (eDA) an den Versicherer

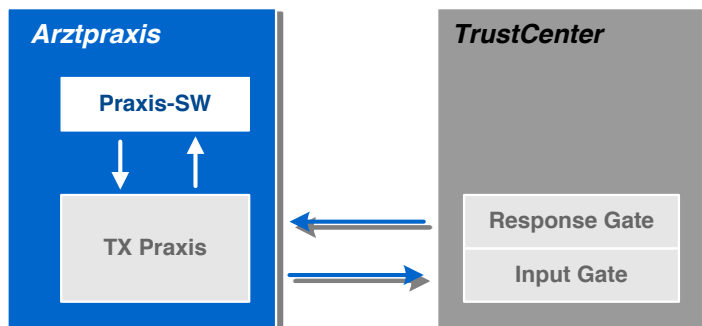
2.1.3 Technische Voraussetzungen zur Datenanlieferung ans TrustCenter

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 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 Kompatibilität TrustX Praxis und HIN

Der HIN-Client setzt den Einsatz von TrustX Praxis Version 3 oder höher voraus. Ältere Versionen von TX Praxis sind nicht kompatibel mit dem HIN-Client.

	ASAS-Client	HIN-Client
TrustX Praxis Version 2.4.x	JA	NEIN
TrustX Praxis Version 3	JA	JA

2.2.2 Datenbereitstellung – Anlieferung an InputGate

Grundsätzlich sind ALLE abgerechneten Rechnungsdaten – nicht nur KVG – **UNMITTELBAR** nach der Fakturierung in der Arztpraxis an das TrustCenter zu übermitteln. Dies ist unter anderem aus folgenden Gründen wichtig:

- In dem alle Rechnungsdaten zuhanden der Statistik angeliefert werden, wird die Qualität der Rechnungskollektive (Statistik) markant verbessert.
- Je schneller die Rechnungen angeliefert werden, desto schneller stehen diese im Praxisspiegel zur Verfügung.
- Im Tiers garat ist es wichtig, dass die XML-Rechnungen im TrustCenter bereits vorliegen, wenn sie vom Versicherer mittels Token (Dokumenten-Identifizier) abgeholt werden möchten. Deshalb müssen die Rechnungen unmittelbar nach der Fakturierung in der Praxis ans TrustCenter übermittelt werden.

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

2.2.3 Rechnungsantworten – Empfang von ResponseGate

TrustX Praxis unterstützt den Tiers payant vollumfänglich. Rechnungsantworten werden im Anschluss an die Rechnungs-/Mahnungsübermittlung automatisch empfangen und im vorgegebenen Verzeichnis

abgelegt. Bedingung ist die korrekte Konfiguration. Siehe dazu auch Kapitel 5.3 „Register <Konfiguration>“ sowie Kapitel 6.2 „Schnittstellendefinition“.

→ **Hinweis:** Der Empfang von Rechnungsantworten ist nur relevant, wenn die Praxis **TP-Rechnungen via TrustCenter** an die Versicherung schickt!

2.2.4 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;
- Formaler Check der bereitgestellten Dateiformate der Rechnungen;
- Bestimmung des Personen-Identifiers² sowie Verschlüsselung und Anonymisierung der Personalien;
- Transport der Daten von der Arztpraxis an das TrustCenter (HIN-gesichert);
- Rechnungsantworten abholen.

2.2.5 Unterstützte Betriebssysteme

TrustX Praxis wurde auf folgenden Betriebssystemen getestet (Stand Januar 2011)

- Windows XP / Vista / 7, Windows Server 2008 / 2008 R2
- Mac OS X 10.4.11 und höher

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

2.3 XML-Standards

Folgende XML-Formate werden von *TrustX Praxis* unterstützt:

- Arztrechnung und Antwort auf eine Arztrechnung (mdInvoiceRequest/Response 4.0)
- Apothekerrechnung und Antwort auf eine Apothekerrg. (PharmacyInvoiceRequest/Response 4.0)
- Rechnungs-Mahnung und Antwort auf eine Mahnung (InvoiceReminderRequest/Response 4.0)
- Generelle Rechnung und Rechnungsantwort (generallInvoiceRequest/Response 4.1)

→ XML 4.3 ist in Vorbereitung.

Die vollständige Dokumentation der XML-Standards finden Sie auf der Website des Forum Datenaustausch unter (<http://www.forum-datenaustausch.ch/de/index.htm>)

² Für statistische Zwecke wird aus den Patienteninformationen ein Hashcode gebildet und im XML-File gespeichert.

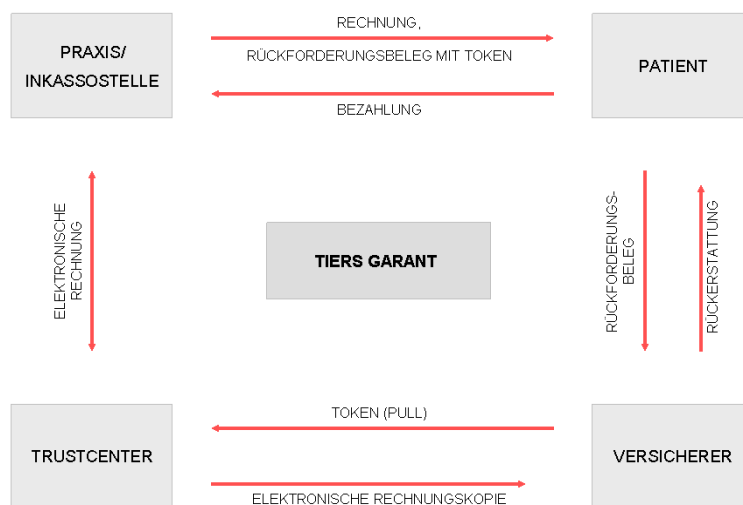
3 Zusatzprodukte der TrustCenter

Das TrustCenter bietet neben dem Praxisspiegel im Zusammenhang mit dem elektronischen Datenaustausch (eDA) verschiedene Zusatzprodukte an. In diesem Kapitel werden die Abrechnungsarten, die Zusatzprodukte sowie die damit in Zusammenhang stehenden Steuerungsmöglichkeiten in der Praxissoftware erläutert.

3.1 Abrechnungsarten

3.1.1 Das „Holprinzip“ (Tiers garant)

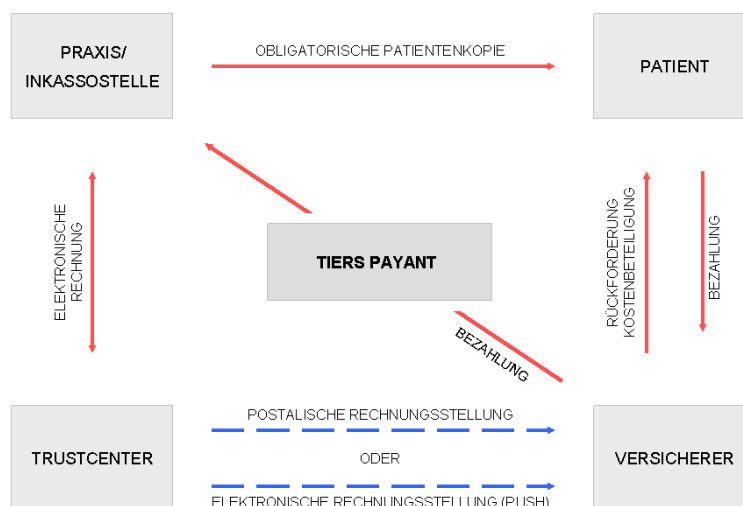
Um den elektronischen Datenaustausch im Tiers garant umsetzen zu können, wurde das so genannte „Holprinzip“ entwickelt. Dabei wird auf den Rückforderungsbeleg ein eindeutiger Dokumenten-Identifizier (= Token) gedruckt. Reicht der Patient den Rückforderungsbeleg bei seinem Versicherer ein, kann dieser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



3.1.2 Das „Bringprinzip“ (Tiers payant)

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

Im Tier soldant entfallen die Verbindungen zum Patient. Das TrustCenter tritt gegenüber Arzt und UVG-Versicherer als Intermediär auf.



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
Rechungsantwort	nicht unterstützt	unterstützt	unterstützt
Mahnung	technisch möglich, jedoch derzeit nicht unterstützt	unterstützt für elektr. und Papierrechnung	unterstützt für elektr. und Papierrechnung
Format Originalrechnung	Frei	eRF 4.0 ³	eRF 4.0
Rechnungskopie	Rückforderungsbeleg	Patientenkopie (im KVG vorgeschrieben)	---
Format Rechnungskopie	eRF 4.0	frei	---
Token	auf Rückforderungsbeleg; nur aufdrucken, falls eine elektronische Rechnungskopie ans TrustCenter geschickt wird und der Arzt am eDA TG teilnimmt!	---	---
ESR	---	Ja, gem. eRF 4.0 auf Originalrechnung	Ja, gem. eRF 4.0 auf Originalrechnung
Gesetze	KVG, VVG, (UVG/MV/IV)	KVG, (VVG)	UVG/MV/IV
eDA-Prinzip	Holprinzip (Pull-Verfahren)	Push-Verfahren	Push-Verfahren

³ eRF: elektronisches Rechnungsformular.

3.2 Zusatzprodukte eDA und Rechnungsdruck

Die TrustCenter bieten den Arztpraxen neben dem Praxis Spiegel weitere Zusatzprodukte im Bereich elektronischer Datenaustausch (eDA) und Rechnungsdruck an.

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. 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 Rechenungskopie)	TX-Print der obligatorischen Rechenungskopie 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

Das Zusatzprodukt „eDA TG“ wird bereits seit 2004 angeboten und wird von den meisten Arztpraxen benutzt. TG Rechnungen machen mehr als 90% aller Arztrechnungen aus.

Der „Druck und Versand“ (D&V) sind 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 (TC). Alle benötigten Angaben werden durch das TC in TX-Administration als Vertrag bzw. Zusatzvertrag abgelegt.

3.3 Steuerung der Zusatzprodukte

Die Steuerung der Zusatzprodukte geschieht über das Flag *resend*⁴ (Original/Kopie). Nachfolgend sind die verschiedenen Vertragsmöglichkeiten (Arzt mit / ohne TC) und deren Steuerungsmöglichkeiten (Original / Kopie) tabellarisch dargestellt. Hinweise zur Implementation in der Praxissoftware sind in Kapitel 4.5 „Hinweise zur Steuerung von Zusatzprodukten“ dargestellt.

	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 Flag Original	-	Wird nicht beachtet	Druck und Versand (D&V) der Rechnung durch das TrustCenter. → D&V von Mahnungen wird derzeit nicht unterstützt	Weiterleitung der Rechnung / Mahnung an Versicherer	Druck und Versand durch das TrustCenter	Weiterleitung der Rechnung / Mahnung an Versicherer
Einfluss von Flag 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 ⁵	JA	-	-	-

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 Praxis

- 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 *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.

⁴ Unter dem Flag *resend* ist das Element *resend* (10170) im XML-File zu verstehen.
Original: *resend* = false, Kopie: *resend* = true.

⁵ Nur, falls sich der Arzt am eDA mit dem Versicherer beteiligt!

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

Zur Nutzung der Zusatzprodukte *Elektronischer Datenaustausch im Tiers payant* (eDA TP) und *Elektronischer Datenaustausch im Tiers soldant* (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 Feinsteuerung folgt für die vertraglich vereinbarten Zusatzprodukte und Optionen über das Flag *resend*. Es werden nur Original-Rechnungen (*resend=false*) prozessiert. Wird z.B. fälschlicherweise eine TG-Rechnung zweimal als Original ans TrustCenter geschickt, wird diese zweimal gedruckt und verschickt.

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

Wählt eine Arztpraxis den *eDA TP* 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 eine Steuerung durch die Praxissoftware NICHT notwendig. Grundsätzlich fließen alle Rechnungen in den Praxisspiegel ein. Allfällig doppelt angelieferte Rechnungen werden auf der Ebene Statistik eliminiert. Das Flag *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 der *Patientenkopie TP*: Es werden nur Originalrechnungen prozessiert.
- Praxisspiegel: Es erfolgt keine Steuerung; Token und das Flag *resend* sind nicht relevant. Doppelt angelieferte Rechnungen werden für die Statistik eliminiert.

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 entweder mit eigener Benutzeroberfläche (Kapitel 5 "Funktionale Beschreibung von TrustX Praxis") betrieben werden oder über eine Schnittstelle direkt aus der Praxissoftware angesteuert (Kapitel 6 „Schnittstelle zu TrustX Praxis“) und so vollständig in die Praxissoftware integriert werden (Abschnitt 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. Dies bedeutet insbesondere, dass das Flag *resend* (Original/Kopie) zur Feinsteuerung differenziert gesetzt werden kann.
- 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 (siehe Abschnitt 2.2.1 „Kompatibilität TrustX Praxis und HIN“). Es sind also **keine** Modifikationen an den elektronischen Rechnungen notwendig, damit sie von *TrustX Praxis* ans TrustCenter geschickt werden können. Diese müssen lediglich in ein 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, den Rückforderungsbeleg mit dem Token zu drucken und die elektronischen 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 Kapitel 6 „Schnittstelle zu TrustX Praxis“) Dies erlaubt es zusätzliche Konfigurationsparameter, die in der Praxissoftware bekannt sind, – bspw. das Rechnungsverzeichnis – in *TrustX Praxis* zu setzen. Anschliessend 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 Kapitel 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 Praxis. 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 3.3 „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 als **Kopie** (resend=true) ans TrustCenter geschickt werden. Doppelte Rechnungen an den Patienten entstehen ebenfalls, wenn dieselbe Rechnung mehrfach als Original ans TrustCenter geschickt wird.

4.5.2 eDA TP und TS

Im elektronischen Datenaustausch nach dem Push-Prinzip (*eDA TP* und *eDA TS*) kann es, neben der Möglichkeit eine einzelne Rechnung als „Original/Kopie“ und „Druck in Praxis nein/ja“ zu kennzeichnen, sinnvoll sein, diese Selektion für einzelne Versicherte und Gesetze vornehmen zu können.

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 4.0 Standard fordert, dass die folgenden drei Felder eine Rechnung/Mahnung eindeutig identifizieren:

Rechnung	Mahnung
Biller_EAN (11402)	Biller_EAN (11402)
Invoice_ID (10153)	Reminder_ID (10186)
Invoice_TimeStamp (10152)	Reminder_TimeStamp (10185)

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. Diese sind (blau = zusätzlich gegenüber XML 4.0):

Rechnungsinformation	Bemerkung
Biller_EAN (11402)	gem. XML 4.0
Invoice_ID (10153)	gem. XML 4.0

Invoice_TimeStamp (10152)	gem. XML 4.0
Provider ZSR (11803)	Ist zusammen mit der Angabe des TrustCenters die eindeutige Identifikation des TrustCenter-Kunden innerhalb des TrustX Systems.
Amount (10330)	Dient der Unterscheidung von Rechnung und Storno (innerhalb des Logfiles).
Invoice / tierspayant / purpose (11265)	Angabe über den Rechnungszweck im Falle eines TP-Storno zur Unterscheidung von Rechnung und Storno (innerhalb des Logfiles).

Diese sechs Werte müssen eine Rechnung eindeutig identifizieren. Weicht ein Wert ab, handelt es sich nicht mehr um eine Doublette und das XML-File kann ans InputGate des TrustCenters angeliefert werden. 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 eine Rechnung explizit nochmals im TrustCenter gedruckt werden soll.

4.6.2 Erkennung doppelter Rechnungen in der Statistik-Datenbank

Immer wieder tritt jedoch der Fall auf, dass Rechnungen versehentlich mehrmals an das TrustCenter geliefert werden. Das Regelwerk, welches beim Einleseprozess in die Statistikdatenbank zur Anwendung kommt, erkennt doppelte Rechnungen ebenfalls und markiert diese, damit sie nicht in die Statistik (Praxisspiegel) einfließen. Doppelte Rechnungen werden hier nach den folgenden Kriterien erkannt (gelb = zusätzlicher Check gegenüber *TrustX Praxis*):

Rechnungsinformation	Bemerkung
Biller_EAN (11402)	gem. XML 4.0
Invoice_ID (10153)	gem. XML 4.0
Invoice_TimeStamp (10152)	gem. XML 4.0
Provider ZSR (11803)	Ist zusammen mit der Angabe des TrustCenters die eindeutige Identifikation des TrustCenter-Kunden innerhalb des TrustX Systems.
Amount (10330)	Dient der Unterscheidung von Rechnung und Storno.
Invoice / tiers_payant / purpose (11265)	Angabe über den Rechnungszweck im Falle eines TP-Storno zur Unterscheidung von Rechnung und Storno.
TrustCenter	Ist zusammen mit der Angabe der Provider ZSR die eindeutige Identifikation des TrustCenter-Kunden innerhalb des TrustX Systems.
Biller ZSR (11403)	Es werden alle in eine Rechnungsstellung involvierten Parteien berücksichtigt.
Patient-Id (16720)	Analog Biller ZSR.

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 (11402)	gem. XML 4.0
Reminder_ID (10186)	gem. XML 4.0
Reminder_TimeStamp (10185)	gem. XML 4.0
Amount (10330)	Dient der Unterscheidung von Rechnung und Storno (innerhalb des Logfiles)

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

WICHTIG: Mehrfach ans InputGate gelieferte Mahnungen werden **mehrfach** prozessiert!

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)	Betrag	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 Zuverlässigkeit 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 Betragsfeldern mit minus eins multipliziert 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 Feld 11265 (XML 4.0) = „annulment“ (request / invoice / tiers_payant / purpose) ebenfalls als Storno erkannt.

Die *TrustX Praxis* Versionen Windows 2.1.14 sowie MAC 2.1.15 (und älter) können das Feld *purpose* (11265) jedoch noch nicht verarbeiten. Deshalb musste als Workaround zusätzlich der <Gesamtbetrag> (10330) auf minus gesetzt werden.

Ab *TrustX Praxis* Version 2.2 wird auch das Feld *purpose* (11265) ausgewertet und ein so gekennzeichnetes Storno als solcher erkannt. Der Gesamtbetrag muss daher nicht mehr auf minus gesetzt werden, sofern request / invoice / tiers_payant / purpose auf „annulment“ gesetzt wird.

4.8 XML 4.0

Seit 1. Juli 2005 ist der XML 4.0-Standard für den Rechnungsdatentransfer obligatorisch. TrustX Management AG hat die Unterstützung des Rechnungsformates 3.0 per Ende 2007 eingestellt. Rechnungen im Format XML 3.0 werden vom InputGate zurückgewiesen.

XML 4 unterstützt das Holprinzip (Pull-Verfahren) im Tiers garant explizit (Felder 11151-11153). Es ist somit möglich, das Token, welches auf dem Rückforderungsbeleg gedruckt wird, auch in der XML-Datei zu speichern. Für TG-Rechnungen werden diese Felder wie folgt ausgefüllt:

`tc_demand_id (11151) = 0`

Die tc_demand_id wird vom TrustCenter bei der Rechnungsauslieferung gesetzt; Das Feld ist aber obligatorisch, deshalb Wert 0 eintragen.

`tc_token (11152) = Token`

Beispiel: 5000001570507>332435000001111122222333337+ 010033018>

`insurance_demand_date (11153) = Verarbeitungsdatum in der Arztpraxis`

Datum wird ebenfalls durch das TrustCenter richtig gesetzt; Feld ist aber obligatorisch, deshalb Fakturierungsdatum eintragen.

Wir bieten Ihnen die Möglichkeit, Ihre Software auch XML 4 zertifizieren zu lassen. Weitere Informationen zur Zertifizierung finden Sie in Kapitel „4.12.2 Zertifizierung“.

4.9 Hinweise zu PLZ in XML-Rechnungen

Der XML-Standard definiert das Feld für die PLZ (Element postal/zip) als String (9). TX Praxis bis Version 2.1 hat nur bei Schweizer-Postleitzahlen, bestehend aus vier Ziffern, eine gültige Patienten-Id berechnet. Bei PLZ in anderen Formaten (ausländische PLZ) wurde eine Fehler-Patienten-Id erzeugt und die XML-Rechnungen in der Statistik nur teilweise berücksichtigt. TX Praxis ab Version 2.3 unterstützt nun auch ausländische Postleitzahlen.

WICHTIG: Für die statistischen Auswertung ist es zentral, dass gültige PLZ entsprechend den Richtlinien des jeweiligen Landes eingetragen werden, d.h. für die Schweiz eine 4-stellige Zahl. Bei ausländischen Adressen ist zudem gemäss XML-Standard das Land (Element postal/zip/countrycode=) explizit anzugeben (beim Fehlen des Landes gilt als Default Land = CH).

4.10 Neue AHV-Nummer

Die neue, 13-stellige AHV-Nummer ist (analog der 11-stelligen Nummer) gemäss XML-Schema ebenfalls erlaubt. Die Elemente `ssn` erlauben das folgende Pattern: "[1-9][0-9]{10}" | "[1-9][0-9]{12}".

4.11 Datensammlung für Praxisspiegel und eDA über einen Intermediär

Im Regelfall erfolgen der eDA und das Datensammeln für den Praxisspiegel über das TrustCenter. Dabei wird das TrustCenter (EAN-Nummer) als Intermediär in der XML-Rechnung eingetragen (Feld 10052). Falls XML-Rechnungen für den eDA über einen anderen Weg, z.B. via MediData oder H-Net, zu einem Versicherer gelangen, können die Rechnungen trotzdem zu statistischen Zwecken ans TrustCenter übermittelt werden. Dabei ist folgendes zu beachten:

- Als Intermediär wird die EAN des Transporteurs belassen (Feld 10052).
- Bei der Rechnung handelt es sich um eine Kopie. Deshalb muss die Rechnung als Kopie (`resend = true`) gekennzeichnet werden (Feld 10170).

Die Gefahr, dass eine Rechnung zweimal zum Versicherer gelangt, besteht selbst dann nicht, wenn eine Rechnung von einem Transporteur als Original zu statistischen Zwecken ans TrustCenter geschickt wird, da eine Rechnung nur weiter geleitet wird, wenn eine Arztpraxis explizit die Zusatzprodukte `eDA TP` oder `eDA TS` mit dem TrustCenter abgeschlossen hat. Im `eDA TG` besteht

die Gefahr nicht, weil mit dem Aufdruck des Tokens auf dem Rückforderungsbeleg angezeigt wird, wo die Rechnung abgeholt werden kann.

4.12 Tests und Zertifizierung

4.12.1 TrustCenter "TC Test"

Zu Testzwecken steht ein Test-TrustCenter „TC Test“ zur Verfügung. Bitte verwenden Sie für Tests **AUSSCHLIESSLICH** "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>

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

4.12.2 Zertifizierung

Jeder Praxissoftware-Hersteller hat die Möglichkeit, seine Praxissoftware zertifizieren zu lassen. Dabei geht es insbesondere darum, die Unterstützung der Zusatzprodukte und die Qualität der XML-Rechnungen sicher zu stellen. Der Hersteller wie auch die Anwender erhalten dadurch die Gewähr, dass die Rechnungsqualität den verschiedenen Anforderungen der Versicherer und der TMA genügt.

4.12.3 Dokumentationen und Downloads

Informationen zu *TrustX*, *TrustX Praxis* sowie der *Zertifizierung* finden Sie unter www.trustx.ch.

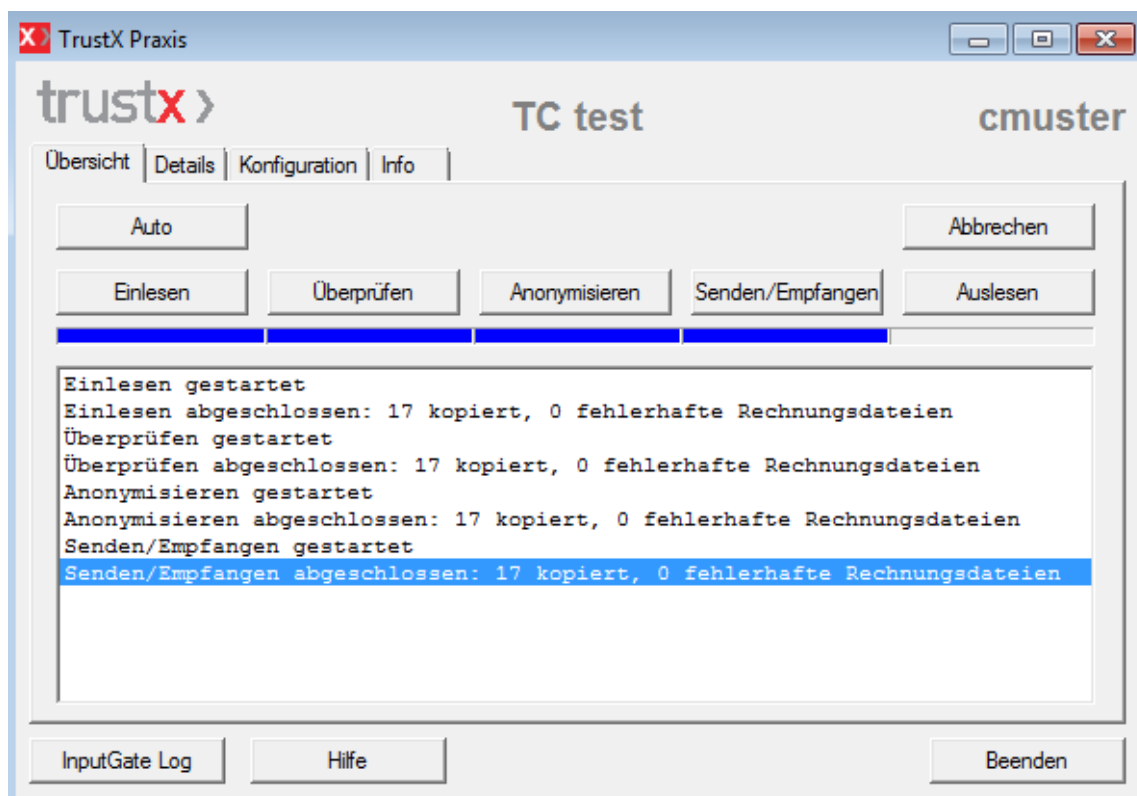
5 Funktionale Beschreibung von TrustX Praxis

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

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

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

- <Übersicht> Beinhaltet alle wichtigen Funktionen zur Bedienung
- <Details> Gibt Einblick in verschiedene 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, um die Rechnungsanlieferung online einzusehen.

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"

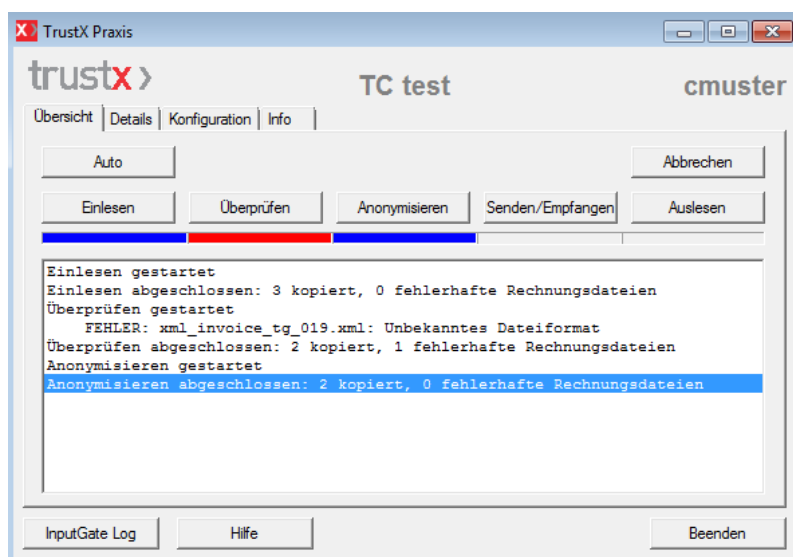
Über den Button <Auto> werden alle Schritte mit den vorgewählten Konfigurationsparametern automatisch abgearbeitet. Der blaue Balken zeigt den Fortschritt der Verarbeitung an.

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 <i>TrustX Praxis</i> gemäss den Einstellungen im Internet Explorer aufgebaut und wieder geschlossen.
<Auslesen>	Daten werden ins Antwortverzeichnis (im Register <Konfiguration> definiert) verschoben.

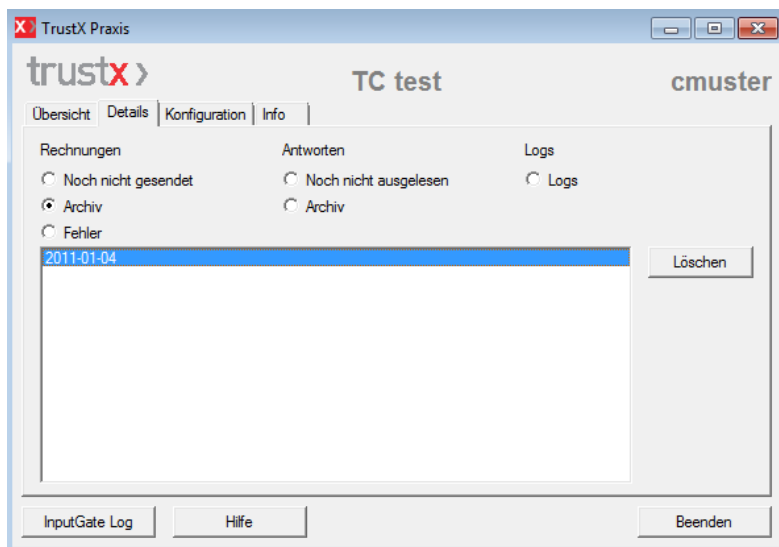
5.1.3 Darstellung des Fortgangs



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 "0

Register <Details>", Seite 24.

5.2 Register <Details>

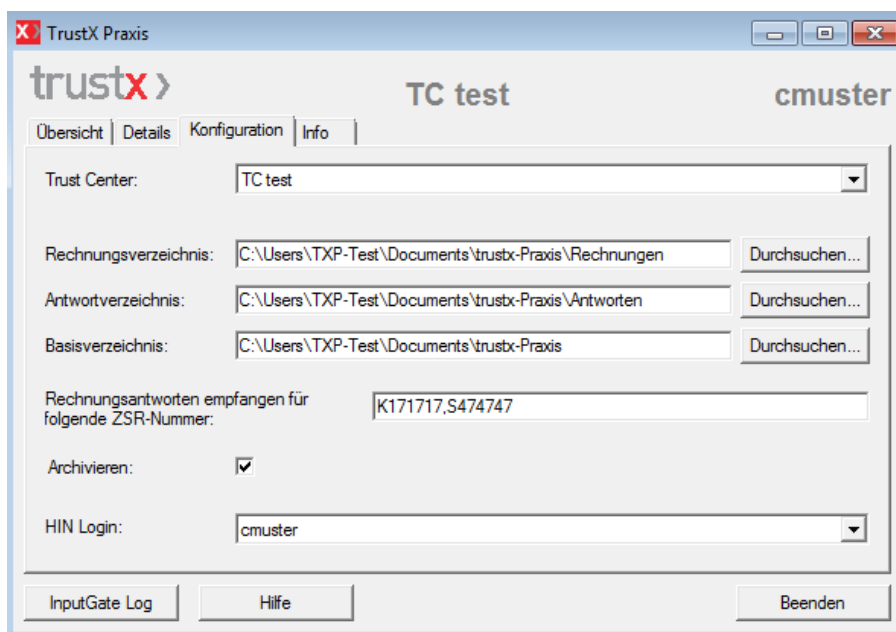


Das Register <Details> stellt verschiedene Detailansichten zur Verfügung. Über die Auswahlfelder (oben) können entsprechende Listen im darunter liegenden Fenster dargestellt werden.

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. |

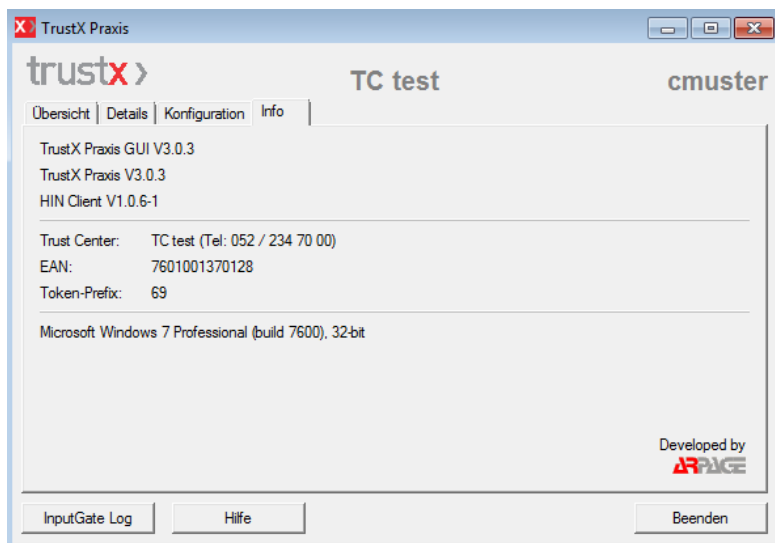
5.3 Register <Konfiguration>



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 "TC Test".
- <Rechnungsverz.> 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 "Verzeichnisstruktur" (Seite 31).
- <Rechnungsantworten empfangen für folgende ZSR-Nummer> Tragen Sie hier Ihre ZSR-Nummer(n) ein, für welche Sie Rechnungsantworten empfangen möchten.
- 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 angegebenen 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"⁶ definiert (siehe 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 6.3 "Events").

6.2 Schnittstellendefinition

Hinweise:

- Änderungen gegenüber *TrustX Praxis* Version 2.4 sind **gelb hervorgehoben**
- Zum besseren Verständnis siehe auch Kapitel 5 „Funktionale Beschreibung von TrustX Praxis“

Schnittstellendefinition	Bemerkung
// \$Id: trustx.idl,v 1.6 2007/03/26 07:46:18 franco Exp \$	
#include "client.idl"	
module Trustx {	
interface ITrustx {	
typedef sequence<string> strings;	
readonly attribute string trustxVersion	Version von <i>TrustX Praxis</i>
readonly attribute strings trustCenters	Liste aller TrustCenter
attribute string trustCenter	TrustCenter auswählen
attribute string inputDirectory	Pfad für Rechnungsverzeichnis
attribute string workDirectory	Pfad für Basis- oder Arbeitsverzeichnis
attribute boolean withPDF;	Option elektronische Rechnungskopie und PDF-Datei
attribute boolean withArchive;	Option verschickte Rechnungen archivieren
readonly attribute string esr;	TrustCenter-Code für "Dokumenten-Identifizierer ohne PDF" lesen, abhängig vom gewählten TrustCenter
readonly attribute string esrPDF;	TrustCenter-Code für "Dokumenten-Identifizierer mit PDF" lesen
readonly attribute string session;	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
readonly attribute string asasVersion;	Version des ASAS Clients lesen
readonly attribute strings asasLogins;	Liste mit verfügbaren ASAS Logins anzeigen
attribute string asasLogin;	ASAS Login für <i>TrustX Praxis</i> vorgeben
attribute boolean activateGUI;	Zur Zeit nicht implementiert, siehe Kapitel „Aufruf von TrustX Praxis aus der Praxissoftware“, Seite 14.
attribute boolean debug;	Debug-Meldungen generieren ja/nein
void Auto();	Automatisches Verarbeiten und Senden der

⁶ Object Management Group Interface Definition Language

Schnittstellendefinition	Bemerkung
	Rechnungsdateien analog dem Button <Auto>
void Read();	Verschieben der Rechnungsdateien aus ..\{Rechnungsverzeichnis}\.. in ..\{basisverzeichnis}\input analog dem Button <Einlesen>
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 geschrieben.
void Send();	Rechnungsdaten werden aus ..\{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 den eingestellten ASAS-Login gesucht</p> <p>ZSR1,ZSR2... Liste von "Komma-delimited" ZSR-Nummern.</p> <p>Es werden alle Antworten für die angegebenen ZSR-Nummern und für den eingestellten ASAS-Login gesucht.</p>
// Added in V3	
readonly attribute string getUpdateMsg;	Gibt die Update-Meldung aus, welche auf dem Server hinterlegt ist.
};	
};	

6.3 Events

Jedes TrustX-Modul 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
<code>enum MsgClass {</code>	Message Class
<code> mcSystem = 0,</code>	Meldung kommt vom Betriebssystem (z.B. Schreibfehler) <code> wird vom Betriebssystem übernommen
<code> mcASAS = 1,</code>	Meldung kommt von der ASAS-Infrastruktur (z.B. Zertifikat nicht gefunden); (nur Module <code>moPEncrypter</code> , <code>moPSender</code>)
<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
<code>enum MsgOrigin {</code>	Message Origin
<code> moPReader = 0,</code>	Meldung kommt vom TrustX Praxis-Modul - Reader
<code> moPChecker = 1,</code>	Meldung kommt vom TrustX Praxis-Modul - Checker
<code> moPAnonymizer = 2,</code>	Meldung kommt vom TrustX Praxis-Modul - Anonymizer
<code> moPEncrypter = 3,</code>	Meldung kommt vom TrustX Praxis-Modul - Encrypter
<code> moPSender = 4,</code>	Meldung kommt vom TrustX Praxis-Modul - Sender
<code> moIGReceiver = 5,</code>	Meldung kommt vom TrustX-InputGate-Modul - Receiver
<code> moIGChecker = 6,</code>	Meldung kommt vom TrustX-InputGate-Modul - Checker
<code> moPDownloader = 7,</code>	Meldung kommt vom TrustX Praxis-Modul - Downloader
<code> moPWriter = 8,</code>	Meldung kommt vom Auslese-Modul
<code>};</code>	

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 ASAS 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

Klassifizierung	Bemerkung
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)
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 Module ("Prozesse") (Auto, Read etc.) werden asynchron verarbeitet. 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 Seite 35).

Wird ein Modul 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 <i>TrustX Praxis</i> . 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 <i>Archivieren</i> im Register <Konfiguration> angewählt wurde
..\{bz}\responses-archive	In diesem Verzeichnis werden alle empfangenen Rechnungsantworten abgespeichert, sofern die Option <i>Archivieren</i> 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 ist das Rechnungs- und das Antwortverzeichnis zu unterscheiden. Diese können unabhängig vom Basisverzeichnis gewählt werden.

6.7 Wichtige Dateien von TrustX Praxis

6.7.1 Programmfiles

trustxui.exe *TrustX Praxis* Anwendung / Exe-Datei (Windows)
 mscvcp71.dll Dlls von *TrustX Praxis*
 mscvcr71.dll
 trustx.dll

trustx-Praxis *TrustX Praxis* Anwendung / ausführbare UNIX-Datei (Mac)

6.7.2 Konfigurations- und Logfiles

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

Filename	Inhalt	pro Mandant	Pfad
trustx.log	Rechnungsinformationen, welche für den Doublettencheck benötigt werden.	Ja	Im Root vom eingestellten Basisverzeichnis, i.d.R. Windows: C:\Dokumente und Einstellungen\MEIN-PROFIL\Eigene Dateien\trustx-Praxis\ Mac: /Users/MY_USER/Documents/trustx-Praxis
YYYY-MM-DD.log	Detaillierte Loginformationen zu jedem Verarbeitungsschritt	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.fmwk.plist	Konfigurationseinstellungen	Ja	Windows: C:\Dokumente und Einstellungen\MEIN-PROFIL\Anwendungsdaten\Arpage\trustx-Praxis Mac: /Users/MY_USER/Library/Preferences/ch.trustx.praxis.fmwk.plist
trustx.xml	Informationen über die TrustCenter → wird bei der Installation angelegt	Nein	Windows: C:\Programme\Arpage\TrustX Praxis Mac: /Applications/trustx-Praxis.app/Contents/SharedFrameworks/trustx.frameworks/Versions/A/Resources/trustx.xml
trustx.xml	Informationen über die TrustCenter → wird bei der Rechnungsübermittlung angelegt	Ja	Windows: C:\Dokumente und Einstellungen\MEIN-PROFIL\Anwendungsdaten\Arpage\trustx-Praxis Mac: /Users/MY_USER/Library/Application Support/trustx-Praxis/trustx.xml

7 Anhänge

Anhang 1: ESR-Zeile als Dokumenten-Identifizier (Token)

Als Dokumenten-Identifizier (Token) wird eine ESR-Zeile (esr9-Verfahren mit 27-stelliger ESR-Ref.-Nr. und Betrag) verwendet, die sich von der ESR-Zeile der Rechnung ableitet. 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 versichererseitig 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	52
hawatrust	53
medkey	54
PonteNova	55
syndata	56
TC Aargau	57
thurcare	58
TC Ticino	59
TC züridoc	60
trustmed	61
GTS ⁷	62
Physiotrust	63
TC Test	69

⁷ General Transport Service

Anhang 2: TrustX-Event Beispiel (VB6)

Der Status `INFO_MODULE_DONE` kommt immer als letzte Meldung im Event-Ablauf, diese 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

```