SwissdecAdapter

Detailspezifikation

Versionsgeschichte			
Version 00.10	2012-02-02	Marco Stettler	Basisversion
Version 00.80	2012-02-08	Marco Stettler	Workshop CTP
Version 00.81	2012-02-09	Marco Stettler	Workshop AWK/CTP
Version 00.82	2012-02-15	Marco Stettler	Vorabinfo für Info-Pilot-kantone
Version 00.90	2012-02-23	Marco Stettler	Version vor abschliessen-der Detailspez. Sitzung
Version 00.91	2012-02-29	Marco Stettler	Nach Detailspez. Sitzung vom 24.02.2012. Schema noch nicht fertig, deshalb nicht V1
Version 00.92	2012-04-13	Marco Stettler	Draft Version für Pilot-Phase
Version 00.93	2012-04-25	Marco Stettler	Befunde AWK und CTP eingepflegt
Version 00.94	2012-05-03	Marco Stettler	Befunde Runde 3 AWK und CTP eingepflegt
Version 00.95	2012-02-09	Marco Stettler	Draft Version für Final
Version 00.96	2013-02-25	Marco Stettler	Befunde AWK Review
Version 01.00	2013-05-02	Marco Stettler	Finale Version
Version 01.10	2020-02-12	Marco Stettler	Anpassungen ELM V5
Version 01.11	2020-03-06	Christoph Brunner	Anpassungen ResultIden-tifier
Version 02.00	2020-04-03	Marco Stettler	Final für ELM V5 (20200220)
Version 02.01	2020-04-07	Marco Stettler	Befunde AWK Review

Inhaltsverzeichnis

1. Allgemeines 1
1.1. Vorausetzungen 1
1.2. Übersicht zu den verschiedenen Dokumenten 1
1.3. Architekturentscheide aus dem Konzept 2
1.4. Verschiedene konzeptionelle Details zur Übermittlung 2
1.4.1. Terminologie 2
1.4.2. Swissdec Prozess 3
1.4.3. Declaration ID 3
1.4.4. Mehrfachübermittlung 4
2. Funktionale Anforderungen 5
2.1. Ziele und Hauptfunktionen des Systems 5
2.1.1. Aus fachlicher Sicht 5
2.1.2. Aus technischer Sicht 5
2.2. Grobe Übersicht 5
2.3. Detaillierte Übersicht 6
2.4. Wartungsfenster registrieren 6
2.5. Konfiguration einstellen 7
2.5.1. Konfigurationsoptionen 7
2.5.2. Verhalten bei Fehlkonfiguration 9
3. Ergänzende Anforderungen 10
3.1. Rahmenbedingungen 10
3.2. Anforderungen Infrastruktur 10
3.2.1. Hardware 10
3.2.2. Software 10
3.2.3. SSL/TLS Terminierung 11
3.3. Zusammenfassung einer Institution 11
3.4. Kommunikationsstandards 12
3.5. Verfügbarkeit 12
3.6. Performance / Durchsatz 13
3.7. Mehrfachübermittlungen 13
4. Architektur 14
4.1. Systemübersicht 14
4.2. Systemteile 14
4.2.1. Schnittstelle Receiver - Integration 14
4.2.2. Installationsvarianten 15
5. Schnittstellen 17
5.1. Swissdec 17
5.2. sM-Client 17
5.2.1. Format der Daten 17
5.2.2. Verarbeitungsprinzipien 18
5.2.3. Behandlung von Fehlern 22
5.3. Kostenpflichtige Zusatzfunktionalität 22
5.3.1. SSL/ Kunde (Schuldner steuerbarer Leistung) Checker 22
5.3.2. Tarifcode Checker 22
5.3.3. Fach-Anwendung 22
5.3.4. Mail 22
A. Anhang 23
A.1. Referenzierte Dokumente 23
A.2. Glossar 23
A.3. Beispiel Dateien 23

Abbildungsverzeichnis

1.1. Was wird wo spezifiziert (Swissdec Spez-Skizze) 1
1.2. System Übersicht 2
1.3. Swissdec Prozess aus [KZPTQST] 3
1.4. DeclarationID 3
2.1. Use-Cases Übersicht 5
2.2. Use-Cases detailliert 6
3.1. Installationsskizze SwissdecAdapter 11
4.1. Architektur STEP/SwissdecAdapter 14
4.2. Installationsvariante En bloc 15
4.3. Installationsvariante Verteilt 16
5.1. Swissdec Schema-Imports 17
5.2. Request verarbeiten 18

Tabellenverzeichnis

2.1. Use-Case Wartungsfenster registrieren 6
2.2. Use-Case Konfiguration einstellen 7
2.3. SwissdecAdapter Receiver Konfiguration 7
2.4. SwissdecAdapter Integration Konfiguration 8
4.1. Schnittstelle Receiver - Integration 15
5.1. Fehler im SwissdecAdapter 22

Liste der Beispiele

A.1. envelope.xsd 23

1. Allgemeines

Im vorliegenden Dokument werden die Ergebnisse der Detailspezifikationsphase des Projekts "Swissdec Adapter" für die Kantonalen Steuerverwaltungen im Zusammenhang mit dem Projekt "Lohnstandard-Quellensteuer" zusammengefasst. Ziel des Dokuments ist es, sämtliche Ergebnisse aus der Detailspezifikation vorzustellen, sowie die daraus resultierenden Anforderungen für die Produktion aufzuzeigen. Diese Ergebnisse sind technischer Natur. Als Basis für die Detailspezifikation dienen sowohl die Erkenntnisse aus dem Prototyp als auch das [KZPTQST] der Swissdec.

1.1. Vorausetzungen

Dieses Dokument basiert auf folgenden Spezifikationen:

- [SDOVERVIEW] - Übersicht zum Swissdec Standard
- [KZPTQST] - Konzept Swissdec Lohnstandard-CH Quellensteuer

Ergänzend wird in dieser Detailspezifikation auf folgende Dokumente verwiesen:

- [ENDRECREQ] - Anforderungen an Endempfänger
- [ACKNOTI] - Spezifikation der Mitteilungen
- [RECSEC] - Security- Anforderungen an Endempfänger

Die Swissdec-Spezifikation ist zu beziehen via www.swissdec.ch. Zur Zeit der Erstellung dieses Dokuments liegt diese Spezifikation in der Version 5 vor und liefert damit die Grundlage zu diesem Dokument.

1.2. Übersicht zu den verschiedenen Dokumenten

Die Dokumente in der Folgenden Skizze sind auf www.swissdec.ch verfügbar.

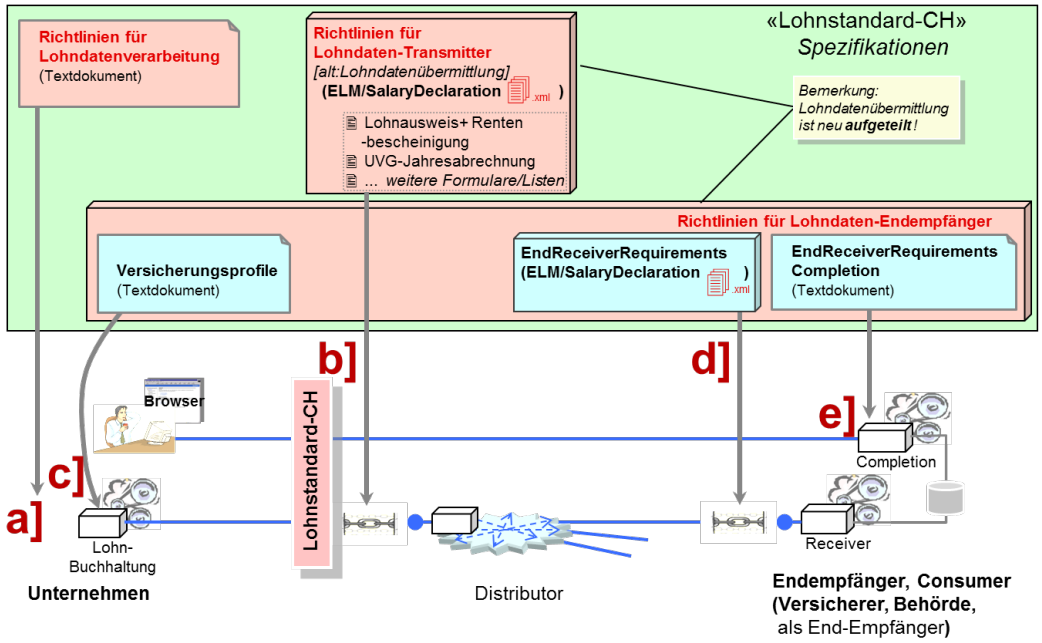


Abbildung 1.1. Was wird wo spezifiziert (Swissdec Spez-Skizze)

a] Richtlinien für Lohndatenverarbeitung (Seite ERP)

b] Richtlinien für Lohndaten-Transmitter (Seite ERP)

c] Richtlinien für Lohndaten-Empfänger (Versicherungsprofile)

d] Richtlinien für Lohndaten-Empfänger ([ENDRECREQ])

e] Richtlinien für Lohndaten-Empfänger (EndReceiverRequirement für Completion)

Zusätzlich wird die Standard-Implementierung des Endempfängers Teil d] und seine Kommunikation zum sM-Client in diesem Dokument beschrieben. Diese Beschreibung wird durch das Technische Design sM-Client/CTP ergänzt.

1.3. Architekturentscheide aus dem Konzept

Im[KZPTQST], Kapitel 6.2.2, sind verschiedene Kopplungsvarianten beschrieben. Dieses Dokument bezieht sich auf die Variante 3, direkte Integration des Distributor/ELM in den sM-Client, ohne sedex zu verwenden. Der SwissdecAdapter wird architektonisch analog dem sedex-Adapter in den sM-Client integriert.

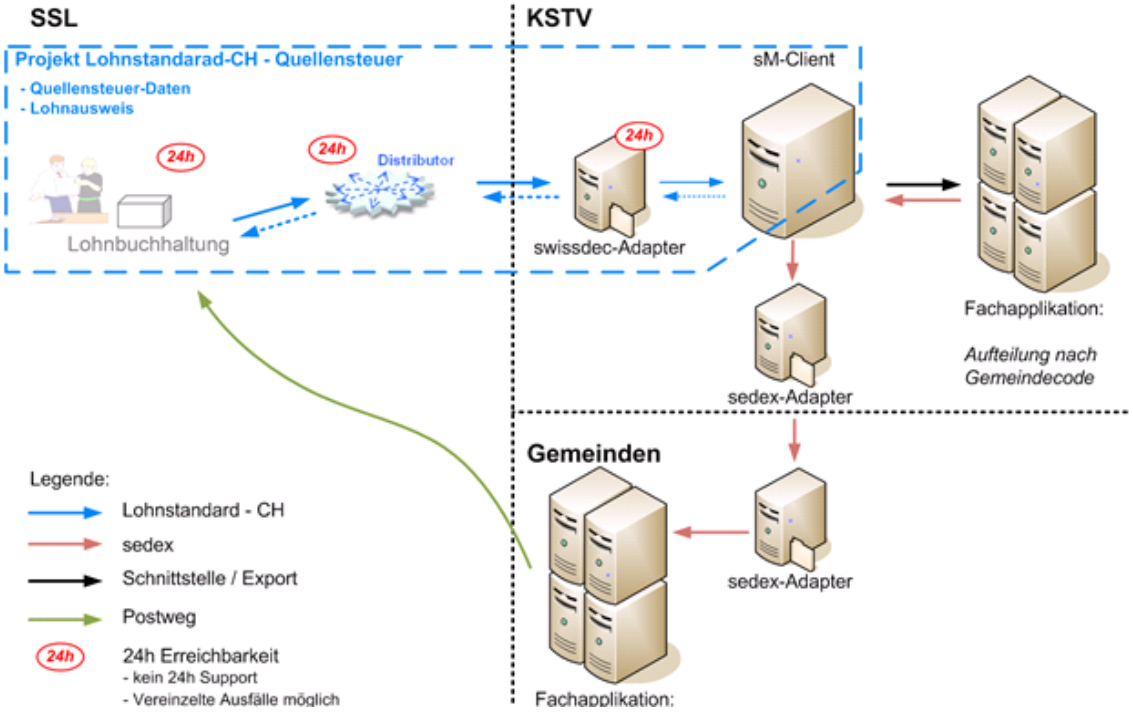


Abbildung 1.2. System Übersicht

1.4. Verschiedene konzeptionelle Details zur Übermittlung

1.4.1. Terminologie

In diesem Projekt besteht die Problematik, dass sowohl die Fachexperten wie auch Techniker das Kürzel „SSL“ verwenden, jedoch mit unterschiedlicher Bedeutung.

Fachliche Bedeutung	„Schuldner steuerbarer Leistung“, also Kunde der Steuerbehörde.
Technische Bedeutung	„Secure Socket Layer“, ein verschlüsseltes Transportprokoll (wie zum Beispiel auch bei HTTPS).

Deshalb werden in diesem Dokument konsequent die beiden Ausdrücke „SSL/Kunde“ für die fachliche beziehungsweise „SSL/TLS“ für die technische Bedeutung verwendet.

1.4.2. Swissdec Prozess

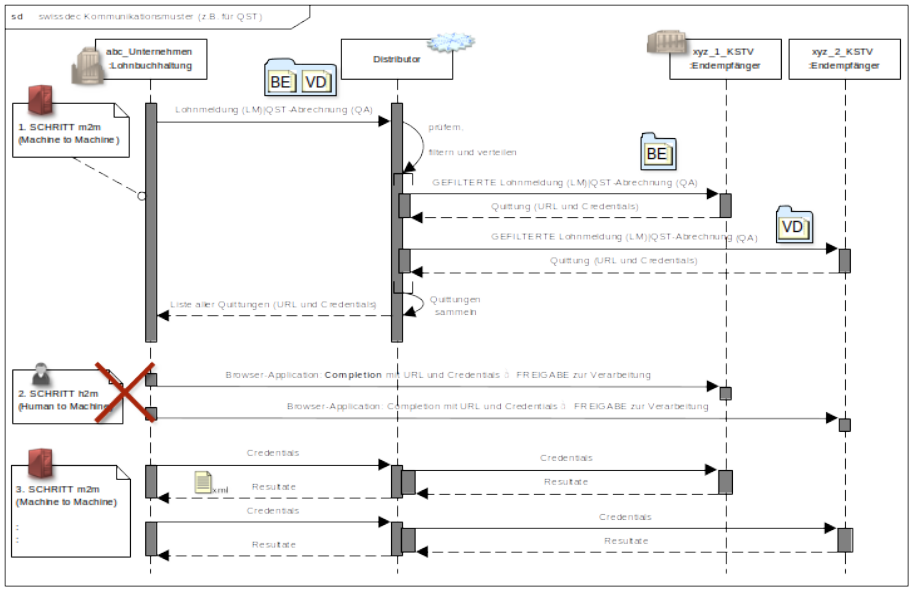


Abbildung 1.3. Swissdec Prozess aus [KZPTQST]

1. Die Lohnbuchhaltung extrahiert die notwendigen Lohndaten und übermittelt diese via Distributor an die Endempfänger Institution (Versicherer, Behörde, …). Der Endempfänger prüft die Lohndaten und quittiert diese mit einer URL und Credentials (Key, Password)

2. Die Lohnbuchhaltung startet einen Browser mit der URL aus der Endempfänger-Quittung. Der Lohnbuchhalter meldet sich an (Key, Password). Er kontrolliert seine gesendeten und vom Endempfänger geprüften Lohndaten. Der Lohnbuchhalter kann die Lohndaten zur Verarbeitung freigeben (oder löschen). Danach können die freigegebenen Lohndaten verarbeitet werden.

3. Falls der SwissdecAdapter/sM-Client und die Fachapplikationen entsprechend konfiguriert sind, können weitere Daten beim Endempfänger mittels der Credentials abgeholt werden (QST-Abrechnungsresultat)

Für die Domänen „Quellensteuer“, "Lohnausweis" und "Grenzgäner-Meldung" verzichtet die Swissdec auf den 2. Schritt (Completion,[KZPTQST], Kapitel 2.5.6).

1.4.3. Declaration ID

Gleiche ID in allen Requests (Distributor Endempfänger), Responses (Distributor Transmitter), Masken und Pdf's, welche zum identischen Geschäftsvorfall gehören. Die Swissdec spezifiziert als identischen Geschäftsvorfall eine Meldung, sowie alle nachfolgenden Schritte für diese Meldung (Completion, GetResult). Eine Korrektur/ Substitutionsmeldung referenziert die originale Meldung, erhält jedoch eine neue Declaration ID und ist somit aus Sicht Swissdec ein neuer Geschäftsvorfall. Damit kann sie als Fallnummer für den Support verwendet werden. Diese wird nur vom Distributor vergeben, kontrolliert und jedem Partner (Lohnbuchhaltung/Transmitter und Endempfänger) in die zusammengehörigen Responses oder ConsumerRequests automatisch eingesetzt.

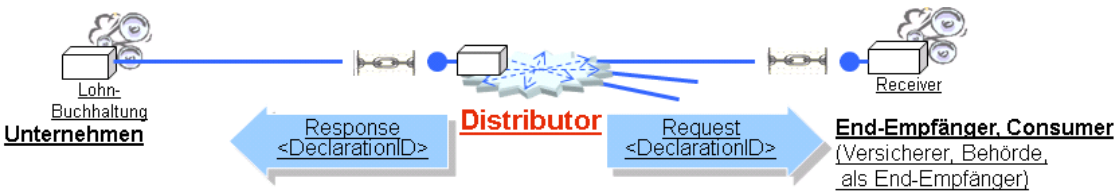


Abbildung 1.4. DeclarationID

1.4.4. Mehrfachübermittlung

Die Mehrfachübermittlung tritt auf, wenn ein Unternehmen mehrere Lohnmeldungen pro Institution und Vertrag übermittelt. Die Mehrfachübermittlung muss möglich sein, d.h. unabhängig vom Empfangs-Zustand bereits empfangener Meldungen möchte das Unternehmen die Lohnmeldung ein weiteres Mal übermitteln.

In der Mehrfachübermittlung gibt es folgende Fallunterscheidungen:

Identische Meldungen = Dubletten (siehe Dubletten-Detektierung,[ENDRECREQ])

- Meldungen ohne „Substitution“ (keine explizite Abhängigkeit)
- Meldungen mit „Substitution“ (explizite Abhängigkeit)

Nicht identische Meldungen

- Meldungen ohne „Substitution“ (keine explizite Abhängigkeit)
- Meldungen mit „Substitution“ (explizite Abhängigkeit)

Da im Lohnstandard-CH keine Teil-Lieferungen definiert sind, sollte in allen Fällen nur eine Lohnmeldung gültig sein.

Bei der Übertragung von mehreren Lohnmeldungen eines Absenders (Unternehmen) kann die Reihenfolge nicht garantiert werden. Daher muss der Endempfänger, die Completion-Applikation oder die spätere Verarbeitung dies selber bestimmen.

Das Verhalten des SwissdecAdapters bei Mehrfachübermittlungen ist in Kapitel 4.7 beschrieben. Weiterführende Informationen diesbezüglich in den[ENDRECREQ], Kapitel 3.11ff.

2. Funktionale Anforderungen

2.1. Ziele und Hauptfunktionen des Systems

2.1.1. Aus fachlicher Sicht

Der SwissdecAdapter empfängt über den Web Service die Lohndaten (Quellensteuermeldungen und Lohnausweise) der Firmen. Diese werden aufbereitet und dem sM-Client weitergegeben. Die Kantonalen Steuerverwaltungen können die Meldungen einsehen oder automatisch an eine Fachapplikation weiterleiten. Bei Bedarf kann ein PDF oder CSV der Meldung erstellt werden.

Mit der Einführung der ELM Version 5.0 per Januar 2021 ergeben sich neue Anforderungen:

- Erste Übermittlung mit Lohnstandard-CH Version 5.0 ist ab der QST-Abrechnung für Januar 2021 möglich
- Mehrheit der Unternehmen wird ab Januar 2021 weiterhin Lohnstandard-CH 4.0 einsetzen, da die Zertifizierung und der Rollout der Lohnsoftware Zeit benötigt -> Version 4.0 und 5.0 aus Sicht Swissdec über längere Zeit parallel im Einsatz
- Alle Kantone müssen per Januar 2021 alle Anforderungen des Lohnstandard-CH Version 5.0 unterstützen (inkl. Übergang)

Der SwissdecAdapter muss sowohl ELM Version 4.0, wie auch ELM Version 5.0 paralell empfangen können.

2.1.2. Aus technischer Sicht

Receiver: Webservice Schnittstelle zum Empfangen von Lohnmeldungen vom Swissdec Distributor.

2.2. Grobe Übersicht

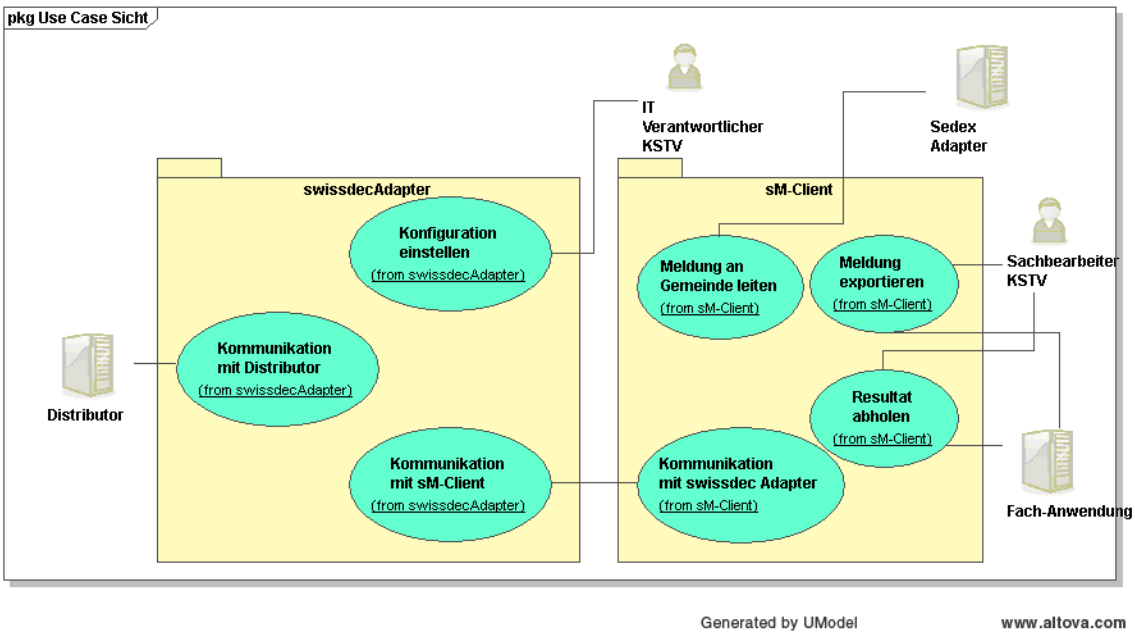


Abbildung 2.1. Use-Cases Übersicht

2.3. Detaillierte Übersicht

Die Use-Cases mit ID (UCxxx) sind offizielle Use-Cases der Swissdec Spezifikation. Diese Use-Cases werden hier nicht noch einmal spezifiziert und sind in den [ENDRECREQ] nachzulesen.

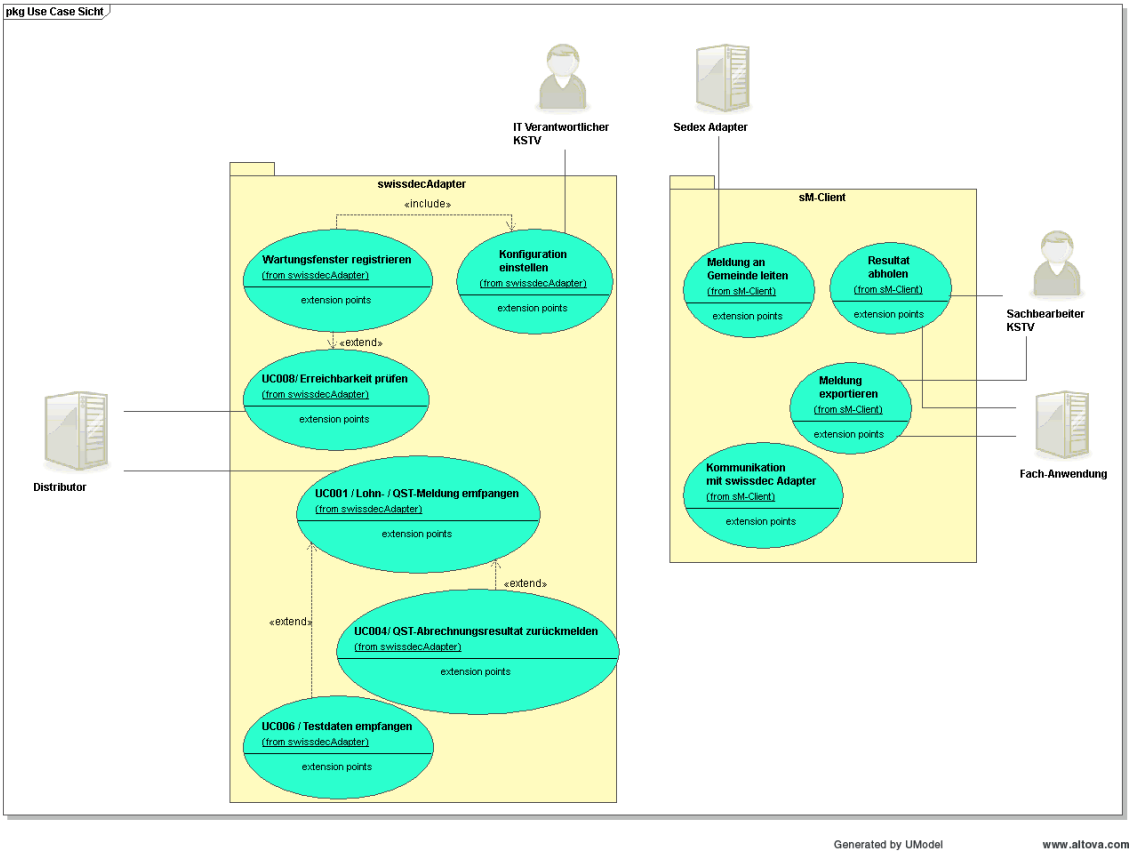


Abbildung 2.2. Use-Cases detailliert

2.4. Wartungsfenster registrieren

ID / Name	Wartungsfenster registrieren
Beschreibung	Der Distributor prüft regelmässig (~30 Minuten), ob die Endempfänger verfügbar sind. Bei diesem "Ping" kann der Endempfänger ein Wartungsfenster registrieren.
Akteure	IT Verantwortlicher KSTV, Distributor
Auslöser	Distributor
Vorbedingungen	Der Endempfänger wurde an den Distributor gekoppelt. Ein Wartungsfenster wurde in der Konfiguration des Endempfängers registriert.
Eingehende Informationen	Aktuelles Datum und Uhrzeit des Distributors
Ergebnisse	Aktuelles Datum und Uhrzeit des Endempfängers, Wartungsfenster
Nachbedingungen	Das Wartungsfenster wurde im Distributor registriert
Ablauf	
Diverses	Dieser Use-Case besteht in der Swissdec Spezifikation (UC008). Er ist hier aufgeführt, weil es möglich sein muss, ein optionales Wartungsfenster zu setzen.

	Wenn der SwissdecAdapter ein Wartungsfenster an den Distributor meldet (Ping-Antwort), wird dieses Ereignis in den Log-Dateien festgehalten.

Tabelle 2.1. Use-Case Wartungsfenster registrieren

2.5. Konfiguration einstellen

ID / Name	Konfiguration einstellen
Beschreibung	Es können verschiedene Parameter gesetzt werden.
Akteure	IT Verantwortlicher KSTV
Auslöser	IT Verantwortlicher KSTV
Vorbedingungen	Installierter SwissdecAdapter
Eingehende Informationen	Veränderte Parameter
Ergebnisse	Der SwissdecAdapter verhält sich gemäss den gesetzten Parametern.
Nachbedingungen	
Ablauf	Die gewünschten Einstellungen werden im Properties-File erfasst.
Diverses	

Tabelle 2.2. Use-Case Konfiguration einstellen

2.5.1. Konfigurationsoptionen

Folgende nicht abschliessende Liste gibt eine Übersicht über mögliche Parameter:

2.5.1.1. Receiver Konfiguration

Option	**Beschreibung**
useragent.institution	Deskriptiver Name der Institution (zum Beispiel „KSTV Bern“)
institution.tas.id	Technische Angabe, für welche Institution QST- Meldungen empfangen werden (zum Beispiel „BE“, „ZH“, „GE“, ...). MUSS richtig konfiguriert sein, da sonst Meldungen abgelehnt werden
institution.tax.id	Technische Angabe, für welche Institution Lohnausweise empfangen werden (zum Beispiel „BE“, „ZH“, „GE“, ...). MUSS richtig konfiguriert sein, da sonst Meldungen abgelehnt werden
institution.tac.id	Technische Angabe, für welche Institution Crossborder-Lohnausweise empfangen werden (zum Beispiel „BE“, „ZH“, „GE“, ...). MUSS richtig konfiguriert sein, da sonst Meldungen abgelehnt werden
tax.accept.ex	Definiert, ob Lohnausweise für im Ausland wohnhafte Personen akzeptiert werden (Default = false). Bitte sprechen Sie mit den Ihrer Fachabteilung, ob diese Option gewünscht ist. Falls ja nehmen Sie bitte Kontakt auf (JIRA oder E-Mail), da dies auf dem Distributor konfiguriert werden muss.
integration.service.protocol	Das Prokotoll, auf dem die SwissdecAdapter Integration läuft. (http:// oder https://)
integration.service.host	Der Host, auf dem die SwissdecAdapter Integration läuft.

Option	Beschreibung
integration.service.port	Der Port, auf dem die SwissdecAdapter Integration läuft.
integration.service.context	Der Context, auf dem die SwissdecAdapter Integration läuft.
org.apache.ws.security.crypto.merlin.file	Pfad zum *Keystore* mit den Zertifikaten für die Webservice-Security.
wss4j.signatureUser	Zertifikatsalias im *Keystore*, mit dem Key die Antworten signiert werden.
wss4j.privateKeyPassword	Passwort zum privaten Schlüssel (wss4j.signatureUser).
org.apache.ws.security.crypto.merlin.keystore.password	Passwort zum *Keystore*.
noSecurity	Zum Ausschalten der Web Service Security. Zum Beispiel beim Einsatz einer Application Firewall.

Tabelle 2.3. SwissdecAdapter Receiver Konfiguration

2.5.1.2. Integrationsapplikation Konfiguration

Option	Beschreibung
released.filesystem.parent.xml	Pfad zum Meldungseingang (PATH_ELM_IN). Bitte absolute Pfadangabe. Muss mit dem sM-Client korrespondieren.
failed.filesystem.parent.xml	Pfad zum "failed to receive" (PATH_ELM_FAILED_TO_RX). Bitte absolute Pfadangabe. Muss mit dem sM-Client korrespondieren.
result.filesystem.parent.xml	Pfad zum Meldungsausgang (PATH_ELM_OUT). Bitte absolute Pfadangabe. Muss mit dem sM-Client korrespondieren.
sent.filesystem.parent.xml	Pfad zum "sent" (PATH_ELM_SENT). Bitte absolute Pfadangabe. Muss mit dem sM-Client korrespondieren.
undeliverable.filesystem.parent.xml	Pfad zum "failed to send" (PATH_ELM_FAILED_TO_TX). Bitte absolute Pfadangabe. Muss mit dem sM-Client korrespondieren.
maintenance.filesystem.parent.xml	Pfad zum Ordner für Wartungsfensterdefinitionen. Bitte absolute Pfadangabe.
canton	Technische Angabe, für welche Institution Meldungen empfangen werden (zum Beispiel „BE“, „ZH“, „GE“, ...).
commune.splitting	Aktiviere Gemeindesplitting für QST-Abrechnungen. Bei aktivem Gemeindesplitting wird await.result implizit „false“, da Gemeinden keine Antworten geben können.
la.splitting	Aktiviere Lohnausweissplitting.
write.original.xml	Soll bei aktiven Splitting die vollständige original Datei auch ins ZIP geschrieben werden.
await.result	Sind QST-Abrechnungsresultate zu erwarten.
processing.default.hours	Nach wievielen Stunden kann ein allfälliges QST-Abrechnungsresultat erwartet werden.
housekeeping.days	Vorhaltezeit für Meldungen. Nach Ablauf werden die Meldungen in der internen Datenbank gelöscht. Danach kann auch kein QST-Abrechnungsresultat mehr angefragt werden.

Tabelle 2.4. SwissdecAdapter Integration Konfiguration

2.5.2. Verhalten bei Fehlkonfiguration

Es gibt zwei Arten von Fehlkonfigurationen: technische, welche den SwissdecAdapter nicht mehr starten lassen (zum Beispiel falsches Verzeichnis konfiguriert), und fachliche, welche erst zur Laufzeit bemerkt werden (zum Beispiel falsche InstitutionID). Bei technischen Fehlern sind die Log-Dateien zu konsultieren. Bei Fehlern in der Übermittlung meldet sich der Distributor via E-Mail.

3. Ergänzende Anforderungen

Die ergänzenden Anforderungen nehmen Bezug auf die [ENDRECREQ] der Swissdec. Es folgt ein Auszug der wichtigsten Anforderungen mit für die Quellensteuer relevanten Ergänzungen.

3.1. Rahmenbedingungen

Im [KZPTQST] wurden folgende Rahmenbedingungen definiert:

- Jeder Kanton verfügt über einen SwissdecAdapter
- Der SwissdecAdapter ist jederzeit („7x24“ Stunden) verfügbar. Weitere Informationen dazu im Kapitel Abschnitt 3.5, „Verfügbarkeit“ „Verfügbarkeit".
- Der SwissdecAdapter läuft als Web Service und wird nur vom Swissdec Distributor via Internet angesprochen.
- Die KSTV Firewalls müssen deshalb für den Swissdec Distributor geöffnet werden. Die Verbindung zum SwissdecAdapter wird vom Swissdec Distributor mittels Authentisierung initiiert (Mutual Authentication bzw. two-way SSL, offizielle Standards und „Best Practices“).
- Zusätzlich sind die Nutzdaten noch auf SOAP-Ebene signiert und verschlüsselt.
- Falls eine zusätzliche Transportschicht nach dem sM-Client (z.B. zentraler Adapter und Weiterleitung auf Kantonsnetz) eingesetzt wird, so fällt dies in die Zuständigkeit der KSTV.
- SMTP-Server für interne und externe Benachrichtigung (Email) über Prozess-Status
- Neben der Möglichkeit die sM-Clients zu verwenden, können die KSTV auch einen direkten Anschluss an den Lohnstandard-CH realisieren.

3.2. Anforderungen Infrastruktur

3.2.1. Hardware

Da der SwissdecAdapter auf Java basiert, ist die Lösung grundsätzlich Plattformunabhängig. Die Basis, das Produkt STEP, wird auf x86-32bit und x86-64bit Plattformen betrieben. Einzig der Arbeitsspeicher (RAM) limitiert die Grösse der zu verarbeitenden Meldungen. Wir schlagen folgende Systemspezifikationen vor:

- Aktuelle 64bit Hardware
- 4GB Arbeitsspeicher für die Java-Umgebung
- Genügend Festplattenkapazität, um die Files temporär speichern zu können (Eine Meldung mit 6'000 Personen kann über 50 MB gross sein)

3.2.2. Software

- Plattformen (Betriebssysteme): Linux oder Windows. Vorzugsweise 64bit.
- Java 8 oder Java 11 (getestet wird mit OpenJDK8 und OpenJDK11 und künftige OpenJDK LTS Versionen)

3.2.3. SSL/TLS Terminierung

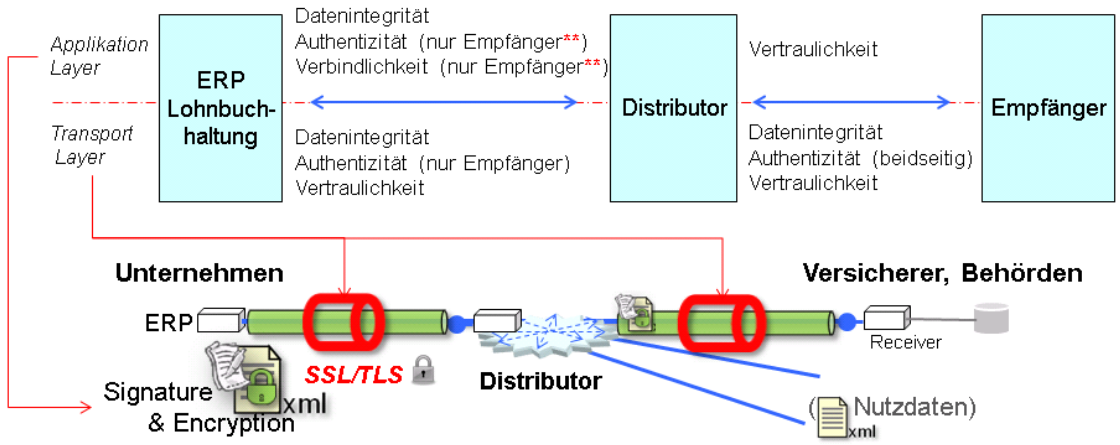


Abbildung 3.1. Installationsskizze SwissdecAdapter

Die Verbindung zwischen Distributor und Endempfänger ist mittels SSL/TLS mit Klienten-Authentifizierung (Mutual Authentication) gesichert. Diese Verbindungen können auf die IP des Distributors restriktiert werden.

3.2.3.1. Terminierung auf SwissdecAdapter

Wird die SSL/TLS Verbindung erst auf dem SwissdecAdapter terminiert, müssen Firewalls, welche topologisch vor dem SwissdecAdapter angesiedelt sind, eingehende HTTPS-Verbindungen bis zum SwissdecAdapter erlauben.

3.2.3.2. Terminierung auf Entry-Server

Wenn die SSL/TLS Verbindung auf einem Reverse Proxy terminiert wird, muss dieser auch für das "Mutual Authentication" (Klient-Authentifizierung) konfiguriert werden. Zusätzlich müssen die Webservice-Endpunkte aus dem Internet verfügbar sein. Dies liegt im Verantwortungsbereich des Betreibers.

3.3. Zusammenfassung einer Institution

Meldungen werden auf dem Swissdec Distributor pro Adressat gefiltert. Eine Meldung wird für jede Domäne einzeln an den Endempfänger verschickt. Das bedeutet, dass der SwissdecAdapter mehrere Kopplungen verfügen muss.

Jede KSTV benötigt mindestens zwei TaxAtSource-Kopplungen (Quellensteuer in den Versionen ELMv4/20130514 sowie ELMv5/20200220). Ein Kanton, welcher Lohnausweise empfangen darf hat zusätzlich die Domäne Tax (Lohnausweise), Kantone mit Abkommen für Grenzgänger haben zusätzlich die Domäne TaxCrossborder gekoppelt.

- ELM V5.0 : Tax (Lohnausweise)
- ELM V5.0 : TaxAtSource (Quellensteuermeldungen)
- ELM V5.0 : TaxCrossborder (Meldung für Grenzgänger)
- ELM V4.0 : Tax (Lohnausweise)
- ELM V4.0 : TaxAtSource (Quellensteuermeldungen)

3.4. Kommunikationsstandards

Die Standardkopplung muss auf der Web Service Technologie (SOAP Version 1.1, WSDL Version 1.1 und WSS Version 1.1) basieren. Die Daten müssen neben dem Transport Layer (two-way SSL, Mutual Authentication) zusätzlich auf Meldungs-Ebene gemäss WSS verschlüsselt werden.

Die Problematik für die KSTV stellt sich nicht in der Webservice-Security (Signatur und Verschlüsselung auf SOAP-Ebene). Diese wird durch den SwissdecAdapter abgehandelt. Da auf Verbindungsebene SSL/TLS mit Mutual Authentication zu Einsatz kommt, wird die SSL/TLS-Terminierung etwas komplexer. Das heisst nebst der Identifizierung (einfaches SSL/TLS) muss die KSTV vom Klienten, in unserem Fall immer der Distributor, zwingend ein Zertifikat verlangen (Mutual Authentication).

Folgende Zertifikate müssen pro SwissdecAdapter verfügbar sein:

- Swissdec CA-Zertifikat, zur Validierung der Signatur des Distributors
- Privater Schlüssel, zum Entschlüsseln der Nachricht
- Zertifikat zum Verschlüsseln der Antworten
- Schlüsselpaar, für Mutual Authentication auf SSL/TLS Ebene
- CA-Zertifikat, um das Klient-Zertifikat auf SSL/TLS Ebene zu prüfen

Grundsätzlich liegt die Hoheit über die Zertifikate bei den Endempfängern. Swissdec hat eine nicht akkreditierte CA und kann Zertifikate für alle Bereiche bereitstellen. Swissdec empfiehlt:

- SSL/TLS Serverzertifikat organisiert durch KSTV
- SSL/TLS Klienten Zertifikat der Swissdec (Die KSTV Konfiguriert das Swissdec CA-Zertifikat für die SSL/TLS Mutual Authentication)
- Webservice Zertifikate der Swissdec

3.5. Verfügbarkeit

Die Betrachtungseinheit umfasst den Distributor und alle gekoppelten Endempfänger, das heisst der SSL/Kunde (Lohndatenquelle) erlebt das ganze System als Einheit. Sollte ein Endempfänger nicht in geforderter Qualität betrieben werden, vermindert dieser Empfänger die Zuverlässigkeit des ganzen Systems. Alle Teilnehmer müssen sich deshalb auf eine minimale Zuverlässigkeit einigen.

Anforderung aus dem Lohnstandard-CH (ELM)

Die gesamte Übermittlung in der 1. Phase: m2m (Machine to Machine) erfolgt in „Echtzeit“. (7x24h Internet-Verfügbarkeit)

Diese Anforderung hat für den Empfänger folgende Konsequenzen

- auch die Institutionen bzw. ihre Endempfänger müssen mindestens zum Empfangen der Daten einen 7x24h Dienst anbieten.
- Geplante Unterbrüche müssen an Randzeiten durchgeführt werden und müssen vorher mittels Wartungsfenster angekündigt werden.

Zielorientiertes Vorgehen bezüglich des Themas Verfügbarkeit: Wir möchten eine kundenorientierte Sicht einnehmen. Die Verfügbarkeiten der Systeme sind als zukünftige Zielwerte zu verstehen. Damit werden die Unternehmen motiviert, ihre Meldungen elektronisch zu übermitteln.

Ziel ist eine Pragmatische Lösung = „lightweight construction“ und „Best Effort“

3.6. Performance / Durchsatz

Ergänzungen zur Swissdec Spezifikation:

- Mindestens 6'000 Mitarbeiter in einer Lohnmeldung
- 1000 Mitarbeiter in einer Minute

Diese Anforderungen sind für aktuelle Hardware-/Softwaresysteme kein Problem. Falls der SwissdecAdapter auf ein er virtuellen Umgebung betrieben wird, muss diese so dimensioniert werden, dass die Minimas der Swissdec Spezifikation erfüllt werden.

3.7. Mehrfachübermittlungen

In der Swissdec Spezifikation wird unterschieden zwischen identischen Meldungen, so genannten Doubletten, und nicht identischen Meldungen. Doubletten werden vom Distributor markiert und vom Swissdec Adapter verarbeitet, jedoch nicht an den sM-Client weitergegeben. Der SwissdecAdapter liefert in diesem Fall die Quittung der ursprünglichen Meldung an den Distributor und schreibt einen entsprechenden Eintrag ins Logfile. Das heisst das Unternehmen bekommt die gleiche Antwort und die gleichen Credentials wie für die originale Meldung. Dies bedeutet das für beide Meldungen das gleiche QST-Abrechnungsresultat geliefert wird. Bei nicht identischen Meldungen muss im sM-Client optisch festgehalten werden, dass es sich um eine Mehrfachübermittlung handelt. Diese Information liefert der Distributor in der SOAP-Nachricht mit, sofern diese vom SSL/Kunde explizit gesetzt wurde. Weiterführende Informationen diesbezüglich in den [ENDRECREQ] , Kapitel 3.11ff.

```
<DeclareSalaryConsumer
      xmlns="http://www.swissdec.ch/schema/sd/20090803/SalaryDeclarationConsumerServiceTypes"
      xmlns:sd="http://www.swissdec.ch/schema/sd/20090803/SalaryDeclaration"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <DistributorRequestContext
         xmlns="http://www.swissdec.ch/schema/sd/20090803/SalaryDeclarationConsumerContainer">
      …
<SubstitutionMapping>
     <MappedPredecessorDistributorRequestID>95120</MappedPredecessorDistributorRequestID>
        <MappedPredecessorConsumerResponseID>12500</MappedPredecessorConsumerResponseID>
   </SubstitutionMapping>
…
</DeclareSalaryConsumer>
```

In diesem Fragment eines ELM-Request wurde vom ERP-System eine Ersatzmeldung für den Request "95120" geschickt. Das Element "SubstitutionMapping" ist optional, das heisst, es wird nur gesetzt falls es sich um eine Ersatzmeldung handelt.

4. Architektur

4.1. Systemübersicht

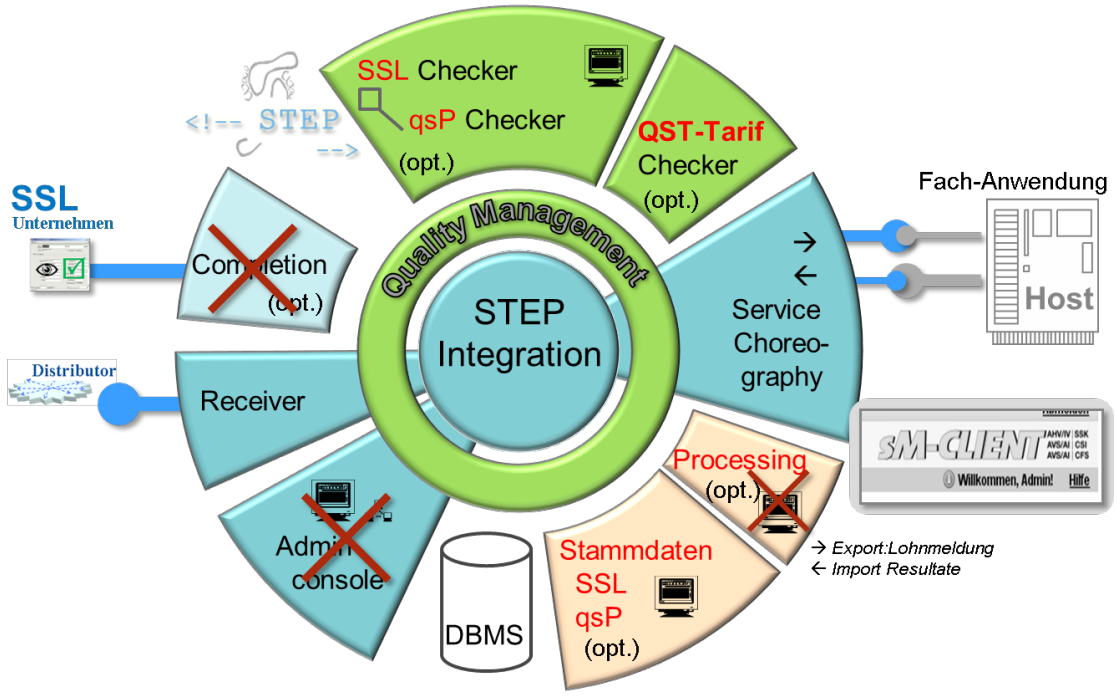


Abbildung 4.1. Architektur STEP/SwissdecAdapter

4.2. Systemteile

Um in verschiedene Umgebungen integriert werden zu können, ist der SwissdecAdapter in zwei Teile unterteilt:

Receiver: Die Receiver-Komponente startet einen Web Service Dienst, welcher von Internet erreichbar sein muss. Er ist verantwortlich für das Abwickeln der WS-Security (Entschlüsseln der Meldung, Signaturprüfung, Verschlüsseln und Signieren der Antwort), validiert und plausibilisiert die Meldung und gibt sie in die Integrationskomponente weiter.

Integration: Die Integrations-Komponente ist für die Orchestrierung der Prozesse zuständig. Sie startet einen Web Service Dienst, welcher von der Receiver-Komponente bei eingehenden Meldungen konsultiert wird. Die komplette Business Logik wird hier abgearbeitet, Laufzeitdaten in einer Datenbank (Apache Derby) gehalten sowie Umsysteme angesprochen.

4.2.1. Schnittstelle Receiver - Integration

Die Kommunikation zwischen Receiver- und Integrationskomponente wird über HTTP (SOAP) geführt. Die Kommunikation wird immer von der Receiver-Komponente (Client) initiiert. Es gibt drei Operationen, analog der Swissdec-Schnittstelle.

Swissdec Operation	Adapter Operation	Parameter
PingConsumer	pingIntegration	In: keine Out: Informationen über allfällige Wartungsfenster
DeclareSalaryConsumer	saveSalaryDeclaration	In: Metadaten der Übermittlung In: SOAP Request Out: Quittungsinformationen
GetResultFromDeclareSalaryConsumer	getResult	In: Schlüssel zur Identifikation der Meldung In: Passwort aus der Quittung Out: Abrechnungsresultat, falls Vorhanden

Tabelle 4.1. Schnittstelle Receiver - Integration

4.2.2. Installationsvarianten

Grundsätzlich müssen folgende Bedingungen erfüllt sein:

- Receiver muss aus dem Internet erreichbar sein
- Swissdec-Konforme Kopplung an den Distributor
- Integration muss vom Receiver erreichbar sein
- muss ein Dateisystem mit dem sM-Client teilen (analog sedex-Adapter)

Daraus resultieren 2 Installationsvarianten:

4.2.2.1. En bloc

Alle Komponenten des SwissdecAdapters werden in einer Installation, auf einer physikalischen Maschine installiert. Die Kommunikation geschieht auch hier via HTTP (SOAP). Falls der sM-Client in einer anderen Zone installiert ist, muss ein Dateisystem zonenübergreiffend verfügbar gemacht werden.

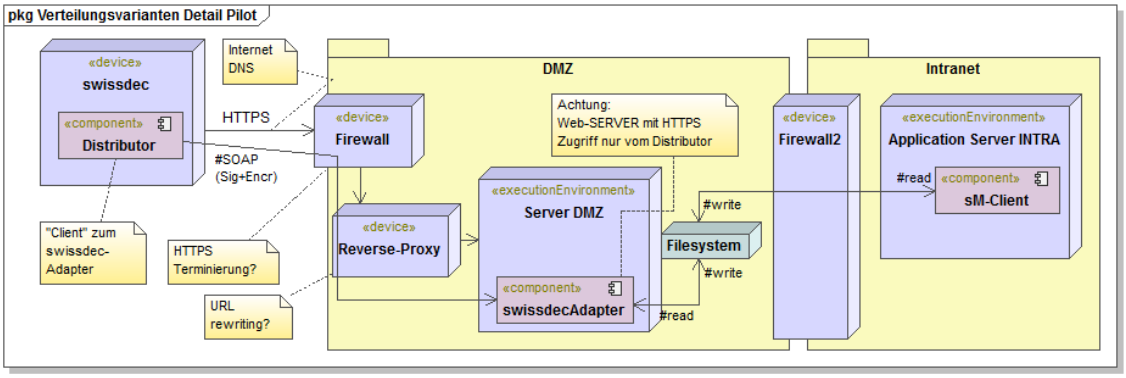


Abbildung 4.2. Installationsvariante En bloc

4.2.2.2. Verteilt

Typischerweise läuft die Receiver Komponente in der DMZ, die Integrationskomponente sowie der sM-Client im Intranet. Die Kommunikation zwischen den Zonen geschieht via HTTP(SOAP) und wird vom Receiver initiiert (Klient). Es muss sichergestellt werden, dass der Web Service Endpunkt der Integration vom Receiver erreichbar ist.

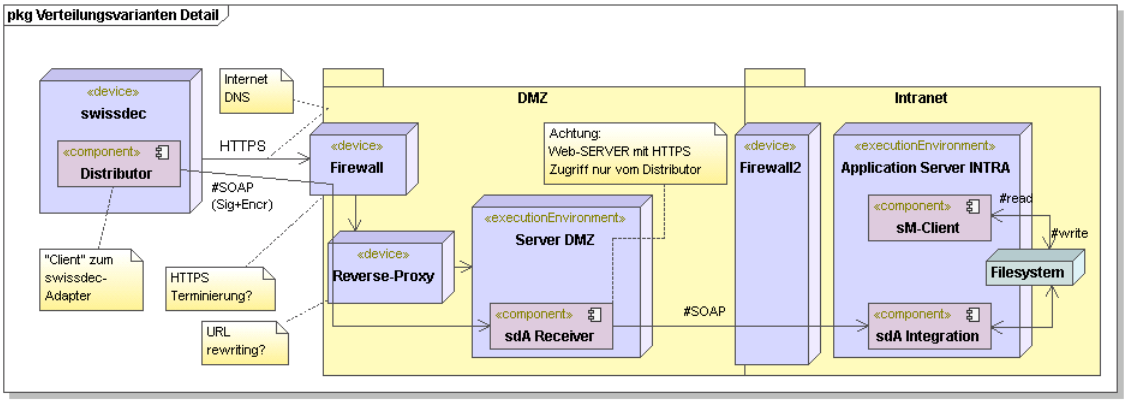


Abbildung 4.3. Installationsvariante Verteilt

5. Schnittstellen

5.1. Swissdec

Die Swissdec-Schnittstelle wurde im itServe Produkt STEP bereits implementiert, und bei mehreren Kunden installiert und von der Swissdec abgenommen. Die Anforderungen für Swissdec Endempfänger sind in folgenden Dokumenten der Swissdec-Spezifikation beschrieben:

- [SDOVERVIEW]
- [ENDRECREQ]
- [ACKNOTI]
- [RECSEC]

5.2. sM-Client

Diese Schnittstelle ist als Dateisystem-Schnittstelle definiert und dient zum Datenaustausch in beide Richtungen. Das heisst, der SwissdecAdapter schreibt eingehende Anfragen auf das Dateisystem und liest die Antworten vom sM-Client. Verarbeitet werden folgende Daten:

- QST-Abrechnung (eingehend)
- QST-Abrechnung aufgeteilt nach Gemeinden (eingehend)
- Lohnausweise (eingehend)
- Grenzgänger-Meldung (eingehend)
- QST-Abrechnungsresultat (ausgehend)

5.2.1. Format der Daten

Grundsätzlich werden Swissdec Lohnstandard XML Dateien auf der Schnittstelle ausgetauscht. Der SwissdecAdapter kennt dabei sowohl den Lohnstandard Version 4.0, wie auch den Lohnstandard Version 5.0.

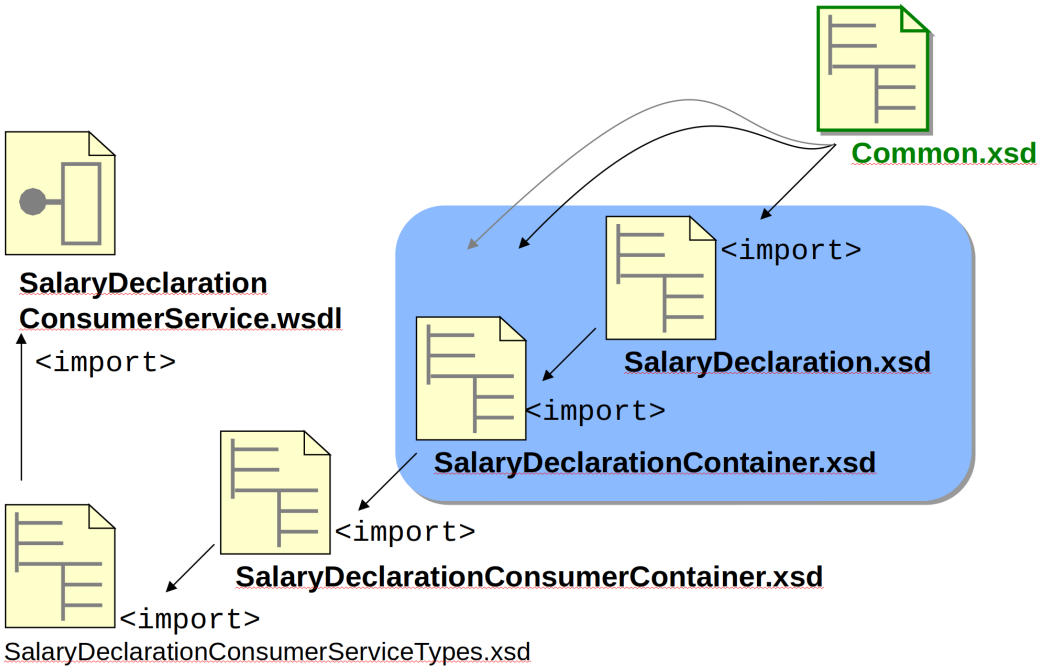


Abbildung 5.1. Swissdec Schema-Imports

5.2.1.1. Der SwissdecAdapter schreibt folgende Meldungen:

- QST-Abrechnung:

  Entschlüsselter SOAP-Request, mit "SalaryDeclarationConsumer" Body.

- QST-Abrechnung aufgeteilt nach Gemeinden:

  Mehrere entschlüsselte SOAP-Requests, mit gefiltertem "SalaryDeclarationConsumer" Body.

- Lohnausweise:

  Entschlüsselter SOAP-Request, mit "SalaryDeclarationConsumer" Body.

- Grenzgänger-Meldung:

  Entschlüsselter SOAP-Request, mit "SalaryDeclarationConsumer" Body.

5.2.1.2. Der SwissdecAdapter liest folgende Meldungen:

- QST-Abrechnungsresultat:

  "SalaryDeclarationConsumerResponse" mit Antwort.

5.2.2. Verarbeitungsprinzipien

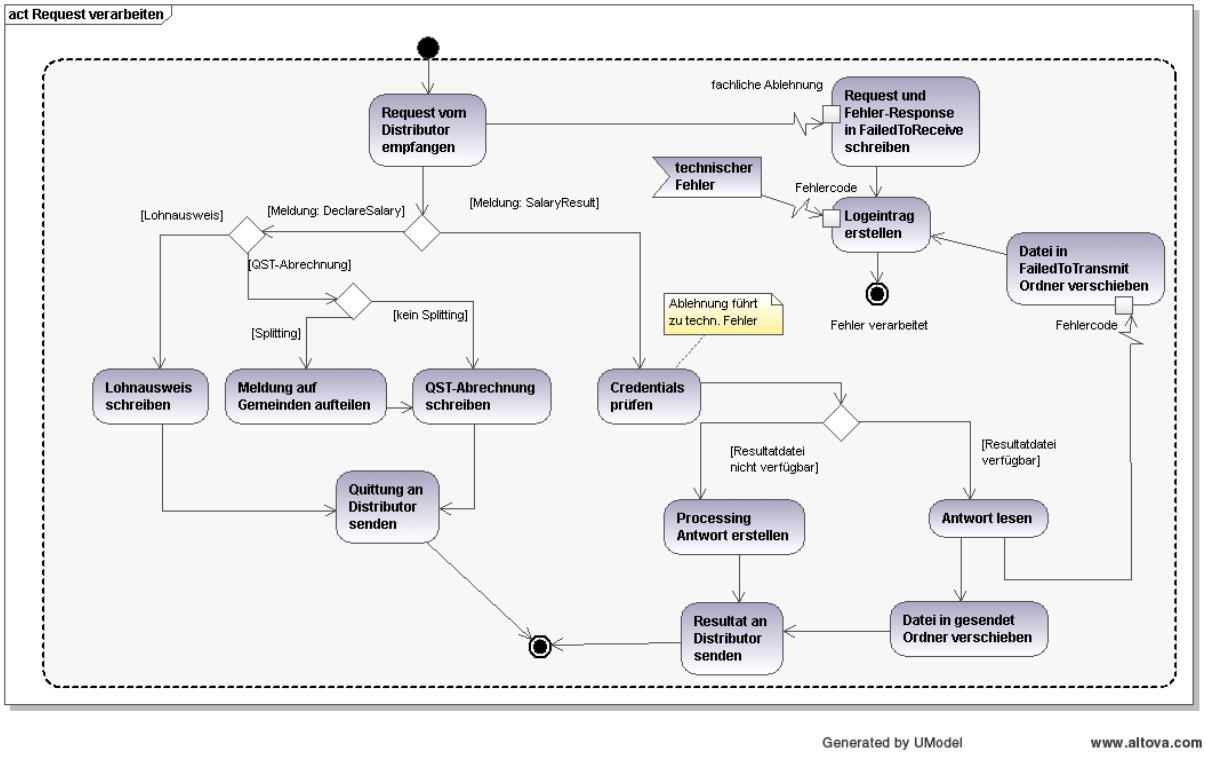


Abbildung 5.2. Request verarbeiten

5.2.2.1. Allgemein

- Die technischen Prüfungen (Entschlüsselung, Signatur- und Schemavalidierung, …) geschehen im Schritt „Request vom Distributor empfangen“
- Alle Meldungstypen (Lohnausweise, QST-Abrechnung und Grenzgänger-Meldung) in allen Versionen (ELM V4.0 und ELM V5.0) werden in die selben Ordner geschrieben
- Der SwissdecAdapter kann ELM V4.0 Meldungen nach ELM V5.0 konvertieren
  - falls QST-Abrechnungsresultate generiert werden, müssen diese in der Version der Meldungen auf der Schnittelle sein
- Gesamthaft werden 5 Ordner benötigt:
  1. PATH_ELM_IN: Meldungseingang (Lohnausweis und QST-Abrechnung)
  2. PATH_ELM_FAILED_TO_RX: Failed to receive (Lohnausweis und QST-Abrechnung, Fehler beim Empfang)
  3. PATH_ELM_OUT: Meldungsausgang (QST-Abrechnungsresultat)
  4. PATH_ELM_SENT: Gesendet
  5. PATH_ELM_FAILED_TO_RX: Failed to transmit (für fehlerhafte QST-Abrechnungsresultate)
- Alle Dateien auf der Schnittstelle sind im ZIP Format komprimiert
- Die Nutzdaten (Meldungen) sind als XML im Swissdec Format
- Da Swissdec Lohnmeldungen ausgetauscht werden, sind alle Meldungen gemäss ELM Schema definiert.
- Die Dateinamen für Meldungen sind wie folgt:
  - ["Tax"|"TaxAtSource"|"Tax5"|"TaxAtSource5"|"TaxCrossborder5"]_DeclarationId_ResultIdentifier_["MIXD"|"SPLT"].zip
- Die Dateien sollen in der Reihenfolge des Schreibens gelesen werden (First-In-First-Out)
- Um die Zustellung zu garantieren ([ENDRECREQ], Kapitel 3.11) werden die Dateien zunächst mit der Datei-Endung ".tmp" geschrieben. Nachdem das File komplett geschrieben wurde, wird das File umbenannt ("Atomic write"). Dieses Verhalten wird auch vom sM-Client erwartet.
- Die Originalmeldung wird in allen ZIPs mitgeliefert
  - Dateiname ["Tax"|"TaxAtSource"|"Tax5"|"TaxAtSource5"|"TaxCrossborder5"]_DeclarationId_ResultIdentifier_MIXD.xml
- In jedem ZIP ist eine XML Datei mit Meta-Informationen zur Meldung
  - Dateiname envelope.xml
  - Inhalt:
    - DeclarationID
    - HR-RC-Name
    - UID-BFS
    - CantonID
    - TransmissionDate
    - TestCase
    - Substitution (mit Original RequestID)
    - Domäne
    - ELMVersion

- OriginalVersion
- ResultIdentifier
- Splitting
- Auflistung der Dateien im ZIP

– Das Beispiel A.1, „envelope.xsd“ dazu ist im Anhang

5.2.2.2. Lohnausweis

- Der Empfang des Lohnausweises erfolgt direkt, ohne Korrekturmeldung
- Lohnausweise kommen im Paket pro Firma, also mehrere Personen in einer Meldung
  – Eine Person kann mehrere Lohnausweise/Rentenbescheinigungen haben
  – Die Meldung kann pro Lohnausweis gesplittet werden, es kann also mehrere Dateien für eine Person geben
  – Beispiel Dateinamen für Meldung:
    - Nicht Gesplittet: Tax5_Dabc123def_983kjd89c_MIXD.zip
      – Tax für Lohnausweis in ELM V4.0, Tax5 für Lohnausweis in ELM V5.0
      – DeclarationId "Dabc123def"
      – ResultIdentifier "983kjd89c"
      – MIXD für nicht gesplittet
      – Die ungesplittete Datei im ZIP Tax5_Dabc123def_983kjd89c_MIXD.xml
    - Gesplittet: Tax5_ Dabc123def _983kjd89c_SPLT.zip
      – Tax für Lohnausweis
      – DeclarationId "Dabc123def"
      – ResultIdentifier "983kjd89c"
      – SPLT für gesplittet
      – Die gesplitteten Dateien im ZIP Tax5_ Dabc123def_983kjd89c_SVNR_LAUFNUMMER.xml

5.2.2.3. QST-Abrechnung

- Der Empfang der Quellensteuerabrechnung erfolgt direkt. Es sind jedoch Korrekturmeldungen möglich.
- Quellensteuerabrechnungen kommen im Paket pro Firma, also mehrere Personen in einer Abrechnung
  – Kann gesplittet werden nach Gemeinde
- Beispiel Dateinamen für Meldung:
  – Nicht Gesplittet: TaxAtSource5_Dabc123def_983kjd89c_MIXD.zip
    - TaxAtSource für QST-Meldung in ELM V4.0, TaxAtSource5 für Meldung in ELM V5.0
    - DeclarationId "Dabc123def"
    - ResultIdentifier "983kjd89c"
    - MIXD für nicht gesplittet
    - Die ungesplittete Datei im ZIP TaxAtSource5_Dabc123def_983kjd89c_MIXD.xml
  – Gesplittet: TaxAtSource5_Dabc123def_983kjd89c_SPLT.zip

- TaxAtSource für QST-Meldung
- DeclarationId "Dabc123def"
- ResultIdentifier "983kjd89c"
- SPLT für gesplittet
- Die gesplitteten Dateien im ZIP TaxAtSource5_Dabc123def_983kjd89c_GEMEINDENR.xml

5.2.2.4. TaxCrossborder

- Der Empfang der TaxCrossborder erfolgt direkt. Es sind jedoch Korrekturmeldungen möglich.
- Quellensteuerabrechnungen kommen im Paket pro Firma, also mehrere Personen in einer Abrechnung
- Beispiel Dateinamen für Meldung:
  - Nicht Gesplittet: TaxCrossborder5_Dabc123def_983kjd89c_MIXD.zip
    - TaxCrossborder5_ für Meldung in Version 5.0
    - DeclarationId "Dabc123def"
    - ResultIdentifier "983kjd89c"
    - MIXD für nicht gesplittet
    - Die ungesplittete Datei im ZIP TaxCrossborder5_Dabc123def_983kjd89c_MIXD.xml

5.2.2.5. QST-Abrechnungsresultat

- Es kann Konfiguriert werden, ob QST-Abrechnungsresultate geliefert werden können
- Falls kein QST-Abrechnungsresultat verfügbar ist, wird eine „QST- Antwort pendent“ (Kapitel 7.3.5) generiert. Dies geschieht im Schritt „Processing Antwort generieren“ (Abb. 11)
- Die Dateinamen für Antworten sind wie folgt:
  - DeclarationId_ResultIdentifier.zip
- Beispiel Dateiname für Antwort:
  - Dabc123def_983kjd89c.zip
    - DeclarationId "Dabc123def"
    - ResultIdentifier "983kjd89c"
    - Die XML-Datei im ZIP Dabc123def_983kjd89c.xml
- Die Antworten werden zur Zeit des Requests (GetResult) gelesen
- Die Antwort-XML enthalten alle Informationen, wie im Swissdec-Schema definiert:
  - Status der Verarbeitung
  - Fehlermeldungen (Notifications gemäss[ACKNOTI])
  - Beträge, falls nicht korrekt gemeldet
  - QST-Codes, falls nicht korrekt gemeldet
  - Beispiel im Anhang, Kapitel 7.3.4
- Gemeinden können keine Antworten geben, das heisst keine Rückantwort via sedex
- Sobald die Files konsumiert wurden werden sie verschoben

– Bei Erfolg in den Ordner „Gesendet“ (PATH_ELM_SENT)

– Bei Fehler in den Ordner „Failed to transmit“ (PATH_ELM_FAILED_TO_TX)

5.2.3. Behandlung von Fehlern

Fehlerquelle	Art des Fehlers	Verhalten des Adapters
ERP/"SSL/Kunde"	Fachlich	Die Meldungen, die aufgrund von fachlichen Tests abgewiesen werden, schreibt der SwissdecAdapter inklusive die (Fehler-) Antwort in den Ordner „failed to receive“ (PATH_ELM_FAIL_TO_RX).
Distributor	Technisch	Falls im SwissdecAdapter technische Probleme auftreten, werden diese ins Log-File geschrieben.
Filesystem	Technisch	Falls das Filesystem nicht verfügbar ist, wird dem SSL/Kunden mit einem Fehler geantwortet, und falls möglich in die Log-Datei geschrieben.
sM-Client	Fachlich / Technisch	Falls das QST-Abrechnungsresultat vom sM-Client Mängel aufweist, wird dem SSL/Kunden mit einem Fehler geantwortet und die fehlerhafte Antwort in den Ordner „failed to send“ (PATH_ELM_FAIL_TO_TX) geschrieben.

Tabelle 5.1. Fehler im SwissdecAdapter

5.3. Kostenpflichtige Zusatzfunktionalität

Folgende Schnittstellen sind denkbar, jedoch nicht im Umfang dieses Projekts und des Produktes STEP.

5.3.1. SSL/ Kunde (Schuldner steuerbarer Leistung) Checker

Wenn der SwissdecAdapter Informationen zu den bekannten SSLs abrufen kann, könnte eine Prüfung zur Zeit des Empfangs der Meldung durchgeführt werden. Integriert könnte dies über ein Stammdaten-File (zum Beispiel CSV) oder über ein Call-Schnittstelle ins Backend werden. Der Vorteil würde darin liegen, dass der SwissdecAdapter autonom und bereits zur Empfangszeit der Meldung gewisse Prüfungen durchführen könnte.

5.3.2. Tarifcode Checker

Wenn der SwissdecAdapter Informationen zu den bekannten Tarifcodes abrufen kann, könnte eine Prüfung zur Zeit des Empfangs der Meldung durchgeführt werden. Man könnte dies über ein Stammdatenfile (zum Beispiel CSV) oder über eine Call-Schnittstelle ins Backend realisieren. Der Vorteil würde darin liegen, dass der SwissdecAdapter autonom und bereits zur Empfangszeit der Meldung gewisse Prüfungen durchführen könnte.

5.3.3. Fach-Anwendung

Grundsätzlich wäre es auch denkbar, dass der SwissdecAdapter direkt mit den Fach-Anwendungen kommuniziert. Dies ist vor allem dann sinnvoll, wenn die Fach-Anwendung synchron arbeiten kann. Dies würde bedeuten, dass eine so genannte "Vollintegration" von den ERP-Systemen direkt in die Backend-Systeme der KSTV realisiert werden könnte. Interessant wird diese Vollintegration bei Kantonalen Steuerverwaltungen, die viele Quellensteuer-Abrechnungen via ELM erhalten. Weiter ist eine Vollintegration langfristig sicher ein Wunsch der SSL/Kunden.

5.3.4. Mail

Der SwissdecAdapter könnte, falls ein SMTP-Server verfügbar ist, für gewisse Statusänderungen der Übermittlung Mails verschicken. Dies sowohl intern an den Sachbearbeiter, wie auch den SSL/Kunden, zum Beispiel bei Verfügbarkeit der Antwort.

A. Anhang

A.1. Referenzierte Dokumente

[KZPTQST] *Konzept für das Projekt "Lohnstandard-Quellensteuer"*. Swissdec. 20110331.

[SDOVERVIEW] *Übersicht zum Swissdec Standard*. Swissdec. 20200220.

[ENDRECREQ] *Anforderungen an Endempfänger*. Swissdec. 20200220.

[ACKNOTI] *Spezifikation der Mitteilungen*. Swissdec. 20190903.

[RECSEC] *Security- Anforderungen an Endempfänger*. Swissdec. 20160818.

A.2. Glossar

Keystore	Ein Keystore ist eine Datenbank von Schlüsseln und Zertifikaten in einer Datei.

A.3. Beispiel Dateien

```
            <!--

  Copyright 1996-2021 itServe AG. All rights reserved.

  This software is the proprietary information of itServe AG
  Bern Switzerland. Use is subject to license terms.

-->
<xs:schema targetNamespace="http://www.itserve.ch/step/shared/smc"
        xmlns:tns="http://www.itserve.ch/step/shared/smc"
        xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
        attributeFormDefault="unqualified" version="0.0">
   <xs:element name="envelope">
      <xs:complexType>
         <xs:sequence>
            <xs:element name="DeclarationID" type="tns:IDType"/>
            <xs:element name="DistributorRequestID" type="tns:IDType"/>
            <xs:element name="HR-RC-Name" type="xs:string"/>
            <xs:element name="UID-BFS" type="tns:UID-BFSType"/>
            <xs:element name="CantonID" type="tns:CantonType"/>
            <xs:element name="TransmissionDate" type="xs:dateTime"/>
            <xs:element name="TestCase" type="tns:EmptyType" minOccurs="0"/>
            <xs:element name="SubstitutionMapping" type="tns:SubstitutionMappingType"
                  minOccurs="0"/>
            <xs:element name="Domain" type="tns:DomainType"/>
            <xs:element name="ELMVersion" type="xs:string"/>
            <xs:element name="OriginalVersion" type="xs:string"/>
            <xs:element name="ResultIdentifier" type="xs:string"/>
            <xs:element name="Splittet" type="tns:EmptyType" minOccurs="0"/>
            <xs:element name="Files" type="tns:FilesType"/>
         </xs:sequence>
      </xs:complexType>
   </xs:element>

   <xs:complexType name="SubstitutionMappingType">
      <xs:sequence>
         <xs:element name="MappedPredecessorDistributorRequestID" type="tns:IDType"/>
         <xs:element name="MappedPredecessorConsumerResponseID" type="tns:IDType"/>
      </xs:sequence>
   </xs:complexType>

   <xs:complexType name="FilesType">
      <xs:sequence>
         <xs:element maxOccurs="unbounded" name="File" type="xs:string"/>
      </xs:sequence>
   </xs:complexType>

   <xs:complexType name="EmptyType"/>

   <xs:simpleType name="IDType">
      <xs:restriction base="xs:string">
         <xs:minLength value="1"/>
         <xs:maxLength value="255"/>
      </xs:restriction>
```

```
  </xs:simpleType>

  <xs:simpleType name="UID-BFSType">
    <xs:restriction base="xs:NMTOKEN">
      <xs:pattern value="CHE-[0-9]{3}\.[0-9]{3}\.[0-9]{3}"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="CantonType">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="AG"/>
      <xs:enumeration value="AI"/>
      <xs:enumeration value="AR"/>
      <xs:enumeration value="BE"/>
      <xs:enumeration value="BL"/>
      <xs:enumeration value="BS"/>
      <xs:enumeration value="FR"/>
      <xs:enumeration value="GE"/>
      <xs:enumeration value="GL"/>
      <xs:enumeration value="GR"/>
      <xs:enumeration value="JU"/>
      <xs:enumeration value="LU"/>
      <xs:enumeration value="NE"/>
      <xs:enumeration value="NW"/>
      <xs:enumeration value="OW"/>
      <xs:enumeration value="SG"/>
      <xs:enumeration value="SH"/>
      <xs:enumeration value="SO"/>
      <xs:enumeration value="SZ"/>
      <xs:enumeration value="TG"/>
      <xs:enumeration value="TI"/>
      <xs:enumeration value="UR"/>
      <xs:enumeration value="VD"/>
      <xs:enumeration value="VS"/>
      <xs:enumeration value="ZG"/>
      <xs:enumeration value="ZH"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="DomainType">
    <xs:restriction base="xs:NMTOKEN">
      <xs:enumeration value="TAC"/>
      <xs:enumeration value="TAS"/>
      <xs:enumeration value="TAX"/>
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
```

Beispiel A.1. envelope.xsd